HOMENETRESCUE: UM SERVIÇO SDN PARA DETECÇÃO E SOLUÇÃO DE PROBLEMAS EM REDES DOMÉSTICAS

ALISSON RODRIGUES ALVES

HOMENETRESCUE: UM SERVIÇO SDN PARA DETECÇÃO E SOLUÇÃO DE PROBLEMAS EM REDES DOMÉSTICAS

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

Orientador: Daniel Fernandes Macedo
Coorientador: Marcos Augusto Menezes Vieira

Belo Horizonte
Outubro de 2017

© 2017, Alisson Rodrigues Alves.
Todos os direitos reservados

Ficha catalográfica elaborada pela Biblioteca do ICEx - UFMG

Alves, Alisson Rodrigues

A474h HomeNetRescue: um serviço SDN para detecção e solução de problemas em redes domésticas / Alisson Rodrigues Alves – Belo Horizonte, 2017.
xxiii, 87 f.: il.; 29 cm.

Dissertação (mestrado) - Universidade Federal de Minas Gerais – Departamento de Ciência da Computação.

Orientador: Daniel Fernandes Macedo
Coorientador: Marcos Augusto Menezes Vieira

1. Computação – Teses. 2. Redes de computadores. 3. Redes de computadores – Administração. I. Orientador. II. Coorientador. III. Título.

CDU 519.6*22(043)

UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

HomeNetRescue: Um Serviço SDN para Detecção e Solução de Problemas em Redes Domésticas

ALISSON RODRIGUES ALVES

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

PROF. DANIEL FERNANDES MACEDO - Orientador
Departamento de Ciência da Computação - UFMG

PROF. MARCOS AUGUSTO MENEZES VIEIRA - Coorientador
Departamento de Ciência da Computação - UFMG

PROF. ALEX BORGES VIEIRA
Departamento de Ciência da Computação - UFJF

PROF. JOSÉ MARCOS SILVA NOGUEIRA
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 27 de Outubro de 2017.

Aos meus familiares, aos meus orientadores e aos meus amigos por todo apoio e motivação para a conclusão desta dissertação.

Agradecimentos

A etapa final se aproxima e logo será finalizada. São tantos sentimentos, pensamentos e sensações que me vem à mente, mas um sentimento que pulsa fortemente chama-se gratidão. É meu dever expressá-lo, reconhecendo que a conclusão desta dissertação só foi possível mediante a colaboração de várias pessoas durante o seu processo de desenvolvimento. Destaco os meus sinceros agradecimentos:

A Deus por ter permitido que esse sonho amadurecesse ao ponto de se tornar realidade, cumprindo-se a sua promessa prévia. Incontáveis foram as vezes que em Sua presença busquei forças, abrigo, amparo, entre outras coisas necessárias para superar as adversidades à medida em que surgiam;

Aos meus estimados orientadores Daniel Fernandes Macedo e Marcos Augusto M. Vieira pelas ricas orientações, pela confiança, pelo incentivo, pela paciência, pelas cobranças, pelas sugestões, pelas colaborações, pelo empenho e pelas horas dedicadas;

Ao meu Pai (Vancélio Alves), à minha mãe (Rosely Rodrigues), aos meus irmãos (Alécio Rodrigues e Raissa Rodrigues), por serem valiosíssimos para mim, pelo suporte, pelo incentivo e motivação, pela compreensão nos momentos de ausência, pela tolerância quando o humor esteve alterado, por acreditarem, enfim, por tudo que fizeram e fazem por mim sem medir esforços. Estendo também meus agradecimentos aos demais parentes e pessoas queridas que não foram mencionadas;

À equipe Ethanol na pessoa do Henrique Moura e do Jonas Borges pelo tanto que cresci convivendo com vocês, pela sincera amizade, pela motivação, pelos ensinamentos, pelas ajudas, pela confiança, pelo apoio, por abraçarem comigo a empreitada e, por muitas outras coisas que não cabem nesse texto;

Ao Eduardo Moreira pelos dois anos de convivência compartilhados, pelos ensinamentos, pelas ajudas primordiais durante os passos na pós, por todos os favores;

Ao professor José Marcos pelo suporte fornecido em um dos momentos mais necessários, assim como pelo privilégio de ter participado do *Wireless Networks* (Winet);

Aos estimados amigos Erik de Britto, Pablo Goulart, Luis Cantelli, Julio Soto, Vinícius Fonseca, Vinícius Mota, Jefferson Codeiro e Racyus Delano pelas colaborações,

pelos conselhos, pelos momentos de descontração e pelos ricos aprendizados. Estendo também meus agradecimentos aos demais colegas do Winet;

A todos que oraram e torceram por mim;

À toda equipe do PPGCC do DCC;

À Maria José pela preocupação, pela torcida e pela confiança ao fornecer a estadia de sua casa para a minha permanência;

À instituição de apoio e fomento à pesquisa CNPq pelo agraciamento da bolsa de mestrado oriunda do Prêmio Destaque na Iniciação Científica e Tecnológica (ICT) do ano de 2013, processo 133982/2015-7.

Finalmente, agradeço a todos aqueles que não foram mencionados, mas que também colaboraram diretamente e indiretamente para a conclusão desta. Os meus votos são para que do Céu venham as recompensas em vossas vidas. Deus os abençoe!

“ Just Keep Swimming...”
(Finding Nemo)

Resumo

As redes domésticas têm se tornado cada vez maiores devido ao crescente número de dispositivos inteligentes nelas conectados. A inserção de dispositivos pessoais como *smartphones*, *tablets* e TVs inteligentes resulta na densificação dessas redes. Além disso, serviços como *streaming*, armazenamento de dados em nuvem e *torrents* têm se tornado cada vez mais populares entre usuários domésticos. Logo, surge a necessidade de maior confiabilidade e qualidade de serviços (QoS), o que requer um gerenciamento mais eficiente das redes domésticas. Usualmente, o próprio usuário é o responsável por configurar a sua rede. Porém, geralmente quando ocorre um problema em redes domésticas, os usuários são incapazes de inspecioná-lo ou solucioná-lo, principalmente devido à falta de ferramental de apoio à identificação dos motivos que geraram o problema e à sua solução automática.

Nesta dissertação propomos o *HomeNetRescue* (HNR), um serviço que emprega o conceito de redes definidas por *software* para realizar o gerenciamento autônomo de redes domésticas. O *HomeNetRescue* provê apoio à detecção, diagnóstico e solução automáticas de problemas em redes com e sem fio. Sua arquitetura é genérica e modular, o que permite a adição de novas aplicações e dispositivos a serem monitorados. O serviço foi modelado para ser empregado por provedoras de acesso à Internet, de forma que estas gerenciem problemas à distância nas redes domésticas de seus clientes. Ainda, o serviço pode agregar a essas redes funcionalidades de detecção e solução automática de problemas visando que elas se tornem mais estáveis e confiáveis. Isso pode proporcionar redução de custos às provedoras, da demanda por serviços de suporte e do tempo de recuperação em caso de problemas.

Além de propor o *HomeNetRescue*, esta dissertação apresenta uma avaliação do protótipo do serviço em um ambiente real, com base em três cenários. Os dois primeiros cenários correspondem a duas abordagens para a coordenação de potência de transmissão de pontos de acesso: uma distribuída e outra centralizada. No terceiro cenário descrevemos uma abordagem para a seleção dinâmica e coordenada de canais de pontos de acesso em uma rede doméstica. Nessa abordagem o HNR realizou uma

atribuição orquestrada de canais em pontos de acesso sem fio em um ambiente com múltiplos pontos de acesso.

No primeiro cenário, *Controle de Potência de Transmissão sem Coordenação*, o serviço foi capaz de detectar um aumento na perda de pacotes devido a colisões, atuar na rede automaticamente, sendo capaz de mitigar enlaces sem fio ruins. Isso resultou em um ganho de 7% na taxa média de transferência (vazão) da estação analisada. No segundo cenário, *Controle de Potência de Transmissão com Coordenação*, o serviço melhorou a vazão em 66%, reduziu o atraso em 36% e o *jitter* em 15% quando comparado a uma rede sem o controle de potência. Por fim, no terceiro cenário, *Seleção Dinâmica e Coordenada de Canais*, o serviço realizou a atribuição orquestrada de canais aos pontos de acesso da rede em um ambiente com múltiplos pontos de acesso sem fio. Nesse caso, o *HomeNetRescue* melhorou a taxa média de transferência em 131% e reduziu em 46% e 24% o atraso e o *jitter*, respectivamente, quando comparado a algoritmos de atribuição de canais de pontos de acesso comerciais.

A partir da avaliação do protótipo por meio dos três cenários supracitados, concluímos que o *HomeNetRescue* proporcionou benefícios individuais (em cada estação) e globais (somando os valores das métricas das estações utilizadas no experimento) na rede. Esses benefícios correspondem a ganhos na vazão e reduções nos atrasos e no *jitter* das transmissões sem fio das estações.

Palavras-chave: Redes Definidas por Software, Gerenciamento, Redes Domésticas, Solução de Problemas.

Abstract

Home networks have become larger due to the increasing number of smart devices connected to these networks. In fact, the insertion of personal devices such as smartphones, tablets, and smart TVs result in the network densification. In addition, modern services such as *streaming*, cloud data storage, and *torrents* have become popular among home users. Therefore, the need for both higher reliability and Quality of Service (QoS) arises, which requires a more efficient management of home networks. The home user is usually responsible for setting up his/her network. However, whenever a problem affects a home network, its users are often unable to inspect or troubleshoot the problem, mostly due to the lack of tools aimed at supporting the identification of causes and the automatic solution for the problem.

In this dissertation, we propose *HomeNetRescue* (HNR), a software defined networking service for home network autonomous management. *HomeNetRescue* aims at detecting, diagnosing, and automatically solving problems in both wired and wireless networks. The HomeNetRescue's architecture is generic and modular, which allows adding new applications and devices for monitoring. Internet Service Providers (ISP) could employ our service for remotely managing problems that affect the home networks of their customers. In addition, our service allows adding automatic detection and troubleshooting capabilities to these networks to make them more stable and reliable. Using *HomeNetRescue* supports saving costs for the ISPs, reducing the demands for support services, and shortening the network recovery time in case of failure.

Besides proposing *HomeNetRescue*, this dissertation evaluates a prototype of the proposed service in a real environment, based on three scenarios. The first two scenarios correspond to two approaches for controlling the transmission power of access points: one distributed and other centralized approach. In the third scenario we describe an approach for dynamically and coordinately selecting access point channels in a home network. In this approach the HNR performed an orchestrated allocation of channels on wireless access points in an environment with multiple of them.

In the first scenario, *Uncontrolled Transmit Power Control*, the service detects

an increasing packet loss due to collisions, automatically acts on the network which increases the transmission power, and mitigates bad wireless links. The service results in a gain of 7% in the average throughput of the analyzed workstation. In the second use scenario, *Transmit Power Control with Coordination*, the service improves the average throughput by 66%, reduces the delay by 36% and the jitter by 15% when compared to a network without power control. In the third scenario, *Dynamic and Coordination Selection of the Wireless Channel*, the service performs an orchestrated assignment of channels to network access points in a multi-AP environment. The service then improves the average throughput by 131%, and reduces delay and jitter by 46% and 24%, respectively, when compared to algorithms for channel assignment of commodity access points.

By relying on the prototype's evaluation results obtained through the three aforementioned scenarios, we conclude that *HomeNetRescue* provides both individual benefits, i.e., when considering each station, and global benefits (i.e., when grouping the metric values of all stations used in the experiment) for the home network. These benefits correspond to gains in throughput and reductions in both delays and jitter of the stations' wireless transmissions.

Keywords: Software Defined Networking, Management, Home Networks, Troubleshooting.

Lista de Figuras

Lista de Tabelas

Sumário

Capítulo 1

Introdução

Conforme o relatório *Cisco Visual Networking Index* de 2015, estima-se que a quantidade de dispositivos conectados à Internet será três vezes maior do que a população mundial até 2019 [CISCO, 2015b]. Isto indica um crescimento no número de dispositivos que deverão compor as redes de computadores. Sob esta perspectiva, destaca-se, por sua grande utilização pelos usuários nestas redes, uma miríade de dispositivos inteligentes pessoais como *smartphones, tablets, smart TVs*, *notebooks* e dispositivos de Internet das Coisas (IoT) [Perera et al., 2014].

Paralelamente, cresce também o interesse e a demanda por redes domésticas, as quais contêm diversos dos dispositivos supracitados [Lee et al., 2015]. Em suma, as redes domésticas estão se densificando e tornando-se mais complexas de gerenciar. Segundo Tanenbaum & Wetherall [2013], a arquitetura da Internet é representada conforme a Figura 1.1. Nela, são representados diversos tipos de redes, como Provedores de Internet (*Internet Service Providers*- ISP) e *Backbones*[1], Pontos de Troca de Tráfego (PTTs) entre redes e *Backhauls*[2]. Além disso, a figura ilustra também as redes domésticas conectadas a uma das bordas da rede, a sua organização e a organização da Internet.

O volume elevado de conexões móveis, o aumento no tráfego nas redes domésticas, cabeadas ou sem fio, bem como a variedade de problemas ou falhas que nelas podem ocorrer correspondem a outros fatores que contribuem para o aumento da complexidade de seu gerenciamento [Yiakoumis et al., 2011]. Nesta dissertação, problemas podem ser caracterizados como aspectos de baixo desempenho, falhas, ataques à rede ou eventos que possam ser representados por uma condição anômala que tenha uma ou mais

[1]Espinha dorsal de uma rede onde são designadas as ligações centrais de desempenho elevado (núcleo da rede).

[2]Porção de uma rede hierárquica de telecomunicações responsável por fazer a ligação entre o núcleo da rede (ou backbone) e as sub-redes periféricas (por exemplo, redes domésticas ou empresariais).

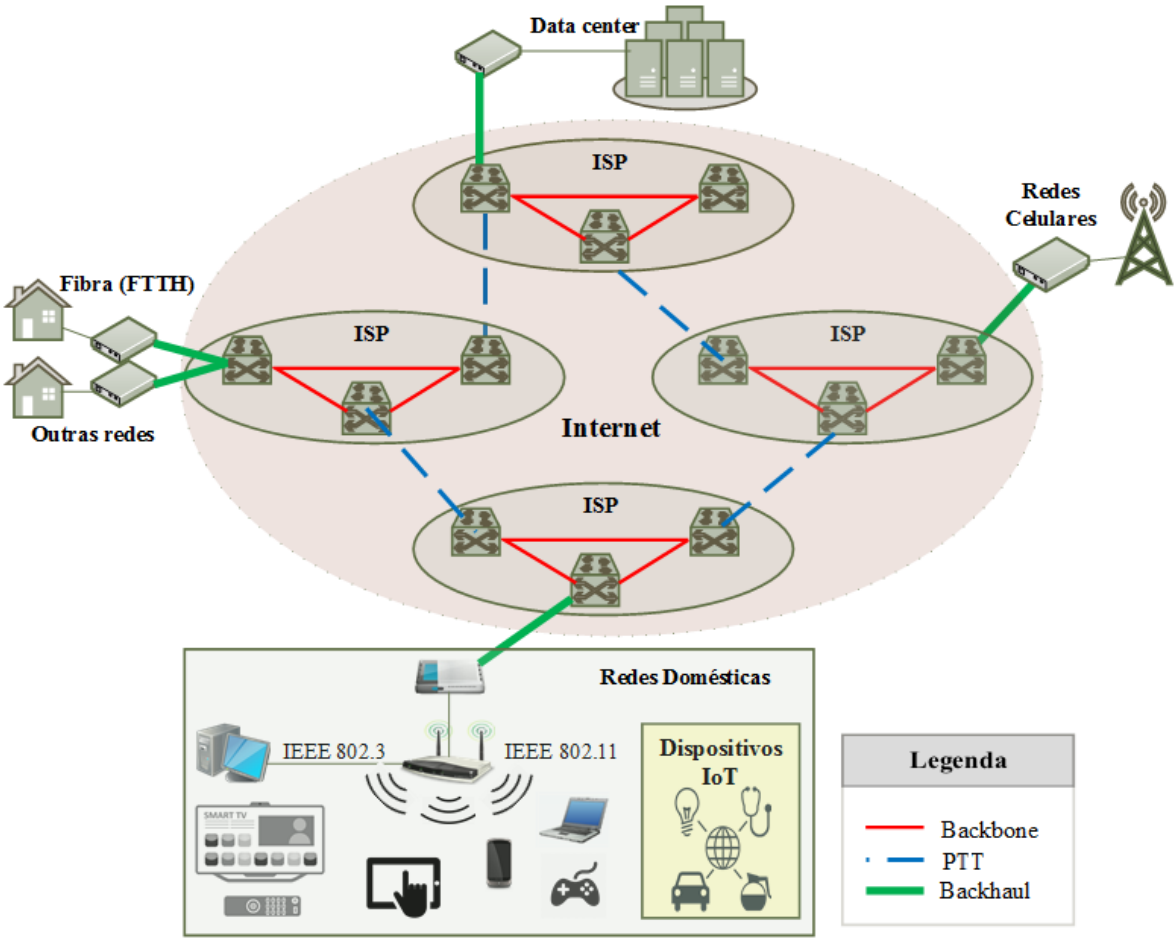


Figura 1.1: Arquitetura da Internet com enfoque em redes domésticas (adaptado de Tanenbaum & Wetherall [2013])

métricas de rede associadas. Geralmente, quando ocorre um problema nessas redes, seus usuários não são capazes de inspecioná-lo ou solucioná-lo, principalmente devido à falta de ferramental de apoio à identificação precisa dos motivos que geraram o problema e à solução automática deste [Fratczak et al., 2013]. Contudo, tais usuários são em geral os responsáveis por configurá-las [Dong & Dulay, 2011].

O diagnóstico e a solução automática de problemas em redes domésticas são benéficos tanto para os usuários, quanto para as provedoras de acesso. Para os usuários, os benefícios são uma rede mais estável e confiável, aumentando a sua satisfação. Para as provedoras, a automatização permite uma redução no seu custo operacional, pois exige uma demanda menor por serviços de manutenção. Além disso, ferramentas que facilitam o diagnóstico da causa do problema implica em um menor tempo para a sua recuperação, aumentando a produtividade da equipe de operação. Sob o ponto de vista de regulação, uma rede mais robusta é importante para alcançar as metas de qualidade de rede e o grau de disponibilidade que são definidos pelos órgãos reguladores, que

podem aplicar multas em caso de descumprimento dessas metas.

Nesta dissertação, propomos o *HomeNetRescue* (HNR), um serviço para o gerenciamento autônomo de redes domésticas voltado à detecção, ao diagnóstico e à solução automática ou minimização de problemas, os quais também pode atuar em aspectos de desempenho, baseado em redes definidas por *software* (ou do Inglês, *software defined networking* - SDN). SDN consiste em uma abordagem para a organização de redes de computadores onde as funções dos planos de controle e dados foram separadas. Através dela, torna-se possível melhorar o gerenciamento da rede, a qualidade de experiência dos usuários e a programação das redes através de aplicações executadas em controladores, dentre outros benefícios [McKeown et al., 2008; Feamster et al., 2014; Guedes et al., 2012b; Macedo et al., 2015]. Isto permite ao HNR atuar onde as provedoras atualmente não atuam, viabilizando um novo modo inteligente de gerenciamento de redes domésticas para a detecção e a solução de problemas.

1.1 Motivação

Nossa principal motivação para a proposição do HNR encontra-se na necessidade emergente de um serviço que apoie o gerenciamento de redes domésticas, a fim de lidar com os problemas que nelas ocorrem e investigar formas de resolvê-los automaticamente. Nessas redes, existem várias dinâmicas comportamentais e estruturais que agravam o seu gerenciamento [Kim & Feamster, 2013], o que sugere uma variedade de desafios abertos para estudo e proposição de soluções.

Segundo outro relatório da Cisco (*Cisco IP Next-Generation Network* - NGN), estima-se que o tráfego sem fio de dispositivos móveis ultrapasse o tráfego de dispositivos cabeados até 2019 [CISCO, 2015a]. O mesmo relatório informa que, em 2014, os dispositivos cabeados correspondiam a somente 54% do tráfego total. Acompanhando o crescimento do tráfego sem fio, cresce também a quantidade de problemas em redes domésticas que utilizam o ar como meio de comunicação. De fato, tais redes são mais suscetíveis a problemas, pois a propagação de sinais no meio sem fio pode sofrer interferências, sombreamento, atenuação, reflexão e desvanecimento, dentre outros fenômenos os quais não ocorrem com a mesma intensidade em uma comunicação cabeada.

Em suma, é importante e necessário gerenciar as redes domésticas, sobretudo os dispositivos de comunicação sem fio. Por exemplo, no contexto de uma provedora, realizar a detecção e a solução de problemas pode contribuir para a redução de custos evitando alocar técnicos para resolver problemas que podem ser resolvidos à distância e automaticamente. Outros benefícios de detecção e solução automática de problemas

são, por exemplo: viabilizar melhorias no nível de qualidade de experiência (QoE) dos usuários e possibilitar às provedoras a identificação de problemas em suas redes, identificando suas causas.

1.2 Objetivos

O objetivo desta dissertação consiste em propor, desenvolver e avaliar o *HomeNetRescue* (HNR). O *HomeNetRescue*, por sua vez, corresponde a um serviço SDN para o gerenciamento autônomo de redes domésticas voltado para a detecção, o diagnóstico e a solução automática ou minimização de problemas. Em outras palavras, o HNR é um serviço de rede de computadores capaz de atuar em um escopo iniciado no *backhaul* das provedoras de serviços de telecomunicação, a partir dos *gateways* domésticos, indo até as estações dos usuários finais (vide a Figura 1.1).

O *HomeNetRescue* tem como intuito apoiar o refinamento dos serviços prestados pelas provedoras, permitindo também o gerenciamento das redes domésticas além dos seus *gateways* de modo a viabilizar melhorias na QoE dos usuários. Isso é feito mediante a detecção, o diagnóstico e a solução de problemas nessas redes. Um diferencial do HNR encontra-se no escopo de sua atuação, pois ele pode fornecer recursos para as provedoras atuarem de forma mais abrangente do que é feito atualmente, representando assim um novo modo inteligente de gerenciamento de redes domésticas para a detecção e a solução de problemas.

Dessa forma, objetivamos:

- Propor a arquitetura do *HomeNetRescue*;
- Implementar um protótipo do *HomeNetRescue*;
- Possibilitar a detecção, o diagnóstico e a solução de problemas em redes domésticas cabeadas e sem fio;
- Viabilizar a redução de custos para as provedoras através do utilização do HNR; e,
- Viabilizar melhorias na qualidade de experiência dos usuários.

1.3 Contribuições

Dentre as principais contribuições desta dissertação, destacam-se:

- A modelagem do *HomeNetRescue*, com base em redes definidas por *software*, para apoiar o gerenciamento de redes domésticas cabeadas e sem fio. Seu principal objetivo é detectar e solucionar automaticamente problemas nessas redes;
- Avaliações do protótipo do *HomeNetRescue* em ambiente real;
- Proposição e avaliação de dois cenários para o controle de potência de transmissão (um centralizado e um distribuído). O cenário para controle de potência centralizado proporcionou melhoria da vazão em 66%, diminuindo o atraso em 36% e o *jitter* em 15%, quando comparado a uma rede sem controle de potência de transmissão; e
- Proposição e avaliação de um cenário para a atribuição orquestrada de canais em pontos de acesso sem fio em um ambiente com múltiplos pontos de acesso sem fio. Nesse, o *HomeNetRescue* proporcionou uma melhoria de 131% na vazão, diminuindo o atraso e o *jitter* em 47% e 24%, respectivamente, em relação aos algoritmos de atribuição de canais de APs comerciais.

Além das contribuições previamente citadas também publicamos dois artigos em duas conferências, nacional e internacional, respectivamente. O primeiro artigo intitulado "Um Serviço SDN para Detecção e Solução de Problemas em Redes Domésticas" foi publicado no $XXII$ Workshop de Gerência e Operação de Redes e Serviços (WGRS) do $XXXV$ Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC 2017) [Alves et al., 2017]. O segundo artigo intitulado "*HomeNetRescue: An SDN Service for Troubleshooting Home Networks*" foi publicado no *IEEE/IFIP 16th Network Operations and Management Symposium (NOMS 2018)* [Alves et al., 2018].

1.4 Organização da Dissertação

O restante da dissertação está organizado respeitando a seguinte estrutura. No Capítulo 2 discutimos os conceitos e os trabalhos relacionados a esta dissertação. No Capítulo 3 discutimos sobre os resultados de uma pesquisa de opinião realizada com diversos usuários de redes domésticas por meio de um questionário, a fim de elicitar os principais problemas encontrados por tais usuários em redes domésticas em seu cotidiano. No Capítulo 4 descrevemos o serviço *HomeNetRescue* apresentando a sua arquitetura. No Capítulo 5 detalhamos a implementação do HNR. No Capítulo 6 descrevemos e discutimos os resultados de nossos experimentos realizados a fim de avaliar o HNR em ambientes reais. Por fim, no Capítulo 7 concluímos esta dissertação com as sugestões de trabalhos futuros.

Capítulo 2

Conceitos e Trabalhos Relacionados

Apresentamos neste capítulo os principais conceitos para a compreensão desta dissertação. Primeiramente, conceituamos: redes domésticas, tanto cabeadas quanto sem fio; problemas em redes de computadores; e redes definidas por *software*. Posteriormente, descrevemos os trabalhos relacionados à nossa pesquisa. Apresentamos uma tabela comparativa que destaca as principais semelhanças e diferenças entre o *HomeNetRescue* e outras pesquisas existentes na literatura, permitindo posicionar o nosso trabalho em relação ao estado da arte da área.

Organizamos este capítulo da seguinte maneira. Nas Seções 2.1, 2.2 e 2.3 apresentamos os conceitos de rede doméstica, problemas em redes de computadores e redes definidas por *software*, respectivamente. Essas tecnologias são adotadas para a implementação do *HomeNetRescue*. Na Seção 2.4 descrevemos a pesquisa bibliográfica realizada. Por fim, na Seção 2.5 concluímos o capítulo.

2.1 Redes Domésticas

Rede Doméstica (*Home Area Network* - HAN) corresponde a um tipo de rede de computadores implantada em espaços físicos limitados como casas, apartamentos, pequenos escritórios ou similares. A sua função consiste em permitir a comunicação e o compartilhamento de recursos de rede (como Internet, impressora, etc.) entre dois ou mais dispositivos domésticos na rede [Lee et al., 2015]. Segundo Tanenbaum & Wetherall [2013], nas redes domésticas os dispositivos devem ser de fácil instalação e financeiramente acessíveis para os usuários, permitindo a expansão gradual da rede (escalabilidade) e que a mesma seja segura e confiável. Em geral, as redes domésticas não possuem administradores de rede e são os próprios usuários os responsáveis por configurá-las [Dong & Dulay, 2011].

As redes domésticas são evidenciadas por estarem presentes no cotidiano de uma fração representativa da população mundial [Calvert et al., 2011]. Uma tendência nessas redes tem sido o crescimento da quantidade e da variedade de seus dispositivos. Entre os dispositivos utilizados nas redes domésticas, podemos citar: APs, roteadores, comutadores, estações (consoles de jogos, *smart TVs*, *smartphones*, *tablets* e *notebooks*), dispositivos IoT, entre outros Estes utilizam a infraestrutura das redes domésticas para acessarem a Internet, conforme ilustrado na Figura 2.1.

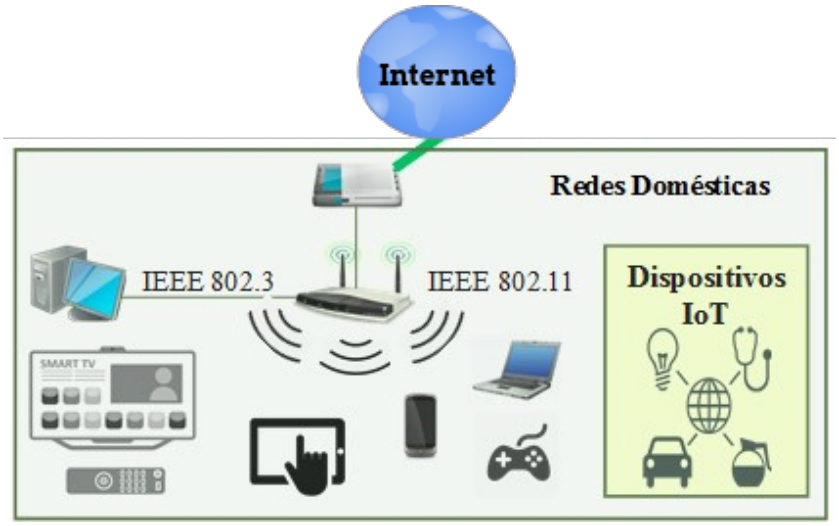


Figura 2.1: Variedade de dispositivos em uma rede doméstica.

Outra característica do ambiente de redes domésticas é a existência de outros tipos de rede. Esses tipos de rede compõem as redes domésticas possibilitando a comunicação entre os dispositivos e se diferenciam pelos meios de comunicação utilizados. Alguns exemplos são: *WiFi/Wireless* (IEEE 802.11), Ethernet (IEEE 802.3), *Power Line Communication* (PLC), Bluetooth (IEEE 802.15), entre outros [Mortier et al., 2012; Bouchet et al., 2014; Lee et al., 2015]. Nesta dissertação, restringimos nosso interesse ao gerenciamento de redes domésticas para a detecção, o diagnóstico e solução de problemas em redes WiFi e Ethernet, devido ao fato de que os outros tipos de rede não utilizam a pilha TCP/IP de forma que a solução seria diferente para redes WiFi e redes IoT, por exemplo. A seguir descrevemos esses dois tipos de rede, enquanto, na Seção 4.1, contextualizamos o uso do *HomeNetRescue* em redes domésticas.

2.1.1 Redes WiFi – IEEE 802.11

O conceito de rede local sem fio (*Wireless Local Area Network* - WLAN) ou rede WiFi[1] corresponde a um tipo de rede amplamente adotado em ambientes residenciais,

[1]Marca registrada pela WiFi Alliance, a abreviação deriva do termo *Wireless Fidelity*.

coorporativas, educacionais e comerciais [Adya et al., 2004]. Este tipo de rede destaca-se dos demais por seu modo de comunicação, possibilitando o acesso sem fio, por radiofrequência, aos recursos da rede. Esse tipo de rede é especificado pelo padrão IEEE 802.11 [IEEE-802.11, 2017].

O padrão IEEE 802.11 é composto por um conjunto de outros padrões que, juntos, formam a família IEEE 802.11. Alguns componentes dessa família são os padrões: IEEE 802.11k, IEEE 802.11f, IEEE 802.11ac, entre outros [IEEE-802.11, 2017]. A partir desses padrões utilizamos os recursos necessários para implementar as funcionalidades do *HomeNetRescue* para lidar com redes locais sem fio. Para acessar tais recursos realizamos implementações com chamadas em espaço de usuário, isso com o auxílio de ferramentas do Linux como *iw*, *iwconfig* e afins, pela facilidade de programação, mas em uma versão de produção do *HomeNetRescue* pode-se realizar chamadas de sistema (espaço de kernel) visando maior eficiência.

2.1.2 Redes Ethernet – IEEE 802.3

Ethernet consiste em outro tipo de rede encontrada em redes domésticas, além das redes WiFi. Este tipo de rede é definido pelo padrão IEEE 802.3 [IEEE-802.3, 2017]. A redes Ethernet são compostas por diversos subpadrões que especificam a camada física, incluindo a subcamada MAC (*Media Access Control*) do modelo de referência OSI (*Open Systems Interconnect*) [Tanenbaum & Wetherall, 2013]. No padrão IEEE 802.3 são definidos o modo de interconexão para redes locais (*Local Area Networks* - LAN) e o protocolo de enlace físico entre dispositivos (hospedeiros, concentradores e comutadores) [Tanenbaum & Wetherall, 2013].

Tanenbaum & Wetherall [2013] categorizam as redes Ethernet em *classic* e *switched*. Na primeira categoria são definidas taxas de transmissão de dados variando entre 3 e 10 *Mbps*, bem como a função do componente que interliga os dispositivos na rede, denominado concentrador ou *hub*. Já na segunda categoria (mais utilizada atualmente) são alcançadas velocidades de 100 Mbps a 40 Gbps (*Fast/Gigabit Ethernet*) e os componentes de interligação são chamados de comutadores. Vale ressaltar que as taxas de transmissão em Ethernet são alcançadas de acordo com o meio físico no qual os dados são transmitidos, como: fibra ótica, cabos de par trançado, entre outros.

2.2 Problemas em Redes de Computadores

Problemas são recorrentes em sistemas de comunicação e significativamente impactam o desempenho de redes de computadores [Steinder & Sethi, 2004]. Em nossa abordagem

conceituamos problemas relacionando-os às anormalidades que podem ocorrer em uma rede, por exemplo: aspectos de baixo desempenho, ataques à rede ou eventos que possam ser representados por uma condição anômala que tenha uma ou mais métricas de rede associadas. Ainda, consideramos faltas, falhas, erros e sintomas como um subconjunto de problemas. A seguir apresentamos os referidos conceitos nos baseando em Avižienis et al. [2004] e Fonseca & Mota [2017]:

- *Falta* é causa raiz que pode levar um sistema a um estado de erro, podendo esta, permanecer em estado de dormência antes de causar um erro;
- *Erro* é a manifestação de uma falta, motivando um sistema a entrar em um estado incorreto para o qual não foi programado. Um erro pode ou não causar interrupções em um serviço;
- *Falha* corresponde ao evento de entregar um serviço desviando-se da sua proposta especificada originalmente (comportamento correto esperado) devido a um ou mais erros; e,
- *Sintoma* é um efeito colateral causado por uma ou mais falhas. Os sintomas podem ser observados no comportamento da rede, ou seja, é a parte perceptível após uma falha.

Adicionalmente, problemas em redes de computadores podem ser categorizados em diversas formas: por tempo de duração (permanente, intermitente e transiente) [Steinder & Sethi, 2004], criticidade, contexto (*hardware* ou *software*), tipo de rede (WiFi ou Ethernet), problemas de conectividade, desempenho, segurança, autenticação de rede, entre outros.

2.2.1 Problemas em Redes Domésticas

Problemas também são recorrentes em redes domésticas [Calvert et al., 2011]. São vários os tipos possíveis nestas redes, como por exemplo, configurações incorretas de *modems* ou roteadores, consumo excessivo de banda por usuários de uma mesma rede, perda de pacotes e atenuação de sinal. Também podemos citar: anomalias no funcionamento do *software* do cliente, indisponibilidade de serviços de rede (como DNS, HTTP, DHCP, etc.), interferências no sinal de radiofrequência pelo uso do espectro não licenciado e degradação de desempenho.

Na Tabela 2.1 listamos de forma não exaustiva outros problemas que podem ser encontrados em redes domésticas. Dentre as consequências desses problemas nas

redes estão: significativos índices de insatisfação por parte de clientes, prejuízos a provedoras de acesso à Internet, entre outros. Neste sentido, dentre os problemas que podem ocorrer em uma rede doméstica, objetivamos detectar e solucionar alguns com o auxílio do *HomeNetRescue*, sendo estes descritos na Seção 5.5.

Tabela 2.1: Lista de problemas em redes de computadores e em redes domésticas.

#	**Problema**	**Referência**
1	Configuração incorreta na estação do usuário	Kim et al. [2014b]
2	Problema no enlace com o roteador	
3	Mau comportamento do roteador local	
4	Interrupção no ISP	
5	Falha no enlace entre ISP e a Internet	
6	Interrupção de rede pelo provedor de serviços remotos	
7	Falha no servidor remoto (Servidor parado)	
8	Provedor de serviço bloqueando o ISP	
9	Servidor bloqueando o ISP	
10	Provedor de serviços bloqueando IP externo	
11	Servidor bloqueando endereço IP	
12	Cabo Ethernet desconectado	
13	Adaptador de rede desabilitado	
14	Conflito de endereço IP	
15	Endereço do *Gateway* incorreto	
16	Configuração incorreta do endereço do DNS	
17	Queda de servidor (Web ou SSH)	
18	*Network Address Translation* (NAT) bloqueando o servidor	
19	ISP bloqueando o servidor	
20	Porta bloqueada pelo NAT	
21	Porta bloqueada pelo ISP	
22	Lentidão no servidor web	
23	Cliente desconectado	
24	Falta de endereço IP	Yishan et al. [2009]
25	Sinal sem fio transformado	
26	Distúrbio de ruídos no sinal sem fio	
27	Rede desconectada	
28	Lentidão após instalação de uma interface de rede (NIC)	
29	NIC funcionando mas incapaz de conectar	
30	Capacidade insuficiente	
31	Perda de pacotes	
32	Problema de configuração da bridge	
33	*Software* cliente não funcionando	
34	Falha no *Routing Information Protocol* (RIP)	
35	Falha no *Open Shortest Path First* (OSPF)	
36	Falha no *Border Gateway Protocol* (BGP)	
37	Falha de *hardware*	
38	Falha no *Transmission Control Protocol* (TCP)	
39	Falha no protocolo *Internet Protocol (IP)*	
40	Atenuação de sinal sem fio	

2.3 Redes Definidas por *Software*

Rede definida por *software* ou *Software Defined Network* (SDN) corresponde a uma relevante concepção para as redes de computadores que vem sendo amplamente utilizada tanto na comunidade acadêmica quanto na indústria [McKeown et al., 2008; Guedes et al., 2012a]. As SDNs destacam-se devido à estagnação da atual infraestrutura das redes de computadores (Internet), em relação à proposição de novos protocolos e tecno-

logias. Este fenômeno, conhecido como "calcificação da rede", ocorre devido aos baixos níveis de flexibilidade e estabilidade atualmente alcançados na Internet [Guedes et al., 2012a].

Uma nova perspectiva para modelagem de redes é apresentada em McKeown et al. [2008]. Nesse sentido, SDN oferece três características principais aos administradores de rede: gerenciamento centralizado, flexibilidade e programabilidade. Além disso, SDN propõe: *a*) desacoplamento de funções de rede via separação dos planos de controle e dados; *b*) fluxo como base para decisões de encaminhamento de pacotes; *c*) atribuição do controle a uma entidade externa logicamente centralizada na rede (controlador); e *d*) programabilidade de redes através de aplicações de *software* executadas em controladores [Feamster et al., 2014].

Kreutz et al. [2015] também discutem as vantagens dessa tecnologia, como obtenção de uma visão global do estado da rede mediante a implementação inteligente de controladores. Com isso, é possível obter uma identificação mais precisa dos problemas nas redes. Este recurso, em especial, corresponde ao núcleo da solução para o gerenciamento das redes domésticas empregado nessa dissertação. Outras vantagens apresentadas com SDN, as quais não eram providas nos outros modelos de rede anteriores, são: adição de maiores recursos de programação à rede; comutação inteligente de pacotes, permitindo que estes sejam controlados por *software* independente de *hardware*; programar redes através de aplicações externas; viabilizar maior controle de rede aos seus administradores, entre outros [Guedes et al., 2012a; Nadeau & Gray, 2013].

2.3.1 Arquitetura de uma Rede Definida por *Software*

Na Figura 2.2 apresentamos a arquitetura de uma rede definida por *software*. Tal arquitetura é composta por planos distintos, a saber: aplicação, controle e dados. O acesso aos referidos planos se dá através de *Application Programming Interfaces* (APIs) abertas que podem ser invocadas para a execução das respectivas funções em cada plano. A seguir descrevemos as características dos planos mencionados.

O plano de aplicação concentra as aplicações que utilizam os princípios SDN. As aplicações desse plano especificam os recursos e os comportamentos requeridos da rede onde estão em execução. Em geral tais aplicações são executadas em um controlador de rede, componente esse localizado no plano de controle, descrito a seguir.

O plano de controle é o responsável por controlar os recursos gerenciáveis da rede. Isto é feito através de um controlador de rede, de forma lógica e centralizada, sendo ele capaz de obter o estado de rede e de traduzir os requisitos das aplicações para os componentes do plano de dados. O controlador corresponde a um conjunto

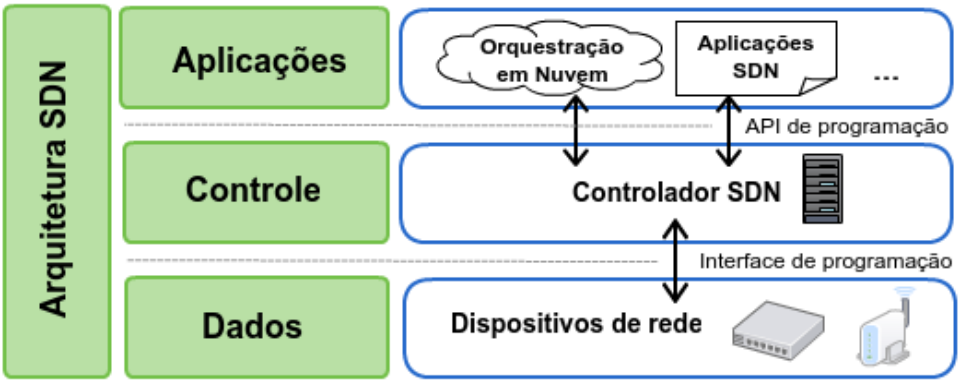


Figura 2.2: Arquitetura de uma Rede Definida por *Software*.

de programas que facilita as atividades das aplicações. Ainda, é representado como um sistema operacional de rede de computadores devido às suas funcionalidades [Lee et al., 2015]. É também um *software* configurável onde são implementadas aplicações com as instruções lógicas para os fluxos e as operações a serem executadas na rede [Nadeau & Gray, 2013]. O controlador é caracterizado por possuir maior capacidade de armazenamento e processamento, sendo capaz de processar regras mais complexas. Com isso, na arquitetura SDN, o controlador determina a forma de encaminhamento de cada fluxo e os componentes do plano de dados os encaminham.

A partir dos enlaces dos dispositivos existentes na rede, com o controlador e com os protocolos (Openflow e Ethanol, descritos a seguir) torna-se possível criar a topologia de rede mencionada anteriormente e assim obter a sua visão global. O controlador é o responsável pela inteligência da rede. Existem diversos controladores de redes SDN, proprietários ou *open source* e desenvolvidos em várias linguagens de programação. Exemplos de controladores SDN são: POX, NOX, RYU, Cisco Open SDN Controller (OSC), Ericsson SDN Controller, Beacon, Floodlight, entre outros [Kreutz et al., 2015; ONF, 2017].

O plano de dados é composto pelos componentes gerenciáveis da rede e estes são os responsáveis por realizar o encaminhamento de pacotes pela rede. Devido à capacidade limitada dos componentes desse plano, exceto das estações (*smartphones*, *notebooks*, entre outras), geralmente, os roteadores e pontos de acesso processam regras mais simples, como encaminhamento de pacotes para uma porta de saída, respondem por requisições de métricas solicitadas pelo controlador, entre outras funções. A seguir apresentamos os dois protocolos que podem ser utilizados nos componentes do plano de dados.

2.3.1.1 OpenFlow

O protocolo OpenFlow, proposto por um grupo de pesquisadores da Universidade de Stanford e atualmente mantido pela *Open Networking Foundation*(ONF), é o principal, mas não único protocolo SDN [ONF, 2017; OpenFlow, 2017]. Dos estudos do protocolo originou-se o conceito de SDN [McKeown et al., 2008]. OpenFlow consiste em um protocolo aberto para programação de fluxos em comutadores capaz de aproveitar os recursos dos dispositivos de rede e encontra-se na versão 1.5.1.

A principal função do protocolo em SDN consiste em viabilizar a comunicação entre o controlador e os componentes de comutação de uma rede. Deste modo, o Openflow é composto por um conjunto de protocolos e API que viabilizam tal comunicação. Vale ressaltar que através do protocolo a tabela de fluxos dos dispositivos de comutação pode ser programada, e com isso, é permitido determinar a ação a ser realizada para cada fluxo. Este protocolo representa uma das mais comuns interfaces *southbound*[2] em SDN [Kim & Feamster, 2013]. Em nossa abordagem a comunicação entre o controlador e os componentes da rede ocorre através dos protocolos OpenFlow e Ethanol, descrevemos o último a seguir.

2.3.1.2 Ethanol

O Ethanol consiste em uma arquitetura SDN para o gerenciamento de redes IEEE 802.11 (WLAN) empresariais e domésticas [Moura et al., 2015a,b]. Na arquitetura, foi definida uma interface *southbound* que permite o controle de roteadores IEEE 802.11, bem como de estações sem fio que implementem padrões IEEE 802.11. É uma arquitetura que possui dois tipos de componentes para o seu funcionamento, sendo eles: controlador Ethanol, executado em um dispositivo local da rede ou virtualizado na nuvem, onde são definidas as regras de gerenciamento; e roteadores Ethanol, roteadores IEEE 802.11 modificados que possibilitam que as aplicações possam lidar com os parâmetros de redes sem fio.

Na arquitetura Ethanol os protocolos OpenFlow e Ethanol podem ser utilizados para o controle da rede. O primeiro protocolo lida com os dispositivos cabeados, enquanto o segundo, descrito no trabalho de Moura et al. [2015a], lida com os dispositivos sem fio (APs e estações). Para tal, no protocolo Ethanol são utilizadas conexões seguras (Socket SSL) para a comunicação do controlador com os APs e com as estações. Através destas características, a arquitetura fornece recursos para controle de *handoff* de estações entre os roteadores, controle de autenticação de usuários, criação de redes virtuais, configuração de QoS, localização de usuários na rede, entre outras.

[2]interface entre o controlador e os comutadores programáveis

Na Seção 5.3 apresentamos detalhes adicionais sobre a arquitetura em conjunto com o *HomeNetRescue*.

2.4 Trabalhos relacionados

A literatura apresenta diversas ferramentas e plataformas para o diagnóstico e a solução de problemas [Fonseca & Mota, 2017; Biswas et al., 2015; Kim et al., 2014b; DiCioccio et al., 2012]. Tais ferramentas diferem nos tipos de redes e equipamentos que suportam, no escopo de problemas tratados, nas abordagens (distribuídas ou centralizadas), se somente detectam ou também solucionam problemas, entre outras características. Na Tabela 2.2 apresentamos uma comparação entre o *HomeNetRescue* e outras propostas da literatura. Em sequência, descrevemos em síntese cada trabalho e ao final da seção discutimos a respeito das principais diferenças. Na tabela mencionada as colunas identificam as seguintes propriedades: soluções focadas em redes domésticas (RD); uso de técnicas para solucionar problemas (*troubleshooting* - TS); utilizam SDN (SDN); fornecem recursos para a detecção automática de problemas (DA); suportam redes ethernet (ET); suportam redes *wireless* (WI); identificam problemas nas estações dos usuários (PE); e possibilitam a adição de novas regras de monitoramento e novos tipos de dispositivos, ou seja, extensibilidade (EX). A seguir apresentamos as propostas mais relevantes.

Do You See What I See? (*DYSWIS*) é um *framework* P2P colaborativo, com uma arquitetura centralizada, para a detecção e o diagnóstico de falhas em redes de usuários finais mediante o monitoramento de pacotes e relatórios de falhas de aplicações. As regras para a detecção de falhas são baseadas em consultas e sondagens distribuídas e recorrem, ainda, à colaboração de usuários para distinção de falhas na rede, bem como a identificação do ponto de origem das mesmas. Cada nó DYSWIS possui um conjunto de módulos de sondagem com programas que investigam várias condições de rede e atualizam um repositório, em um servidor da rede, responsável por armazenar as regras de diagnósticos para os problemas já detectados por outros nós da rede. Através do repositório os usuários podem consultar o histórico de falhas de outros nós (outros usuários) e corrigir falhas em sua rede [Kim et al., 2014b].

Pelle et al. [2015] propuseram o *Epoxide*, um *framework* modular baseado no editor *gnu emacs* que fornece recursos para a integração de ferramentas como *ping*, *traceroute*, *tcpdump*, entre outras, no processo de detecção de problemas em redes de computadores. Na abordagem, tais ferramentas podem ser representadas em um grafo e estas correspondem aos nós do grafo. Os nós se interligam por arestas direcionadas

que indicam o fluxo de dados. Ainda, os autores definem uma sintaxe de entrada e saída de dados utilizada para comunicação das ferramentas de verificação da rede. Os dados de cada ferramenta são adicionados no *buffer* do editor de texto e, por meio desse, realiza-se a análise da rede.

Outra abordagem encontra-se em *Home Network Data Recovery* (*HNDR*), onde Calvert et al. [2011] propuseram uma plataforma SDN autônoma de análise de *logs* de eventos de rede para solução de falhas em redes domésticas cabeadas e sem fio. Na abordagem, foram exploradas aplicações de medição de desempenho da Internet (problemas de desempenho), segurança de rede (detecções de intrusões, filtros de *spam*, etc.) e solução de falhas através de *logs*, em conjunto com a autoconfiguração de dispositivos. Foram utilizados o controlador NOX e o protocolo OpenFlow. Além disso, para a coleta dos eventos de rede foram utilizadas as ferramentas *tcpdump* e *iwevent* em um roteador com Linux Noxbox.

O *HomeNet Profiler* é uma aplicação cliente-servidor para a medição de métricas de redes domésticas, executada em estações clientes [DiCioccio et al., 2012]. Na abordagem o usuário deve executar uma aplicação e, posteriormente, enviar os dados coletados a um servidor para análise. A aplicação monitora redes domésticas cabeadas e sem fio. As informações mensuradas incluem configurações de rede, desempenho, vizinhança WiFi, dispositivos ativos, serviços em execução, entre outras, realizadas via protocolos ZeroConf e UPnP (*Universal Plug and Play*). Por permitir que o usuário execute a aplicação e envie os dados coletados a um servidor para análises futuras a aplicação não provê recursos para automaticamente detectar os possíveis problemas da rede.

HomeVisor, proposto por [Fratczak et al., 2013], corresponde à outra ferramenta para o gerenciamento e a configuração de rede doméstica também capaz de identificar e solucionar problemas nesses ambientes. Proposta para ser utilizada pelos ISPs, ela permite realizar o gerenciamento das redes remotamente, como também realizar análises nos comutadores OpenFlow da rede dos usuários. Na modelagem os autores utilizaram o Flowvisor e o protocolo OpenFlow para possibilitar a divisão dos recursos (*slices*) de *hardware* da rede dos ISPs. Nessa abordagem não foram consideradas redes domésticas com dispositivos de comunicação sem fio.

Outra abordagem é apresentada na arquitetura *Meraki* [Biswas et al., 2015], comercializada pela Cisco, para o gerenciamento de redes comerciais através da nuvem. A arquitetura fornece configuração centralizada, monitoramento e ferramentas de resolução de problemas em redes cabeadas ou sem fio. É uma arquitetura *backend* composta de pontos de acesso (APs), comutadores e *firewalls* que monitoram métricas de redes e as repassam a um sistema de gerenciamento central, quando solicitadas. A arquite-

tura auxilia no monitoramento do fluxo de tráfego dos protocolos de rede (ex: ARP, DHCP, DNS, entre outros), dos sistemas operacionais em utilização, do uso de dados de aplicações, dos padrões de uso de rede pelos usuários, da intensidade de força do sinal sem fio (*received signal strength indicator* - RSSI), dos níveis de interferências em sinais sem fio (*signal-to-noise ratio* - SNR) nos APs e dos padrões comportamentais de usuários.

SDN-RADAR consiste em uma abordagem distribuída para o monitoramento da infraestrutura de rede e a localização de problemas ou falhas de desempenho em redes de grande porte (*data centers*) [Gheorghe et al., 2015]. Recursos SDN são utilizados no processo de identificação de enlaces de baixo desempenho em redes cabeadas. A ferramenta auxilia os administradores de rede a entenderem as prováveis falhas de enlaces de rede, assim como a monitorar e depurar as falhas que afetam os serviços de entrega nas redes dos usuários.

Kanuparthy et al. [2012] propuseram o WLAN-Probe, uma ferramenta em nível de usuário para o diagnóstico de problemas de desempenho em redes domésticas sem fio. A ferramenta é capaz de detectar problemas como baixos níveis de SNR, congestionamento e terminais escondidos realizando verificações, em nível de enlace, nos roteadores da rede. Na abordagem, o monitoramento da rede ocorre através de sondagens ativas e passivas. Ainda, foi proposta uma árvore de decisão que utiliza a taxa de transmissão por pacote e o atraso dos mesmos para inferir os problemas de rede mencionados.

Em *Why is my WiFi Slow?* (*WiSlow*), Kim et al. [2014a] propuseram uma ferramenta para o diagnóstico de problemas de desempenho em redes WiFi (abrangendo redes domésticas) através de sondagens em nível de usuário – envio de rajadas de pacotes UDP para os APs. Esta ferramenta auxilia na localização física e na identificação de causas que impactam o desempenho da rede sem fio. Isto é possível através da análise da perda de pacotes e do número de ACKs IEEE 802.11 recebidos na rede. Na abordagem, foram analisadas interferências causadas por dispositivos que não são WiFi, tais como, microondas, *baby monitors* e telefones sem fio. Com a ferramenta, também foi possível identificar a contenção de canal, inferir o tipo de produto causador da interferência e estimar a posição aproximada dos elementos causadores de interferências no sinal de radiofrequência mediante a colaboração de usuários.

Sundaresan et al. [2013] propuseram o *Where's The Fault?* (*WTF?*), uma ferramenta executada em roteadores domésticos utilizada para a detecção de problemas de desempenho em redes domésticas cabeadas e sem fio. Para tal, são utilizadas informações de *timing* e *buffering* de rede, obtidos do monitoramento passivo dos tráfegos dos roteadores. Com essas informações foi possível detectar gargalos em enlaces físicos e

em redes sem fio, bem como detectar se a demanda pela rede não era suficiente para saturar os respectivos enlaces, por exemplo, em casos de insuficiência do tamanho de fluxo, caminhos de alta latência, ou perdas em redes de grande área.

Tabela 2.2: Comparação entre trabalhos.

Propriedade / Referências	DYSWIS [Kim et al., 2014b]	Epoxid [Pelle et al., 2015]	HNDR [Calvert et al., 2011]	HomeNet Profiler [DiCioccio et al., 2012]	HomeVisor [Fratczak et al., 2013]	Meraki [Biswas et al., 2015]	SDN Radar [Gheorghe et al., 2015]	WLAN-Probe [Kanuparthy et al., 2012]	WiSlow [Kim et al., 2014a]	WTF? [Sundaresan et al., 2013]	**HomeNetRescue** (HNR)
Rede Doméstica (RD)	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓
Soluciona Problemas (TS)	✓	✓	✓	✗	✓	✓	✓	✗	✓	✗	✓
SDN	✗	✗	✓	✗	✓	✓	✓	✗	✗	✗	✓
Detecção Automática (DA)	✗	✗	✓	✗	✗	✗	✗	✓	✗	✗	✓
Redes Cabeadas (ET)	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓	✓
Redes Sem Fio (WI)	✓	✗	✓	✓	✗	✓	✗	✓	✓	✓	✓
Problemas nas Estações (PE)	✓	✓	✗	✓	✗	✗	✗	✗	✓	✗	✓
Extensibilidade (EX)	✗	✓	✓	✗	✓	✓	✓	✗	✗	✗	✓

Certas características diferenciam a abordagem empregada no *HomeNetRescue* com as abordagens de outras propostas na literatura. Dentre os trabalhos relacionados, alguns propõem sistemas colaborativos para a solução de falhas, o que pode acarretar em problemas de privacidade em relação às informações dos usuários da rede. Neste sentido, o *HomeNetRescue* não armazena ou exibe os dados dos usuários a terceiros, respeitando a privacidade e segurança dos mesmos. Além disso, alguns dos trabalhos mencionados são pouco flexíveis, pois, dependem de tecnologias proprietárias, consequentemente a adição de parâmetros de monitoramento e novos tipos de aplicações ficam limitados aos fabricantes dos componentes adotados nas redes. Por sua vez, em nossa solução, o paradigma SDN em conjunto com plataformas Linux embarcadas possibilita a implementação de recursos de controle da rede de forma mais flexível.

Nossa proposta também apresenta um diferencial ao analisar as falhas em todo o escopo de uma rede doméstica (APs, roteadores, estações). O *HomeNetRescue* pode solucionar ou minimizar automaticamente os problemas detectados, diferentemente da maioria das abordagens apresentadas. Embora o *HomeNetRescue* apresente variadas

funcionalidades, a solução herda problemas característicos de arquiteturas centralizadas. O controlador é o ponto central da rede e as mensagens de controle geradas por ele podem gerar sobrecargas na rede. Por depender do controlador, se o mesmo apresentar vulnerabilidades, a execução do serviço também pode ficar comprometida. Todavia, devido a técnicas de alta disponibilidade para controladores serem um problema resolvido na literatura, o controlador não é o principal ponto de falha [Berde et al., 2014].

A visão global do controlador simplifica o gerenciamento da rede, possibilita obter soluções mais eficientes e controlar a rede com um grão mais fino, algo primordial para o processo de identificação e solução de problemas. Através dessa abordagem o serviço pode monitorar todos os componentes da rede em busca do seu funcionamento adequado. Ainda, tal visão possibilita às provedoras identificarem problemas nos dispositivos dos clientes, algo que atualmente depende de visita de técnicos ao domicílio. O resultado disso são reduções de custos para as provedoras (por não precisarem alocar profissionais em casos desnecessários). Outro benefício consiste em viabilizar melhorias de QoS, pois, com o HomeNetRescue os problemas tornam-se passíveis de serem diagnosticados e resolvidos em menor tempo e automaticamente, de modo que a rede se torne mais estável e confiável, aumentando a satisfação dos usuários.

2.5 Resumo

Nesse capítulo apresentamos os conceitos relevantes para a compreensão desta dissertação. Descrevemos também os trabalhos relacionados em uma tabela comparativa para permitir posicionar as diferenças e as semelhanças do *HomeNetRescue* em relação aos referidos trabalhos do estado da arte da área. Apesar das abordagens que utilizam redes definidas por *software* para a detecção e solução automática de problemas em redes domésticas, ainda há uma necessidade por uma abordagem que também realize esses procedimentos em redes sem fio. Conforme discutimos no Capítulo 5, o *HomeNetRescue* provê desse recursos para as redes sem fio, lidando assim com um tema de pesquisa ainda pouco investigado. No próximo capítulo apresentamos uma pesquisa *online* realizada com usuários de redes domésticas que nos permitiu obter uma visão a respeito dos problemas e das falhas presenciadas no uso cotidiano de suas redes.

Capítulo 3

Pesquisa de Opinião com os Usuários

Este capítulo apresenta uma visão a respeito dos problemas e das falhas presenciadas pelos usuários das redes domésticas em seu cotidiano. Desta maneira, conduzimos uma pesquisa de opinião online, utilizando o *GoogleForms*, onde questões relacionadas às redes domésticas dos usuários foram aplicadas. Com esse estudo obtivemos informações relacionadas aos problemas e às falhas presenciadas pelos usuários, como também sobre características das redes domésticas dos mesmos. Os resultados foram utilizados para nos direcionar, permitindo-nos configurar os cenários descritos no Capítulo 5 e avaliados no Capítulo 6.

A organização deste capítulo foi estruturada da seguinte maneira. Na Seção 3.1 apresentamos e descrevemos a metodologia utilizada na elaboração da pesquisa aplicada aos usuários de redes domésticas, ou seja, a configuração da pesquisa. Na Seção 3.2 apresentamos os gráficos resultados das respostas dos usuários e comentamos sobre as considerações a respeito de cada um. Por fim, concluímos o capítulo com a Seção 3.3.

3.1 Metologia da Pesquisa

Orientados por um questionário online que elaboramos, os colaboradores da pesquisa (usuários de redes domésticas) responderam a um conjunto de questões. O questionário continha 16 questões e estas foram utilizadas para configurar os cenários avaliados nessa dissertação. Na Tabela 3.1 apresentamos as questões aplicadas. Das 16 questões, 15 foram de múltipla escolha e uma foi aberta, onde os usuários puderam adicionar informações não cobertas nas questões adotadas. O questionário foi divulgado em listas de grupos de *e-mails* de programas de pós-graduação e graduação de algumas

faculdades, em redes sociais (*Facebook*) e em fóruns relacionados a problemas em redes domésticas. A coleta das respostas ocorreu do dia 16/08/2016 até o dia 28/02/2017. No estudo obtivemos a colaboração de 454 respondentes. Exceto a questão aberta, todas as outras foram configuradas para receber pelo menos uma resposta (obrigatória).

Na Tabela 3.1 categorizamos em quatro grupos as questões aplicadas no questionário. Identificamos as categorias em: (1) nível de conhecimento dos usuários sobre redes de computadores; (2) questões sobre as características ou configurações das redes domésticas que são utilizadas pelos usuários; (3) perfis de utilização da rede; e (4) problemas e falhas presenciadas pelos respondentes. Na seção seguinte apresentamos os resultados e as considerações a respeito das respostas dos usuários. Ressaltamos que as opções das respostas das questões foram omitidas da tabela por delimitação de espaço, mas também serão apresentadas na próxima seção. Nossa pesquisa não é uma pesquisa exaustiva, ou seja, não abrange todos os tipos de problemas em uma rede doméstica.

3.2 Resultados e Considerações

Nessa seção apresentamos os gráficos elaborados a partir das respostas dos usuários. Em cada gráfico, na caixa de legenda, destacamos as opções das respostas das questões que foram omitidas da Tabela 3.1. Para uma melhor visualização dos dados preferimos exibir gráficos de barra em detrimento de gráficos de *pizza*. As barras horizontais correspondem às porcentagens de cada resposta. No topo das barras constam os valores das porcentagens e dentro da barra a quantidade de respostas para cada opção. A ordem das opções na legenda é a mesma das barras apresentadas. Ressaltamos que a quantidade de respostas foi omitida em alguns gráficos por não caber na extensão da barra. Ainda, na descrição da legenda de cada figura repetimos a questão apresentada na tabela.

Na Figura 3.1 podemos notar que os níveis de conhecimento dos usuários que responderam ao questionário foram proporcionalmente similares considerando as opções *Básico*, *Intermediário* e *Avançado*. Isso se deu por conta da metodologia empregada para a divulgação do questionário. Acreditamos que as respostas para *Avançado* foram originárias dos usuários das listas de discussões para a solução de problemas em redes de computadores e das listas de graduação e pós-graduação em ciência da computação. No mesmo sentido, acreditamos que as respostas de *Intermediário* e *Básico* foram originadas pelos usuários que viram o questionário nas redes sociais. Na Figura 3.2 apresentamos as principais provedoras de acesso à Internet utilizadas pelos

Tabela 3.1: Conjunto de questões utilizadas no questionário aplicado.

Categoria	Questões
1	(Q1) Qual seu conhecimento sobre redes de computadores?
2	(Q2) Qual é a sua provedora de Internet?
2	(Q3) Qual a velocidade do plano contratado?
2	(Q4) Na sua casa os equipamentos são ligados ao roteador de que maneira?
4	(Q5) Quais os principais problemas/falhas apresentados na rede da sua casa?
4	(Q6) Com que frequência as falhas mencionadas na questão anterior ocorrem?
3	(Q7) Você ou outros usuários da sua rede utilizam os seguintes serviços? Quais?
3	(Q8) Em média, quantas pessoas utilizam a rede da sua casa ao mesmo tempo?
3	(Q9) Quais destes dispositivos encontram-se a menos de 10 metros do seu roteador:
3	(Q10) Quantos dispositivos (smartphones, tablets, smartTV 's, notebooks, etc) se conectam à rede sem fio?
3	(Q11) Quantos computadores estão conectados na rede sem fio?
2	(Q12) Quantos roteadores sem fio existem em sua casa?
2	(Q13) Você possui repetidores de sinal sem fio na sua casa? Quantos?
2	(Q14) Há quanto tempo seu roteador é utilizado?
3	(Q15) Qual o tipo de sua residência?

usuários. Com a pergunta tentamos cobrir outros tipos de conexões domésticas além da modalidade banda larga (ADSL), como internet via rádio, por exemplo, mas pela pouca quantidade de respostas agrupamos as respostas de outros tipos de conexões na opção *Outros*, sendo portanto a maioria das respostas correspondentes às principais provedoras de acesso Internet no Brasil.

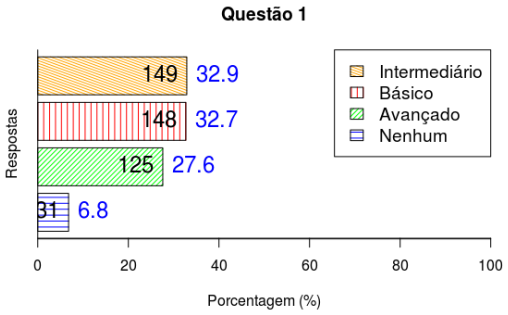


Figura 3.1: Qual seu conhecimento sobre redes de computadores?

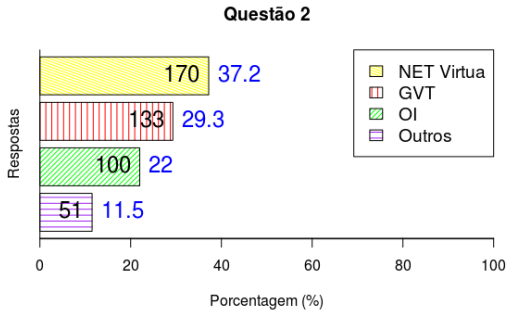


Figura 3.2: Qual é a sua provedora de Internet?

Na Figura 3.3 percebemos que a velocidade da Internet pelo plano contratado pelos usuários é baixa, a maioria apresenta uma rede doméstica com um *link* inferior a 10 *Mbps*. Esse resultado pode ser validado com o *Netflix ISP Speed Index*[1]. O índice indica que a velocidade média mais rápida de uma provedora brasileira foi de 3,18 *Mbps* considerando análises realizadas entre agosto de 2016 e de 2017. Na Figura 3.4 a comunicação através de enlaces sem fio sobressai aos enlaces com fio. Isto se deve por conta da popularização das estações móveis (*smartphones* e *notebooks*) nas redes domésticas. A partir da predominância na quantidade de conexões sem fio, nos cenários do Capítulo 5, priorizamos detectar problemas em redes sem fio em detrimento de redes cabeadas.

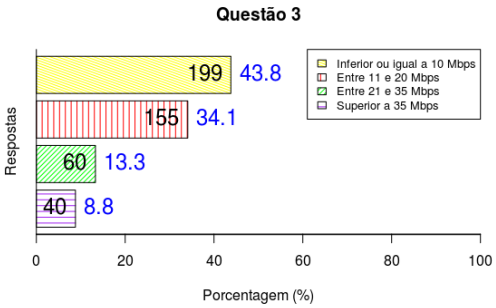


Figura 3.3: Qual a velocidade do plano contratado?

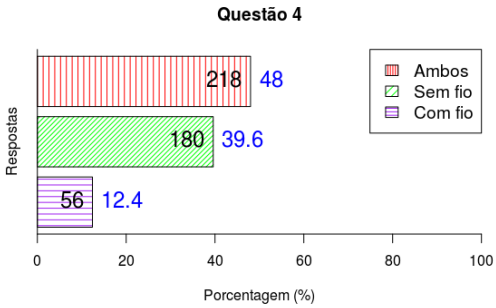


Figura 3.4: Na sua casa os equipamentos são ligados ao roteador de que maneira?

[1]https://ispspeedindex.netflix.com/country/brazil/

Na Figura 3.5 apresentamos os principais problemas constatados pelos usuários em suas redes. Nesta questão os respondentes podiam marcar múltiplas respostas. O percentual mostrado foi feito em relação ao total de respondentes e não ao total de respostas. A velocidade e as possíveis quedas da internet são os principais problemas relatados, correspondendo a 68.1.3% e 46.5%, respectivamente. Nesse sentido, nos cenários avaliados no Capítulo 6, lidamos com problemas de desempenho e de interferências nos enlaces sem fio.

Na Figura 3.6 os referidos problemas são categorizados de acordo com a sua frequência de ocorrência. As respostas indicam com qual frequência a rede do respondente apresenta alguma falha. Desta maneira, aproximadamente 33% deles são presenciados todos os dias nas redes domésticas. Em função da simplicidade do questionário, não aprofundamos na frequência para cada falha, o que poderia indicar as falhas prioritárias e, portanto direcionar o desenvolvimento de outras aplicações do *HomeNetRescue*. Acreditamos que em trabalhos futuros este ponto possa ser aprofundado.

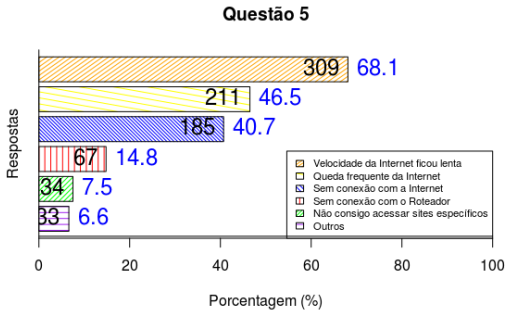


Figura 3.5: Quais os principais problemas/falhas apresentados na rede da sua casa?

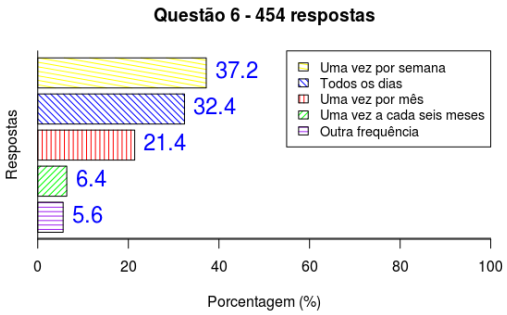


Figura 3.6: Com que frequência as falhas mencionadas na questão anterior ocorrem?

Apresentamos o perfil de utilização da rede pelos usuários na Figura 3.7 e a quantidade de usuários na Figura 3.8. Nesta questão os respondentes também podiam marcar múltiplas respostas. Notamos que os usuários utilizam-se de serviços da Internet que requerem alta de velocidade. Ainda, ilustramos que as redes domésticas de 76% dos respondentes contêm de 2 a 4 pessoas. De acordo com o *Global Web Index* (WGI)[2], estima-se, em média, 3,64 dispositivos por usuário, ou seja, aproximadamente 14 dispositivos concorrendo pelos recursos da rede, o que torna o gerenciamento de problemas em redes domésticas algo complexo de ser realizado. Adicionalmente, a quantidade de usuários aliado à necessidade de redes com maior vazão justifica a importância

[2]http://blog.globalwebindex.net/chart-of-the-day/digital-consumers-own-3-64-connected-devices/

de se propor mecanismos que agreguem robustez à rede, assim como confiabilidade e qualidade de serviço.

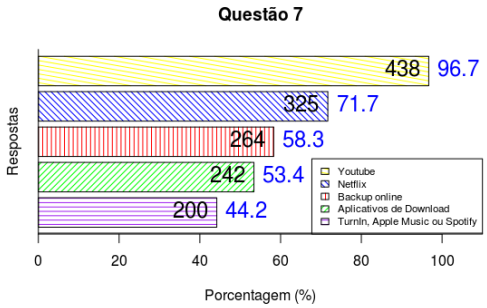


Figura 3.7: Você ou outros usuários da sua rede utilizam os seguintes serviços? Quais?

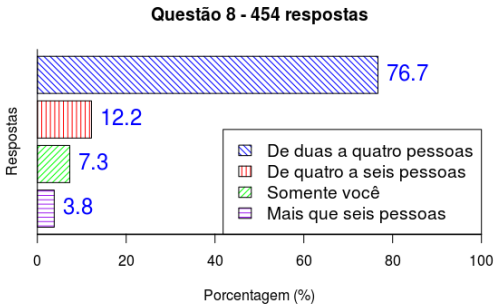


Figura 3.8: Em média, quantas pessoas utilizam a rede da sua casa ao mesmo tempo?

Na Figura 3.9 apresentamos as possíveis fontes interferentes que podem comprometer a qualidade da transmissão sem fio nas redes domésticas dos usuários. Nesse sentido, acima de 40% das redes contém telefones sem fio e aparelhos microondas próximos aos roteadores. Na Figura 3.10, de 4 a 6 dispositivos móveis dos usuários são conectados utilizando-se o meio de comunicação sem fio, ou seja, são passíveis de sofrer interferência por outros dispositivos no ambiente da rede como os mencionados na questão anterior.

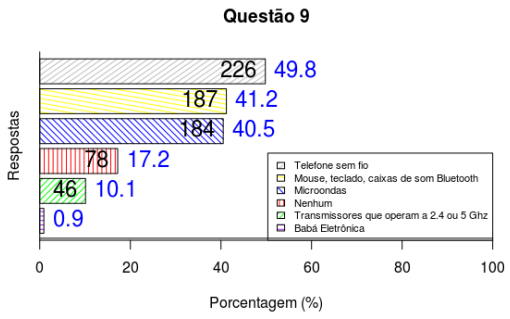


Figura 3.9: Quais destes dispositivos encontram-se a menos de 10 metros do seu roteador:

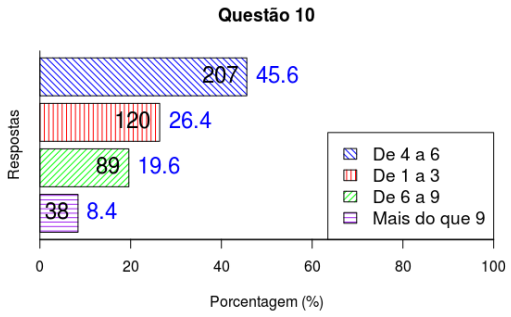


Figura 3.10: Quantos dispositivos (smartphones, tablets, smartTV's, notebooks, etc) se conectam à rede sem fio?

Na Figura 3.11 temos que a quantidade de computadores conectados na rede sem fio varia entre *dois a quatro* unidades. Nessa pergunta *desktop* e *notebooks* foram generalizados. Outra característica que pode ser observada nas redes dos usuários é que no

Brasil as residências ainda apresentam pouca quantidade de roteadores para cobrirem totalmente as suas áreas, conforme ilustramos na Figura 3.12. Neste sentido, a maioria das residências conta com apenas um dispositivo, sendo este, em geral, insuficiente para cobrir totalmente a residência dependendo do tamanho de sua área.

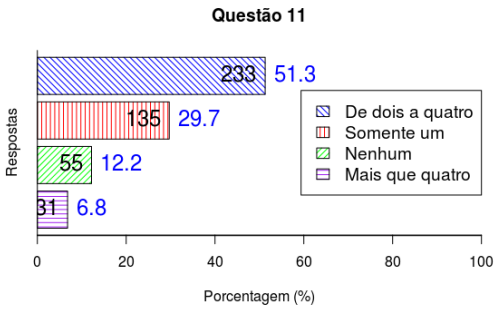


Figura 3.11: Quantos computadores estão conectados na rede sem fio?

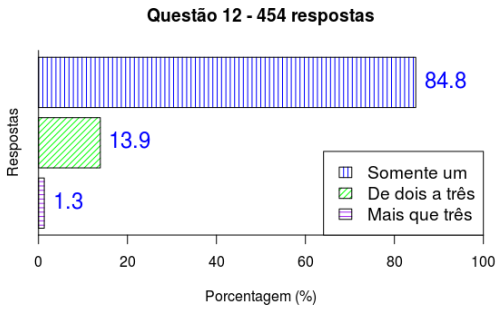


Figura 3.12: Quantos roteadores sem fio existem em sua casa?

Em complemento à questão anterior, na Figura 3.13 percebemos que, em geral, os usuários das redes domésticas também não adotam repetidores de sinal sem fio em suas redes. A utilização de repetidores poderia proporcionar melhorias na área de cobertura da rede. Na Figura 3.14 ilustramos que aproximadamente 66% dos roteadores estavam em utilização há mais de 2 anos nas residências. Na Figura 3.15 apresentamos que as redes dos usuários são implantadas principalmente em *Casas* e *Apartamentos*, ambientes que apresentam distintas características. O ambiente de um apartamento tende a ser mais denso, em termos de quantidade de usuários e redes por área, do que em uma casa, assim como são mais susceptíveis a sofrerem mais interferências pela sobreposição de canais. Em contrapartida, as casas tendem a ter áreas maiores, sendo estas, mais difíceis de serem cobertas. Tais características podem contribuir para a degradação do sinal sem fio levando a baixos índices de qualidade de experiência.

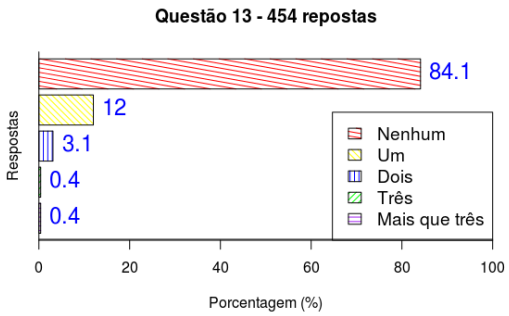


Figura 3.13: Você possui repetidores de sinal sem fio na sua casa? Quantos?

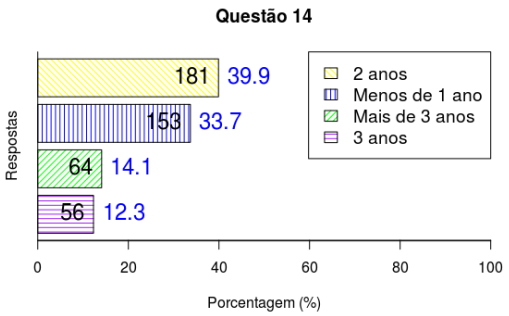


Figura 3.14: Há quanto tempo seu roteador é utilizado?

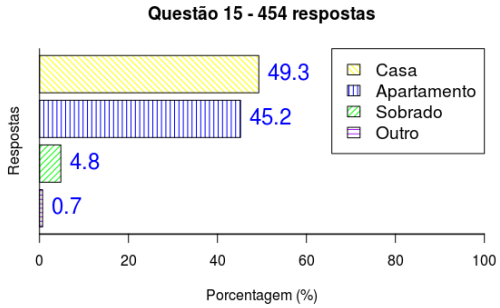


Figura 3.15: Qual o tipo de sua residência?

Por fim, a última questão não contemplada na Tabela 3.1 foi a questão *(Q16) Gostaria de descrever alguma situação de falha (as) ocorrida (as) em sua rede que não foi/foram contemplada (as) nas perguntas?*. Questão aberta onde fornecemos um espaço para os usuários adicionarem informações que não foram cobertas na pesquisa. Para ela obtivemos 54 respostas. Em geral, com essas respostas podemos perceber outras características sobre a rede dos usuários que não foram mencionadas anteriormente. Entre os relatos dos usuários estão: percepção de que os problemas ocorrem intermitentemente ou às vezes no mesmo período do dia (horários de pico); inconformidade com o plano de velocidade contratado ou com os serviços prestados pelas provedoras; problemas de má configuração dos dispositivos da rede, entre outros.

3.3 Resumo

Nesse capítulo descrevemos a metodologia da pesquisa que elaboramos para compreender as características e o perfil de utilização das redes domésticas dos usuários. Apresentamos também as questões utilizadas e, para cada questão, exibimos um gráfico com as respostas dos usuários. Ainda, fizemos algumas considerações a respeito das respostas obtidas. Ressaltamos que em trabalhos futuros realizaremos uma análise mais aprofundada com os dados coletados na pesquisa. Desejamos realizar correlações entre as características da rede, o perfil de uso da mesma e os problemas apontados pelos usuários. No próximo capítulo propomos e descrevemos as características da arquitetura do *HomeNetRescue*.

Capítulo 4

O Serviço *HomeNetRescue*

Neste capítulo propomos e apresentamos o *HomeNetRescue* (HNR). O *HomeNetRescue* é um serviço baseado em eventos para o gerenciamento autônomo de redes domésticas. Em seu contexto, os eventos podem ser representados por uma condição anômala que tenha uma ou mais métricas de redes associadas. Assim, relacionam-se às falhas, aos problemas, as questões de desempenho ou segurança, entre outras anormalidades susceptíveis em uma rede doméstica. No processo de gerenciamento, o objetivo do *HomeNetRescue* consiste em prover recursos para a detecção, o diagnóstico e a solução automática de problemas. Para tal, o serviço baseia-se no paradigma de redes definidas por *software* (SDN) para o gerenciamento de redes domésticas com e sem fio, e é composto por uma arquitetura modular e expansível, podendo ser empregado na resolução de variados problemas nesses ambientes.

O *HomeNetRescue* foi desenvolvido para ser executado sob demanda ou de forma agendada. Ainda, foi concebido para poder ser administrado por provedoras de acesso à internet. Objetivamos que o *HomeNetRescue* possa ser executado a partir de *call centers*, quando for necessário realizar diagnósticos ou soluções de problemas nas redes, através de chamada de suporte (sob demanda) ou de forma agendada (permanecendo em execução automática).

Este capítulo está organizado da seguinte maneira. Na Seção 4.1 apresentamos uma descrição geral do *HomeNetRescue*. Na Seção 4.2 discutimos sobre a arquitetura de *software* com seus componentes e funcionalidades. Descrevemos, na Seção 4.3, casos de aplicações onde o HNR pode ser empregado de modo a detectar e resolver problemas em redes domésticas. Por fim, concluímos o capítulo na Seção 4.4.

4.1 Visão Geral do *HomeNetRescue*

Desde o surgimento do Ethane, precursor do conceito das redes definidas por *software*, o paradigma SDN tem recebido ampla ênfase da comunidade científica e das empresas ao proporem soluções para os problemas e desafios de redes de computadores, conforme consta nos anais de importantes conferências da área, como SIGCOMM, INFOCOM, SBRC, entre outras. A aplicação do paradigma SDN proporciona distintos e importantes recursos às redes domésticas, algo que exploramos na proposição do *HomeNetRescue*. Ao modelá-lo utilizando SDN objetivamos usufruir da flexibilidade proporcionada pelo paradigma em detrimento de uma solução sem a sua utilização. Em nossa abordagem SDN possibilita ao HNR controlar com um grão mais fino o processo de gerenciamento das redes domésticas.

Acreditamos também que SDN proporciona benefícios primordiais para o processo de detecção, diagnóstico e solução de problemas. Entre eles, podemos citar: maior flexibilidade de programação devido à separação dos planos de controle e dados; maior granularidade no processo de monitoramento e solução de problemas; gerenciamento centralizado da rede (visão global); menor custo operacional para desenvolver aplicações e soluções; facilidade na orquestração dos componentes da rede efetuada pelo controlador; possuir interfaces programáticas abertas; entre outros benefícios.

As redes domésticas caracterizam-se por possuírem pelo menos um ponto de comunicação entre a rede local e a rede externa, sendo as provedoras de acesso à internet as responsáveis por permitir tal comunicação. Assim, sob a perspectiva das provedoras de acesso, a arquitetura SDN do *HomeNetRescue* pode permitir a ampliação do escopo de atuação de suas redes, para além dos *gateways*, nas redes dos seus clientes; adicionar maiores recursos para a detecção de problemas, como uma ferramenta adicional aos clientes; viabilizar melhorias de QoS; possibilitar a redução de custos com deslocamento de técnicos às residências e da complexidade de gerenciamento dos componentes; melhorar o controle de fluxos de dados via protocolos; como também proporcionar melhorias no processo de identificação da origem dos problemas em suas redes.

Sob a perspectiva dos clientes das provedoras, os benefícios citados podem refletir em melhorias de QoE; simplificação do processo de detecção de problemas; maior alcance de monitoramento dos componentes a serem monitorados conforme escalabilidade da rede (crescimento da quantidade dos dispositivos convencionais ou IoT nessas inseridos) minimizando seus impactos de gerenciamento, entre outros.

Em complemento às características do HNR destacamos também uma característica herdada do Openflow, o principal, mas não único protocolo SDN. Na proposição do

HNR também utilizamos uma abordagem baseada na programação orientada a eventos. Diferentemente das arquiteturas convencionais que utilizam um fluxo de controle padronizado, no serviço, o controle de fluxos das aplicações são orientadas a eventos e são guiados pelas chamadas destes. Em vez de esperar por comandos para processar a informação de forma síncrona, o serviço permanece em execução aguardando eventos que disparam funções de resposta de acordo com o evento, de forma assíncrona. Em outras palavras, o *HomeNetRescue* monitora parâmetros e condições da rede e caso sejam detectadas anormalidades ou problemas, gera eventos que disparam funções implementadas para a sua solução. A seguir apresentamos a arquitetura de rede e, posteriormente, a arquitetura de *software* do serviço.

4.1.1 Arquitetura de Rede do *HomeNetRescue*

A arquitetura de rede do *HomeNetRescue* abrange os componentes físicos (*hardwares*) que compõem uma rede doméstica. Na arquitetura se enquadram os componentes de controle, os componentes de comunicação cabeada ou sem fio e as estações de trabalho. Conforme ilustrado na Figura 4.1 consideramos uma rede doméstica composta por pontos de acesso, roteadores, comutadores, estações (*notebooks*, *smartphones* e *desktops*), entre outros. Neste contexto, classificamos esses componentes em três níveis hierárquicos conforme a funcionalidade de cada um, sendo eles: gerenciamento e controle; comunicação e ponte; e clientes.

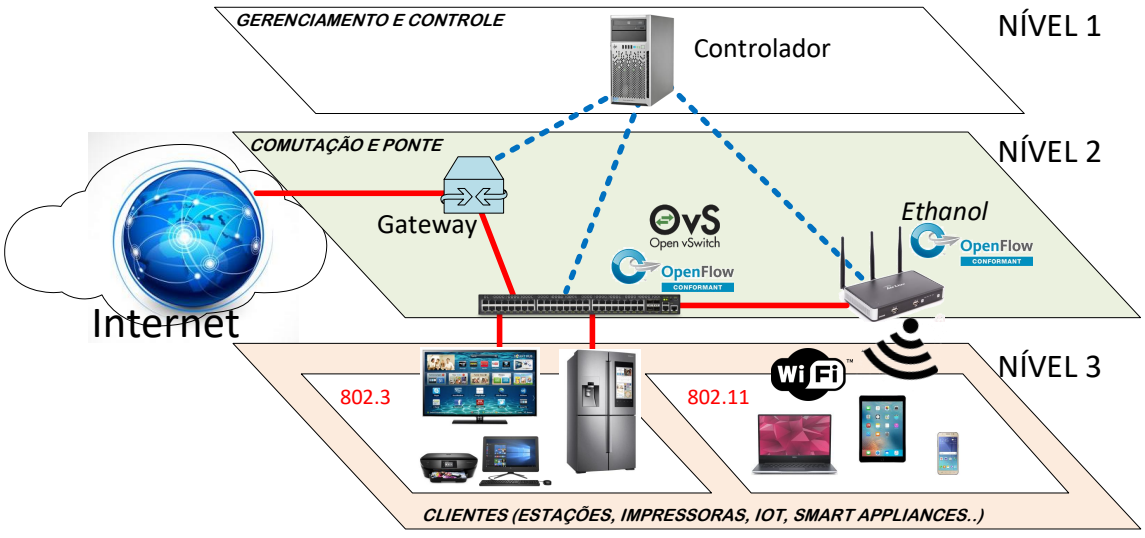


Figura 4.1: Arquitetura de rede do *HomeNetRescue*.

No primeiro nível, *gerenciamento e controle*, temos o *hardware* (controlador) responsável por controlar os demais componentes físicos da rede. Ele é capaz de controlar

os roteadores, os comutadores e as estações. O controlador pode ser representado por um *hardware* com maior capacidade de processamento (*desktops*) ou por componentes com baixa capacidade de processamento como é o caso dos roteadores, por exemplo. O *software* executado no controlador será explicado na próxima seção.

No segundo nível, *comutação e ponte*, temos os *gateways*, os roteadores, os pontos de acesso e os comutadores. Através do *gateway* da rede é realizada a comunicação da rede local com a rede externa. Os componentes desse nível se dedicam a realizar atividades como roteamento, interligação dos dispositivos, interpretação de sinais digitais, encaminhamento de pacotes, estabelecimento de fluxos, solicitação de métricas das estações, entre outras.

Por fim, no terceiro nível, *clientes*, temos as estações dos clientes como *notebooks*, *smartphones* e *desktops*. As estações são ativas permitindo a coleta de informações e desempenham atividades como responder às requisições dos roteadores.

4.2 Arquitetura de *Software* do *HomeNetRescue*

A arquitetura de *software* do *HomeNetRescue* é composta por planos, camadas e módulos. Nessas subdivisões existem *softwares* implementados e estes constituem a referida arquitetura. Esses *softwares* são executados tanto no controlador quanto nos dispositivos a serem controlados, dispositivos esses ofertados pelas provedoras. No planejamento da arquitetura foi considerado o tratamento de eventos de falhas ou de problemas a partir da camada de enlace até a camada de aplicação do modelo TCP/IP. Nesta seção apresentamos também os planos, as camadas e os módulos com suas respectivas funcionalidades, mas antes apresentamos uma terminologia para melhor compreensão dos termos utilizados na descrição da arquitetura de *software* do serviço. Os termos são:

- **Serviço**: programa que pode ser executado nos componentes da rede doméstica com o intuito de fornecer funcionalidades e recursos para o gerenciamento desta.
- **Componente**: corresponde ao *hardware* da rede doméstica que pode ser gerenciado pelo controlador, por exemplo: comutadores, roteadores, estações, entre outros.
- **Plano**: representa uma divisão organizacional, em maior granularidade, que separa o serviço de acordo com suas funcionalidades.
- **Camada**: subdivide os planos hierarquicamente agrupando as funcionalidades semelhantes.

- **Módulo**: corresponde a uma unidade lógica organizacional que desempenha no serviço atividades ou tarefas específicas pelas quais foram programadas.

- **Aplicação**: interface entre o administrador da rede e o *HomeNetRescue*. As aplicações implementam os cenários e fornecem um conjunto de recursos para o gerenciamento da rede ao administrador. São exemplos: aplicação para o gerenciamento de mobilidade em redes sem fio, aplicação para o balanceamento de carga da rede, aplicação para redução de interferência, entre outras.

- **Regra**: definição de um conjunto de parâmetros como limiares, métricas, protocolos, entre outros, utilizados pelo *HomeNetRescue* para o gerenciamento das aplicações.

- **Cliente**: pessoa responsável por contratar e utilizar os serviços oferecidos pelas provedoras de acesso à internet.

- **Conflito**: condição de rede na qual aplicações distintas requisitam simultaneamente os mesmos recursos de rede e o *HomeNetRescue* precisa decidir sobre uma ação, por exemplo: aumentar ou diminuir da potência de transmissão de um mesmo roteador quando requisições de aplicações distintas ocorrem simultaneamente.

Retomando à descrição da arquitetura de *software* do *HomeNetRescue*, temos que a mesma foi dividida em três planos, conforme apresentado na Figura 4.2. Os planos são: *i*) **plano de gerenciamento de problemas**: plano extensível onde são executados e implementados os algoritmos de detecção e gerenciamento de problemas; *ii*) **plano de controle**: corresponde a um *software* (controlador) que pode ser executado de forma distribuída ou centralizada em um ou mais componentes da rede da provedora; e *iii*) **plano de dados**: responsável pelas funções de encaminhamento ou roteamento realizadas nos dispositivos de rede a serem gerenciados pelo serviço. A seguir descrevemos mais detalhes sobre cada plano.

4.2.1 Plano de Gerenciamento de Problemas

O *Plano de Gerenciamento de Problemas* corresponde à interface de interação entre o administrador da rede e o *HomeNetRescue*, assim como ao conjunto de aplicações do serviço. O plano representa a inteligência da rede. Através dele são realizadas as chamadas para as funções a serem executadas nos componentes da rede, como, por exemplo, consultar o estado da rede, obter notificações sobre a ocorrência de eventos,

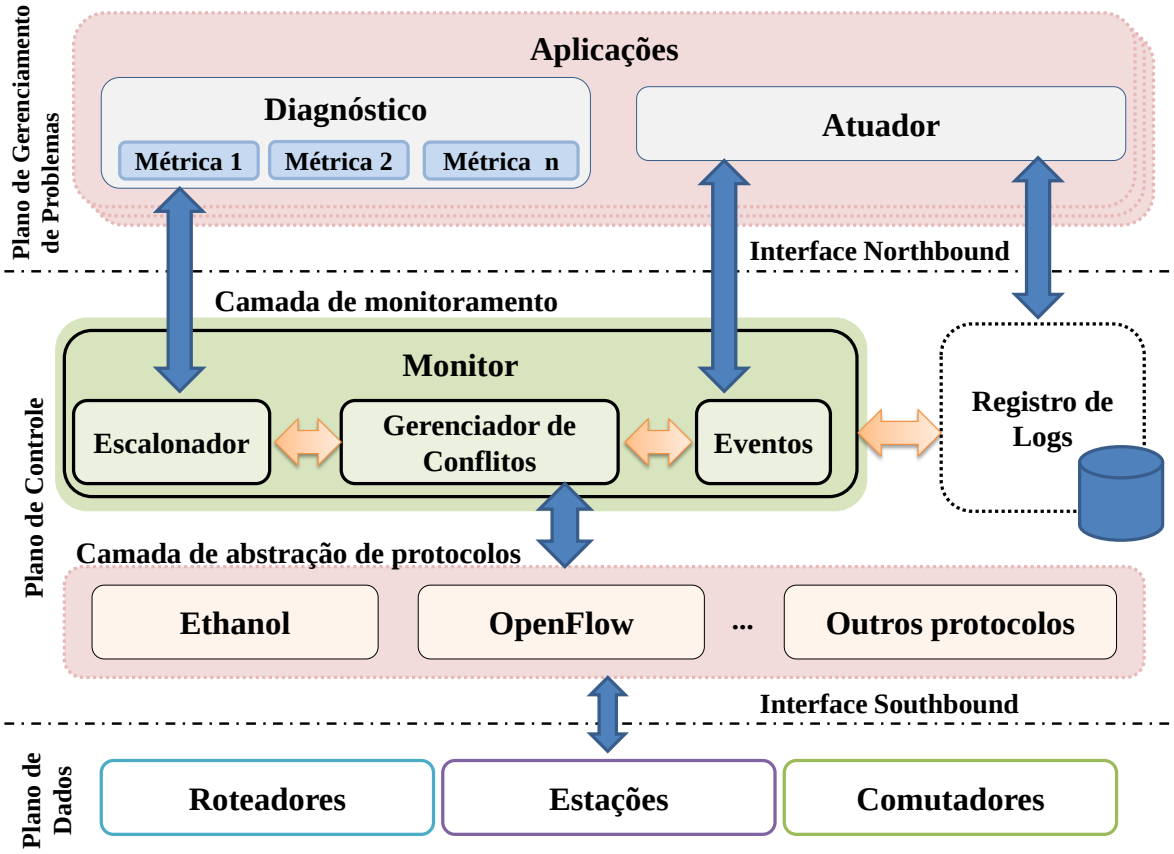


Figura 4.2: Arquitetura do serviço.

entre outras funções. Ainda, é o responsável por possibilitar ao administrador da rede configurar as políticas, as ações, as regras e os parâmetros das aplicações a serem gerenciadas pelo *HomeNetRescue*.

As aplicações correspondem aos *softwares* que permitem ao administrador realizar as configurações das políticas e, essas, são executadas no *HomeNetRescue*. O HNR apresenta aplicações implementadas, mas também fornece recursos para a implementação de novas aplicações. Em seu contexto, uma ou mais aplicações podem ser executadas em paralelo. Tais aplicações adicionam aspectos a serem monitorados, fornecem as funcionalidades de detecção e solução de problemas e cada uma possui dois módulos, diagnóstico e atuador.

O módulo *Diagnóstico* permite ao administrador da rede realizar o registro de métricas com seus intervalos para o monitoramento e as prioridades das aplicações no módulo *Monitor*. Ao executar uma aplicação, inicia-se o diagnóstico das métricas configuradas, ativando também os outros módulos do serviço (descritos na Seção 4.2.2). Esse módulo interliga-se ao submódulo *Escalonador*.

O módulo *Atuador* é o responsável por interagir, através do serviço, com os

componentes da rede possibilitando a alteração de seus parâmetros. Na aplicação são configuradas as ações a serem realizadas caso um evento (resultado do monitoramento) da rede seja gerado. Esse módulo é invocado para atuar na rede sempre que limiares pré-definidos forem atingidos. Com isso, ele pode executar funções que alterem parâmetros nos componentes de rede objetivando solucionar um problema detectado. O *Atuador* interliga-se ao submódulo *Eventos* e ao módulo *Registro de Logs*.

De acordo com a Figura 4.2 ambos os módulos (*Diagnóstico* e *Atuador*) comunicam-se com outros módulos da camada subjacente da arquitetura (*Plano de Controle*). Os módulos desse plano são programáveis de modo a favorecer a flexibilidade para a implementação e a adição de novas aplicações a serem gerenciadas pelo *HomeNetRescue*. Os parâmetros disponíveis pelo *HomeNetRescue* correspondem aos disponibilizados pelos protocolos da *camada de abstração de protocolos* (Seção 4.2.2). Adicionalmente, a interligação entre o *plano de gerenciamento de problemas* e o *plano de controle* se dá através da interface *Northbound*.

A interface *Northbound* é a API que habilita a comunicação aplicações–plano de controle, possibilitando às aplicações realizarem o controle e o monitoramento da rede sem a necessidade do administrador da rede realizar ajustes de detalhes mais finos relativos a essa comunicação. Além disso, também se responsabiliza pela tradução das informações das aplicações em instruções de baixo nível para o controlador, como também repassa às aplicações informações estatísticas geradas pelos componentes e processadas pelo controlador. Tal interface segue a linguagem especificada pelo controlador utilizado.

Como demonstração do funcionamento dos módulos desse plano, apresentamos uma aplicação para a detecção de problemas no sinal de transmissão sem fio proveniente de interferências. Nessa, o administrador configuraria a política de monitoramento da rede. Por exemplo, definiria o diagnóstico da rede a cada 60 segundos considerando as métricas *Signal-to-Noise-Ratio* (SNR) e porcentagem de pacotes perdidos em um roteador. Configuraria também o limiar, a prioridade da aplicação e a ação a ser realizada caso o limiar seja superado. Deste modo, para o diagnóstico, os limiares poderiam ser definidos como 10 dBm para SNR e 10% para taxa de perdas e a prioridade da aplicação, enquanto para a atuação, a ação poderia ser a alteração da potência de transmissão dos APs. A partir desses procedimentos tais parâmetros seriam registrados no módulo *Monitor*. Na seção seguinte descrevemos a continuidade desse exemplo contextualizando com o funcionamento dos módulos de cada camada.

4.2.2 Plano de Controle

O *Plano de Controle* é o responsável por controlar os recursos gerenciáveis da rede. Esse plano contém um conjunto de programas que facilita as atividades das aplicações. É representado por um controlador de rede. Equivale a um sistema operacional de uma rede de computadores. No *Plano de Controle*, o controlador é capaz de realizar o controle da rede de forma logicamente centralizada ou logicamente distribuída. Nesta dissertação utilizamos a primeira abordagem. Por meio dele, os recursos gerenciáveis, componentes do *Plano de Dados* (roteadores, comutadores e estações), aplicações, protocolos, módulos e camadas do *HomeNetRescue* são controlados.

O plano também fornece recursos para a análise do estado da rede; permite a descoberta da topologia ou dos dispositivos conectados (visão global da rede, algo primordial para a solução de problemas); possibilita o controle dos componentes; realiza o processamento e o encaminhamento de eventos, entre outros recursos. Outra função do plano consiste em dispensar (abstrair) que os desenvolvedores se preocupem com funções de baixo nível, como, por exemplo, saber sobre como é realizada a transmissão de pacotes na rede para se definir uma política em uma aplicação. Em sua organização, o plano foi dividido em duas camadas, a saber: *camada de monitoramento* e *camada de abstração de protocolos*, descritas a seguir.

4.2.2.1 Camada de Monitoramento

A *Camada de Monitoramento* é a primeira camada do *Plano de Controle* no *HomeNetRescue*. O papel dessa camada consiste em monitorar as informações recebidas das aplicações e corresponde ao núcleo do serviço. Entre as funcionalidades da camada estão: realizar o monitoramento da rede, escalonar leituras de aplicações segundo prioridades, registrar *logs*, gerenciar conflitos dadas as leituras, disparar eventos em caso de ocorrências, entre outras funcionalidades. A camada é composta pelos módulos *Monitor* e *Registro de Logs*, descritos a seguir.

O módulo *Monitor* é o responsável por realizar o monitoramento da rede. Por meio dele são realizadas as coletas de dados que permitem o serviço detectar e solucionar um problema, a execução de políticas, o gerenciamento de conflitos de leituras e de prioridades de aplicações, e a geração de eventos. No módulo *Monitor* são realizadas as agregações dos dados, o processamento de métricas e a consolidação do parâmetro de tempo de monitoramento informado na aplicação, ou seja, após o módulo *Diagnóstico* registrar as políticas, esse módulo se encarrega de realizar tais atividades para manter as aplicações em execução.

Retomando o exemplo mencionado na seção 4.2.1, temos que após a configuração dos parâmetros das aplicações, o módulo *Diagnóstico* registra as regras no módulo *Monitor*. A partir disso, esse último módulo inicia o monitoramento, a cada 60 s, das métricas SNR e perda de pacotes. Contidos no módulo *Monitor* estão três submódulos, sendo eles:

- *Escalonador*: corresponde a um submódulo organizacional para o escalonamento das leituras das aplicações do serviço. Leituras de aplicações distintas podem ser executadas em paralelo e concorrerem entre si. É o responsável por priorizar a execução da leitura da aplicação com maior prioridade ou com mais recursos. Dadas as aplicações, o módulo realiza requisições periódicas das métricas registradas, consulta conflito de regras e dispara eventos de atuação.

Ainda no exemplo, antes do início do monitoramento, o módulo *Escalonador* verifica entre as aplicações em execução a de maior prioridade. Se houver outra aplicação com prioridade inferior a prioridade informada na aplicação, esta deve ser sobrescrita pelas leituras da nova aplicação, caso contrário, permanece em execução paralelamente às outras. Além disso, se uma aplicação solicita a leitura do SNR a cada 60 s e outra a cada 120 s, o escalonador evita o disparo duplo de leitura de aplicações distintas. Deste modo, o *Escalonador* contribui para a otimização da coleta de dados originados de aplicações distintas, o que consequentemente reduz a carga do controlador na rede.

- *Gerenciador de Conflitos:* corresponde a outro submódulo organizacional para o gerenciamento de possíveis conflitos entre regras das aplicações que podem ocorrer no *HomeNetRescue*. Ele é o responsável por averiguar se as ações de atuação enviadas por diferentes aplicações podem causar comportamento conflitante nos dispositivos ou nos protocolos utilizados na interface *southbound*. Em trabalhos futuros pretendemos usar um sistema gerenciador de conflitos baseado em prioridades.

Na sequência do exemplo, após os procedimentos realizados pelo escalonador o *HomeNetRescue* realiza outra análise. Desta vez sobre o conflito de regras de aplicações, pois podem existir casos onde aplicações distintas requisitam o mesmo recurso. Por exemplo, uma aplicação pode solicitar o aumento da potência de transmissão para reduzir a taxa de perda, enquanto outra pode solicitar a redução da potência de transmissão para reduzir a interferência entre os APs. Em caso similar, o módulo de *Gerenciamento de Conflitos* é o responsável por resolver tais situações considerando também a prioridade de cada aplicação. No HNR aplicações de maior prioridade sobrescrevem as ações das aplicações de menor prioridade.

- *Eventos:* processa os eventos gerados na rede a partir das medições periódicas realizadas pelo submódulo *Escalonador*. Em caso de recebimento de eventos o submódulo dispara o módulo *Atuador* para que o mesmo interaja com a rede aplicando as ações configuradas nas aplicações.

Por fim, descrevendo não mais o exemplo, mas a camada de monitoramento, temos o módulo *Registro de Logs*. Ele é o responsável por registrar os *logs* das operações executadas pelos outros módulos e submódulos do serviço. Este módulo permite obter um histórico dos procedimentos realizados pelo *HomeNetRescue*, viabilizando auditorias da rede e do serviço pelos administradores da rede.

4.2.2.2 Camada de Abstração de Protocolos

A segunda camada do *Plano de Controle* representa o módulo responsável por lidar com os protocolos de comunicação e gerenciamento dos componentes de rede que podem ser utilizados no serviço. Esses protocolos compõem a interface *Southbound* do serviço, ou seja, são as interfaces entre o *Plano de Controle* com os componentes da rede (*Plano de Dados*). Projetamos o serviço para lidar e suportar múltiplos protocolos como, por exemplo, os protocolos Openflow e Ethanol. Ambos os protocolos possibilitam a comunicação entre o controlador com os roteadores, mas somente o Ethanol, com as estações. Contudo, a arquitetura também pode permitir que sejam adicionados outros protocolos como NetConf, SNMP, entre outros.

4.2.3 Plano de Dados

Esse plano é composto pelos dispositivos de encaminhamento ou roteamento da rede que suportam pelo menos um dos protocolos ativos no *HomeNetRescue*, ou seja, é o plano que se responsabiliza pelo encaminhamento de dados na rede. Devido à capacidade limitada dos componentes desse plano, exceto das estações (*smartphones*, *notebooks*, entre outras), geralmente, os roteadores e pontos de acesso processam regras mais simples, como encaminhamento de pacotes para uma porta de saída, respondem por requisições de métricas solicitadas pelo controlador, entre outras.

Nessa dissertação são empregados planos de dados compatíveis com os protocolos OpenFlow e Ethanol, mas outros podem ser utilizados. Assim, o *HomeNetRescue* suporta switches SDN (via OpenFlow), bem como APs IEEE 802.11 (via Ethanol e Hostapd) e estações que implementam recursos de gerenciamento do protocolo IEEE 802.11/2012 (também via Ethanol). Nos *APs* IEEE 802.11, o *HomeNetRescue* pode realizar operações como mudar a frequência e canais de operação, modificar a potência

de transmissão, os parâmetros do protocolo MAC (uso de RTS/CTM, DTIM, etc), solicitar varreduras do espectro, gerenciar parâmetros de QoS, controlar a associação de estações, entre outras operações.

Nas *estações*, devido ao suporte aos padrões IEEE 802.11 mencionados, é possível receber informações sobre a interface de rede sem fio, solicitar varreduras do espectro, ajustar parâmetros da camada física tais como a potência de transmissão, alterar o *bitrate*, realizar a troca para outra rede WiFi, entre outros ajustes. Vale ressaltar que na versão atual do Ethanol as estações devem executar um *software* que implementa os padrões de gerenciamento do IEEE 802.11, pois o kernel do Linux ainda não possui suporte para eles. Entretanto, tal software será desnecessário quando o suporte aos padrões for nativo.

4.3 Aplicabilidade da Solução

O *HNR* foi desenvolvido para permitir o gerenciamento autônomo de problemas e falhas em redes domésticas. Conforme mencionado na Seção 4.2 objetiva-se que o serviço possa ser executado pelas provedoras de acesso à internet, sendo esta uma alternativa de gerenciamento, à distância, das redes nas residências dos clientes. Diferentemente das redes empresariais, planejadas e configuradas por seus administradores, as redes domésticas não apresentam *hardwares* e *softwares* padronizados, assim como administradores dedicados para lidar com os possíveis problemas que essas redes são susceptíveis. Disso, surge a demanda por uma solução extensível, customizável e modular para o gerenciamento desse tipo de rede, como também para a solução de problemas e falhas. Ao permitirmos que novas aplicações possam ser facilmente executadas no *HomeNetRescue*, bem como novos dispositivos e parâmetros sejam por ele monitorados, apresentamos a sua extensibilidade, flexibilidade e customização.

A seguir listamos possibilidades adicionais de aplicação do *HomeNetRescue* para as provedoras de acesso à Internet:

- Coordenação das redes sem fio domésticas de uma mesma provedora em ambientes densos (por exemplo, em um condomínio), ajustando a potência de transmissão e o canal dos pontos de acesso para reduzir a interferência e melhorar a qualidade do serviço oferecido;

- Viabilizar a detecção de áreas de sombra (desvanecimento) nas residências. Essas áreas poderiam ser minimizadas ajustando-se dinamicamente a potência de transmissão dos roteadores;

- Detectar em sua rede roteadores defeituosos ou com SSDIs inválidos através dos *beacons* obtidos da localidade da rede em análise. Disso, a provedora poderia sugerir a substituição do aparelho;

- Gerenciar o ponto de conexão entre os roteadores e as estações, requisitando que as estações reassociem (*handover*) a outros roteadores visando obter melhor qualidade de conexão ou situação de balanceamento de carga;

  Para os usuários, o HNR também pode fornecer outros tipos de benefícios:

- O usuário pode contratar uma empresa para gerenciar sua rede, conforme proposto em [Feamster et al., 2007];

- Em vez de analisar os dados em um dispositivo com recursos restritos (por exemplo, um roteador doméstico), o paradigma SDN permite o controle da rede na nuvem. Isso permite o uso de interfaces mais responsivas e análises mais complexas (por exemplo: utilizando técnicas de *big data*) [Kim et al., 2011];

- Um provedor de serviços, provendo um serviço de transmissão de vídeo, poderia otimizar a configuração da rede doméstica dos usuários para oferecer a melhor experiência aos mesmos. Por exemplo, seria capaz de aumentar a potência de transmissão ou priorizar o fluxo dos clientes de vídeo afim de reduzir as perdas de pacotes e melhorar a taxa de transferência.

4.4 Resumo

Conforme descrito, nesse capítulo descrevemos, em alto nível, as características da arquitetura do *HomeNetRescue* e algumas de suas possibilidades de aplicação. O serviço se destaca por sua arquitetura modular que fornece flexibilidade e extensibilidade para inclusão de novas aplicações, componentes e métricas a serem monitoradas em uma rede doméstica. No capítulo a seguir apresentamos a implementação de um protótipo do *HomeNetRescue*. Através do protótipo foi possível configurar os cenários para avaliar o serviço e descrever as observações a respeito dos seus benefícios em ambiente real.

Capítulo 5

Projeto e Implementação do *HomeNetRescue*

Neste capítulo apresentamos as decisões de projeto para a modelagem e a implementação do *HomeNetRescue*. Descrevemos detalhes como as linguagens e as tecnologias utilizadas, os diagramas, as plataformas de *hardware*, os algoritmos e os cenários elaborados para demonstrar o funcionamento do serviço em um ambiente real.

A organização deste capítulo foi estruturada da seguinte maneira. Nas Seções 5.1 e 5.2 apresentamos os diagramas de classes e de sequência do *HomeNetRescue*, respectivamente. Discutimos os detalhes técnicos para a implementação do serviço na Seção 5.3. Na Seção 5.4 apresentamos a plataforma de *hardware*. Nela comentamos sobre os componentes utilizados na rede do *HomeNetRescue*. Descrevemos na Seção 5.5 os cenários configurados para permitir a avaliação do serviço. Por fim, na Seção 5.6 concluímos o capítulo.

5.1 Diagrama de Classes do *HomeNetRescue*

Nesta seção apresentamos um diagrama de classes para descrever a implementação do *HomeNetRescue* complementando as descrições apresentadas no capítulo anterior. De acordo com o diagrama da Figura 5.1, as propriedades e os objetos das classes são manipulados pelo *HomeNetRescue* com o auxilio do controlador Ethanol (descrito na próxima seção), que realiza as requisições das informações aos componentes de rede, sejam roteadores, comutadores ou estações. As classes implementadas pelas aplicações foram destacadas com a cor mais escura enquanto as classes implementadas pelo *HomeNetRescue* estão com a cor mais clara no diagrama. A seguir apresentamos as descrições das classes listadas na referida figura.

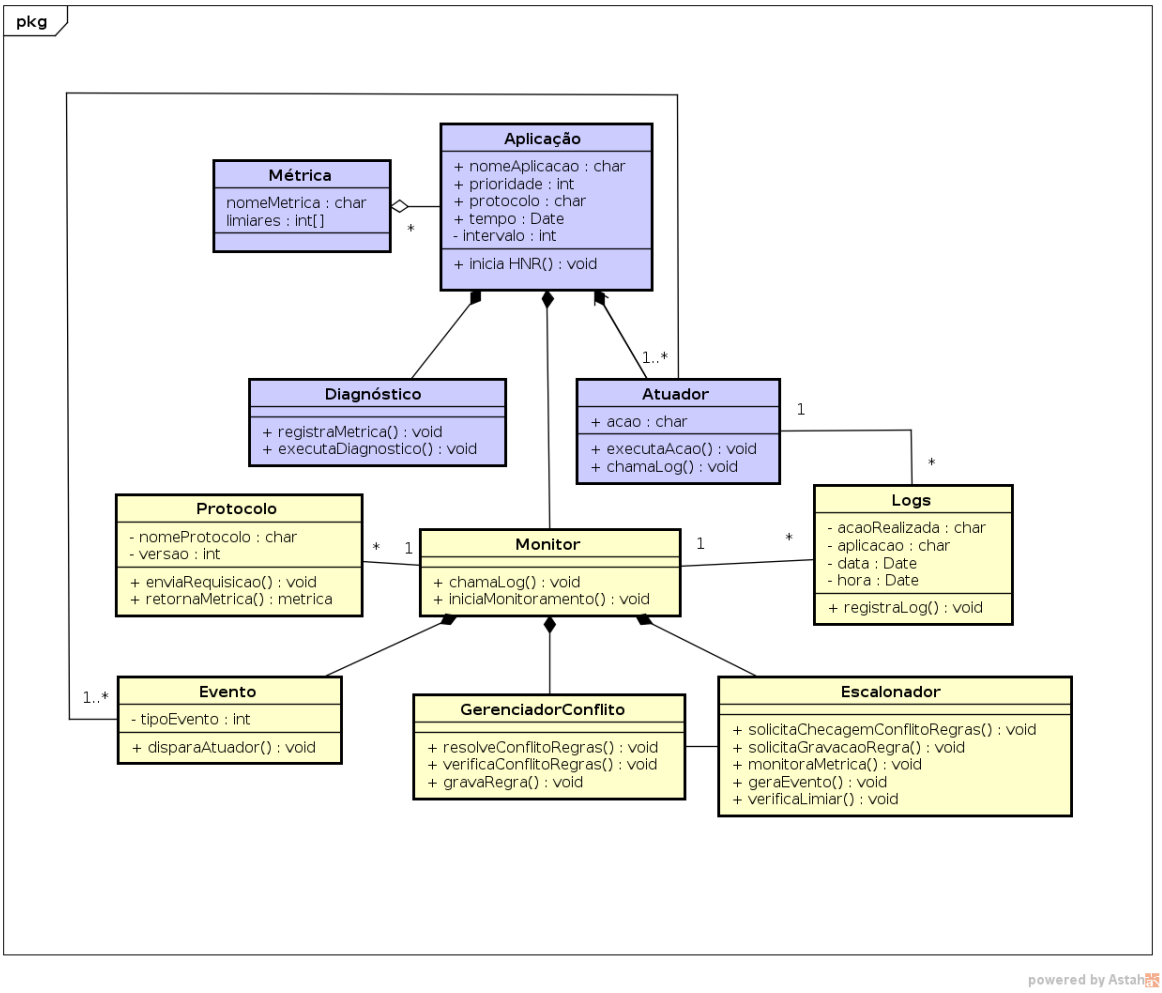


Figura 5.1: Diagrama de classes do *HomeNetRescue*.

- **Aplicação**: esta entidade encapsula os detalhes das aplicações do *HomeNetRescue*. Uma aplicação é composta pelas propriedades: `nomeAplicacao`, atributo identificador da aplicação no contexto do serviço, sejam nos módulos de controle ou no módulo de registro de log; `prioridade`, define a prioridade para a aplicação realizar as leituras das métricas no serviço - quanto maior o valor, maior a prioridade da aplicação; `protocolo`, representa o protocolo a ser utilizado; `tempo`, define o tempo de monitoramento para a aplicação; e, `intervalo`, corresponde aos intervalos de leituras que uma aplicação deseja realizar. O método invocado nessa classe é o `inicia_HNR()`, que inicia a execução do serviço.

- **Métrica**: esta entidade encapsula as métricas e os limiares monitorados pelo *HomeNetRescue*: `nomeMetrica`, corresponde às métricas que podem ser selecionadas em uma aplicação para que o serviço realize o seu monitoramento; e `limiares`, estabelecem os padrões aceitáveis para as métricas de rede.

- **Diagnóstico**: encapsula os métodos que permitem ao serviço realizar o diagnóstico da rede. O método `registraMetrica()` agrupa os parâmetros oriundos da classe *Aplicação* para os registrar na classe *Monitor*. Esse conjunto de parâmetros corresponde a uma regra no contexto do *HomeNetRescue*. O método `executaDiagnostico()` ativa o método `iniciaMonitoramento()` na classe *Monitor*.

- **Atuador**: representa as ações de atuação que podem ser realizadas pelas aplicações. Essa entidade possui o atributo `acao` e os métodos responsáveis por realizar as alterações nos parâmetros na rede. O objeto dessa classe primeiramente realiza a chamada do método `disparaAtuador()`, que realiza a ação na rede invocando o método `executaAcao()`. Essa classe ainda tem o método `chamaLog()` para o registro de logs das operações efetuadas.

- **Protocolo**: corresponde aos protocolos de comunicação que podem ser empregados no *HomeNetRescue*. Possui os atributos `nomeProtocolo` e `versao` para a identificação do nome e da versão do protocolo, respectivamente. Seus métodos são: `enviaRequisicao()`, que repassa aos componentes da rede as requisições realizadas pelo serviço e `retornaMetrica()`, que retorna a informação requisitada.

- **Monitor**: encapsula os detalhes de monitoramento do serviço. É uma classe mais genérica que fornece recursos para as suas classes filhas (*Eventos*, *GerenciadorConflito* e *Escalonador*). Essa classe também herda as características da classe *Aplicacao*. Seus métodos são: `iniciaMonitoramento()`, responsável por ativar os métodos nas classes filhas e `chamaLog()`, para regristrar as operações realizadas.

- **Logs**: classe que representa os objetos dos logs gerados a partir das operações realizadas nos módulos do *HomeNetRescue*. Os atributos são `acaoRealizada`, `aplicacao`, `data` e `hora` e o método `registraLog()`.

- **Evento**: representa eventos da rede do *HomeNetRescue*. Conforme a rede é monitorada pelo serviço eventos podem ser gerados através da detecção de condições que caracterizam as anormalidades de rede. Tal condição, no contexto do serviço, representa um evento. Quando um evento é gerado têm-se a necessidade de atuar na rede e isso é realizado através do método `disparaAtuador()`. O atributo dessa classe é `tipoEvento`, identificando o tipo do evento gerado.

- **GerenciadorConflito**: corresponde à classe onde os conflitos de diferentes aplicações são gerenciados. Possui o método `verificaConflito()` para averiguar a existência de conflitos entre aplicações distintas; o método `resolveConflito()` para priorizar a leitura da aplicação com maior prioridade e o método `gravaRegra()` que grava a regra a ser escalonada na classe *Escalonador*.

- **Escalonador**: encapsula os detalhes de escalonamento de leituras das aplicações. Os métodos dessa classe são: `solicitaChecagemConflitoRegras()`, responsável por solicitar a verificação sobre o conflito de regras de leitura das aplicações; `solicitaGravacaoRegra()`, grava a regra após a verificação de conflito da mesma; `monitoraMetrica()`, realiza o monitoramento das métricas informadas nas aplicações, ou seja, realiza as consultas aos componentes da rede; `verificaLimiar()`, constata se as métricas encontram-se em um limiar aceitável; e, por fim, `geraEvento()`, gera eventos quando os parâmetros da rede não se encontram em conformidade aos limiares e a qualidade da rede esperada pela aplicação.

5.2 Diagrama de Sequência do *HomeNetRescue*

Nesta seção apresentamos detalhes complementares à descrição do *HomeNetRescue*. Neste sentido, detalhamos o seu funcionamento em um diagrama de sequência indicado na Figura 5.2. Na parte superior do diagrama são apresentadas as linhas de vida e o ator do serviço. A seguir, descrevemos a sequência das ações e os métodos do *HomeNetRescue* utilizando o mesmo exemplo da aplicação para a detecção de problemas no sinal de transmissão sem fio proveniente de interferências. Nessa seção ela será chamada de *nova aplicação*.

O ator, administrador da rede, é quem interage com o serviço. As aplicações correspondem às interfaces entre o ator e o *HomeNetRescue*. Para esse exemplo, assumimos outras aplicações em execução. Assim, primeiramente, o ator configura a aplicação a ser executada (*nova aplicação*) definindo seus parâmetros. Na *nova aplicação* são configurados os seguintes parâmetros: intervalo de monitoramento (60 s); métricas (SNR e perda de pacotes); ação (alterar a potência de transmissão dos APs); prioridade; protocolo (Ethanol); tempo de monitoramento (indeterminado[1]); e limiares (10 dBm para SNR e 10% taxa de perdas). Esse procedimento corresponde à definição da regra da aplicação e habilita o serviço a realizar o gerenciamento da rede de forma

[1]Enquanto o sistema estiver em execução.

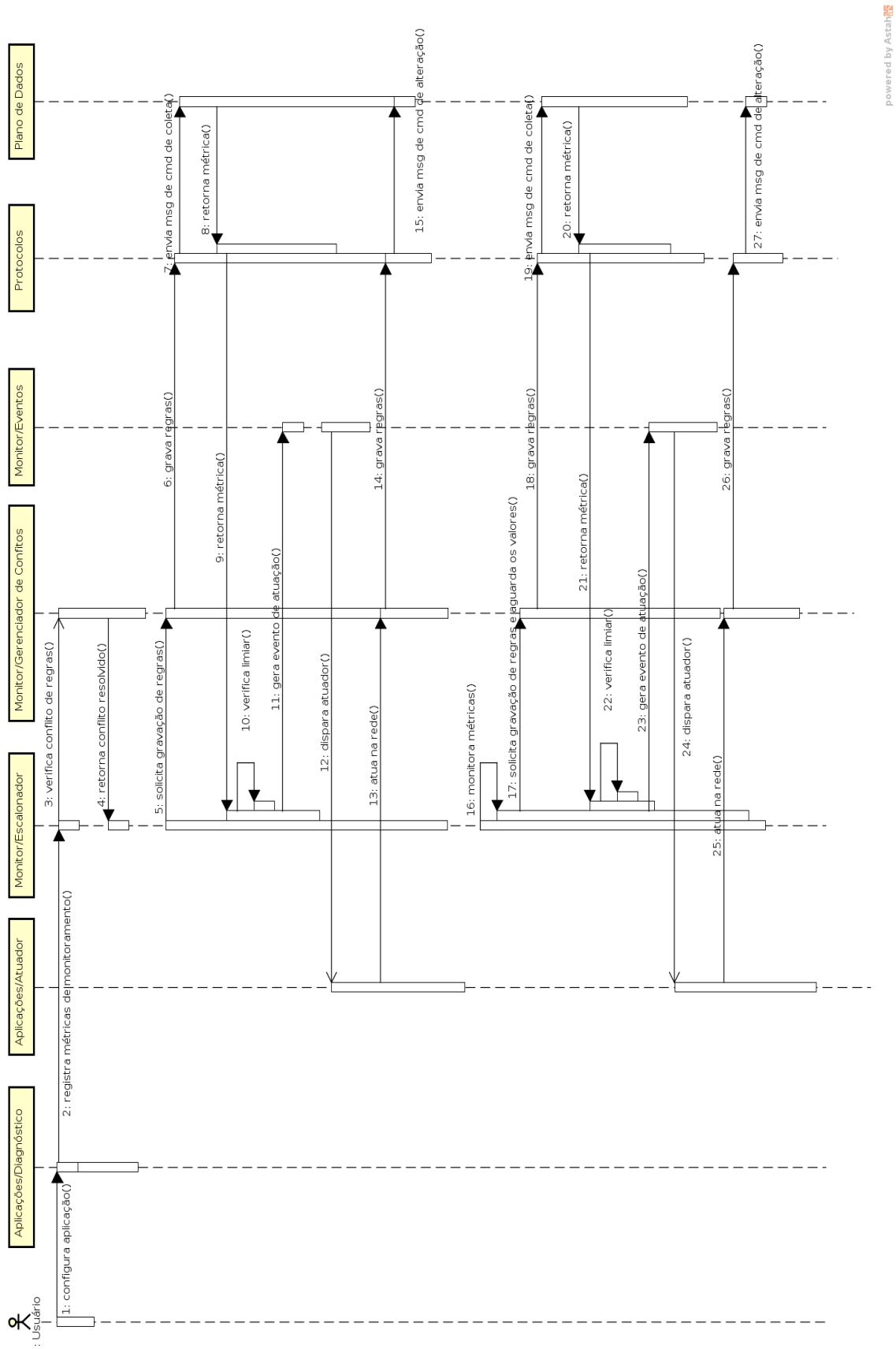


Figura 5.2: Diagrama de sequência do *HomeNetRescue*.

que anormalidades relacionadas aos parâmetros configurados para a aplicação sejam detectadas e solucionadas automaticamente.

Sucessivamente, o módulo *Diagnóstico* registra no módulo *Escalonador* as métricas pré-configuradas que resultam nas regras para o serviço, iniciando assim o monitoramento das métricas definidas na aplicação. Antes do início do monitoramento, o módulo *Escalonador* verifica entre as aplicações em execução a de maior prioridade. Se existir aplicações com prioridades inferiores à informada na *nova aplicação*, estas devem ser sobrescritas pelas regras da nova aplicação, caso contrário, permanecerão em execução paralelamente às outras.

No módulo *Escalonador* são realizadas as chamadas para o módulo *Gerenciador de Conflitos* de modo a verificar se aplicações distintas compartilham de regras similares e se elas conflitam. Podem existir situações, por exemplo, onde uma aplicação solicita o aumento da potência de transmissão para reduzir a taxa de perda, enquanto outra solicita a redução da potência de transmissão para reduzir a interferência entre os APs (nova aplicação). Nesse caso, o módulo *Gerenciamento de Conflitos* é o responsável por resolver a situação, liberando a ação da aplicação com a maior prioridade, neste exemplo, da *nova aplicação*. Após resolver o conflito, o módulo retorna a aplicação com maior prioridade para o módulo *Escalonador* realizar a solicitação do registro da regra com o auxílio do protocolo selecionado.

O protocolo selecionado pelo administrador da rede na aplicação possibilita ao *Gerenciador de Conflitos* realizar o registro das regras. Após o registro das regras, o protocolo realiza a consulta das informações requisitadas ao *Plano de Dados* e retorna o resultado ao módulo *Escalonador* que o analisa verificando se o valor retornado está fora do limiar informado pela aplicação. Em caso afirmativo, o *Escalonador* dispara um evento a ser tratado pelo módulo *Eventos*, que reconhece o evento gerado e dispara o módulo *Atuador*, responsável por interagir com os componentes da rede no intuito de solucionar o problema detectado.

Exemplificando, o módulo *Diagnóstico* permanece em monitoramento e, caso o SNR atinja o limiar de 10 dBm ou a porcentagem de pacotes perdidos seja superior a 10% gera-se um evento de rede. Atingir um limiar dispara o evento correspondente no módulo *Eventos*, que ativa o módulo *Atuador* para realizar a ação configurada na aplicação. A ação de atuação para o evento gerado é realizada através da chamada ao protocolo Ethanol na *nova aplicação*, que envia um comando para a alteração da potência de transmissão dos APs. Caso contrário, o serviço permanece em monitoramento contínuo da rede, ou seja, através das requisições efetuadas pelo *Plano de Controle* (escalonador) ao *Plano de Dados*, o *HomeNetRescue* realiza o monitoramento automático das métricas.

No *HomeNetRescue* o monitoramento da aplicação é interrompido assim que o tempo de monitoramento da métrica expira. Vale ressaltar que antes do módulo *Atuador* realizar as alterações nos parâmetros da rede, primeiramente o módulo *Gerenciador de Conflitos* verifica se a ação a ser tomada não conflita com outra ação do serviço. Dada a inexistência das condições de parada previamente mencionadas, o objetivo do serviço é permanecer em monitoramento contínuo, realizando as requisições em intervalos de tempos variados conforme programação prévia. Assim, o ciclo de procedimentos apresentado é repetido.

Por fim, ressaltamos que as operações realizadas pelos módulos *Atuador* e *Monitor* são armazenadas no módulo de *Logs* para permitir auditorias ou análises sobre o comportamento do serviço, assim como sobre o histórico de ações por ele executadas.

5.3 Implementação do *HomeNetRescue*

A seguir descrevemos os procedimentos e os detalhes técnicos para a implementação do *HomeNetRescue*. De acordo com a Figura 5.3 podemos situar o *HomeNetRescue* em relação à arquitetura Ethanol, proposta por Moura et al. [2015a,b]. Ethanol fornece importantes recursos para o funcionamento do serviço. Através do controlador Ethanol e com o auxílio do protocolo Openflow, o *HomeNetRescue* pode controlar os pontos de acesso sem fio e os comutadores da rede.

Nos pontos de acesso e nas estações do *HomeNetRescue* executamos agentes Ethanol, baseados em linux. O agente em execução no componente é o responsável por responder às requisições realizadas pelo serviço. Incorporado a ele estão ferramentas como $iwconfig$[2] e iw[3] que o auxiliam a realizar as configurações e as coletas de informações das *interfaces* sem fio dos componentes. Ainda, o agente também foi incorporado ao Hostapd (descrito a seguir) executado nos roteadores. A comunicação entre o serviço e os componentes da rede, via agente, é realizada através de uma comunicação segura utilizando Sockets SSL sobre IP.

Para a implementação do *HomeNetRescue* utilizamos a linguagem de programação Python versão 2.7, uma vez que o serviço foi construído como uma aplicação para a arquitetura Ethanol que roda no controlador. O *HomeNetRescue* foi implementado em Python de modo a prover compatibilidade com a linguagem utilizada pelo controlador da rede. Já as funcionalidades nos pontos de acesso e estações são implementadas na linguagem C [Moura et al., 2015b].

[2]https://linux.die.net/man/8/iwconfig
[3]https://linux.die.net/man/8/iw

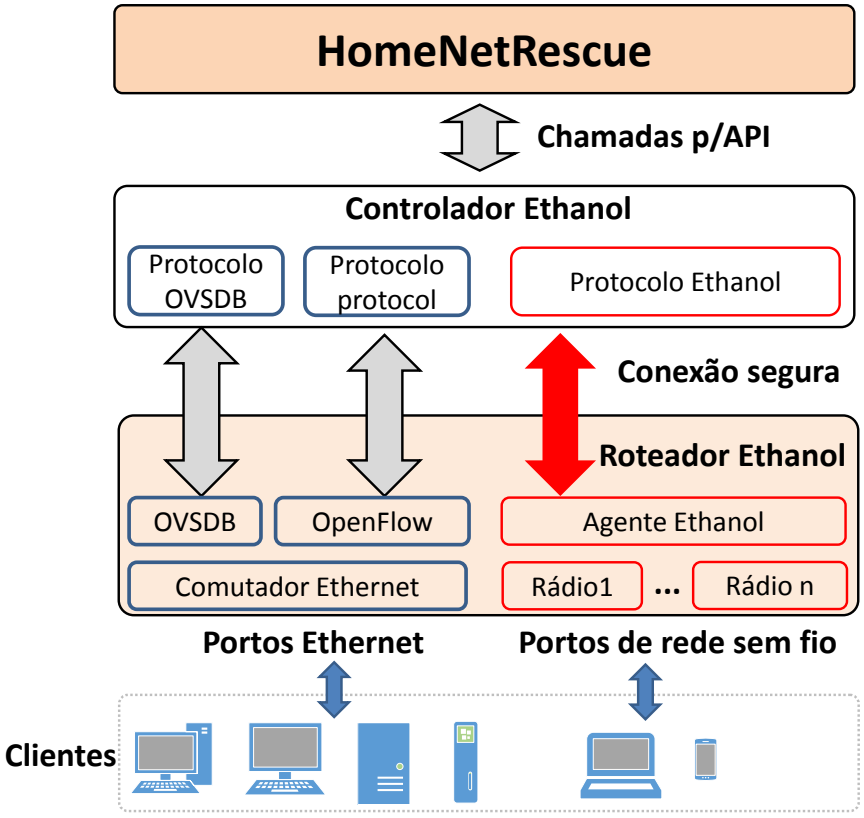


Figura 5.3: HomeNetRescue e Ethanol, adaptado de Moura et al. [2015b].

Tanto o *HomeNetRescue* quanto o controlador Ethanol são executados no controlador POX Dart[4] (versão mais atual do POX). Essa versão do controlador suporta as especificações do protocolo OpenFlow 1.3. Ele foi adotado por permitir rápida prototipação de aplicações, por ser implementado em Python, com código aberto e por ser de fácil configuração e execução em plataformas Linux. Embora o POX não seja o controlador mais adequado para aplicações que demandem alto desempenho, para os estudos de caso, utilizando o *HomeNetRescue* mostrados nesta dissertação, ele não foi um problema. Além disso, nesta dissertação não consideramos que o uso de um controlador crie um ponto de falha principal, pois, técnicas de alta disponibilidade para controladores são consideradas um problema resolvido na literatura [Berde et al., 2014]. Em um ambiente de produção do *HomeNetRescue*, controladores de alta disponibilidade como ONIX [Koponen et al., 2010] e ONOS[5] podem ser utilizados.

Como mencionado, o controlador Ethanol pode usufruir dos recursos do protocolo OpenFlow. Em trabalhos futuros outras versões do protocolo OpenFlow com outras funcionalidades podem ser utilizadas para outras aplicações. Por exemplo, uso

[4]https://github.com/noxrepo/pox/tree/dart
[5]http://onosproject.org/

de múltiplas tabelas no contexto de segurança, entre outras aplicações. Em geral, as mensagens do serviço são trocadas entre o controlador e os agentes Ethanol. O controlador Ethanol permanece em execução enquanto as aplicações do *HomeNetRescue* requisitam informações (métricas) aos agentes em execução nos componentes da rede.

5.4 Plataforma de *hardware* do *HomeNetRescue*

A plataforma de *hardware* do *HomeNetRescue* consiste dos recursos físicos gerenciáveis de uma rede doméstica. Entre eles se destacam o controlador, os pontos de acesso e as estações. Nessa plataforma o HNR foi instalado em computadores que serviram como pontos de acesso na rede doméstica. Para a execução dos experimentos (descritos no próximo capítulo) realizados com o *HomeNetRescue* utilizamos várias combinações de pontos de acesso sem fio e estações sem fio. Cada combinação dos componentes de rede é apresentada na seção de avaliação dos cenários no Capítulo 6.

Devido a fatores como complexidade adicional de programação, tempo para a prototipação e maior complexidade nos mecanismos para a depuração de códigos, decidimos por utilizar computadores pessoais como pontos de acesso ao invés de roteadores como Linksys WRT54GL. No mesmo sentido, buscando estabilidade nos experimentos preferimos por utilizar *hardwares* similares entre si para as estações. Assim, utilizamos como estações os roteadores Linksys WRT54GL ao invés de computadores pessoais. Um roteador *gateway* da rede com maior capacidade computacional poderia executar, além do agente Ethanol, o controlador *HomeNetRescue*, não sendo necessário um componente dedicado da rede para essa finalidade.

O controlador de rede foi configurado em uma máquina virtual. Foi alocada para esta máquina virtual um processador Intel Xeon E312xx @ 2.2 GHz e 4 GB de memória RAM. A máquina virtual executava um sistema operacional Linux Ubuntu versão 14.04 LTS. O controlador utilizava Python versão 2.7 e POX Dart.

5.5 Cenários Avaliados

Nesta seção apresentamos os cenários configurados para a avaliação do *HomeNetRescue* e os algoritmos implementados. A partir desta seção chamamos os pontos de acesso de APs_HNR e as estações de $STAs\_HNR$. Na Subseção 5.5.1 apresentamos um cenário denominado **Mitigando interferências**. O mesmo foi dividido em duas abordagens, uma que descreve uma solução mais simples para o controle da potência de transmissão de um único AP_HNR realizado pelo *HomeNetRescue* (abordagem dis-

tribuída) e outra mais robusta para o controle coordenado da potência de transmissão de múltiplos APs_HNR (abordagem centralizada). Em seguida descrevemos o cenário intitulado **Seleção Dinâmica e Coordenada de Canais**. Ressaltamos que os resultados das avaliações desses cenários são apresentados no Capítulo 6.

5.5.1 Mitigando interferências

Neste cenário apresentamos uma descrição geral para duas situações onde interferências degradam a qualidade da transmissão sem fio dos APs_HNR. Interferências em redes domésticas podem causar perdas e atrasos na entrega de pacotes, retransmissões, diminuição da vazão e podem prejudicar a qualidade dos enlaces sem fio. De acordo com Biswas et al. [2015], as redes sem fio estão piorando devido às interferências. Diante do exposto, o objetivo deste cenário é prover mecanismos para minimizar os efeitos da interferência detectada na transmissão sem fio dos APs_HNR. Para tal, apresentamos duas abordagens com soluções distintas para lidar com o problema de interferências.

Em uma rede doméstica a interferência pode ser gerada por diversos agentes interferentes estando eles situados internamente ou externamente à rede. Como exemplo, podemos citar: sobreposição de canais entre APs de uma mesma rede ou de redes vizinhas, a utilização de fontes interferentes (como microondas, telefone sem fio e babás eletrônicas), barreiras com elementos metálicos ou aquosos entre os roteadores e as estações, entre outras fontes interferentes. No primeiro exemplo, se APs vizinhos compartilham o mesmo canal de comunicação e estão no alcance um do outro, temos que esse tipo de interferência é prejudicial para a comunicação sem fio entre os APs e suas respectivas estações. Não é difícil constatar um cenário similar, pois podemos validar a referida descrição em um condomínio com prédios e apartamentos próximos. Em Mota et al. [2013] foram detectados em média 24,7 pontos de acesso por medição realizada na cidade de Paris.

Nas subseções seguintes descrevemos os algoritmos das duas abordagens utilizadas pelo *HomeNetRescue* para lidar com as interferências de um ambiente de rede doméstica. Uma demonstra os benefícios do serviço em um cenário mais simples com poucos APs, enquanto a outra em um cenário mais elaborado com a coordenação de múltiplos APs. Em ambas as abordagens empregamos a técnica de Controle de Potência de Transmissão ou *Transmission Power Control* (TPC). Ela consiste em um mecanismo utilizado em sistemas de rádio comunicação para dinamicamente controlar a potência de transmissão do rádio transmissor de modo a garantir uma qualidade mínima de um enlace e reduzir interferências em outros dispositivos [Correia et al.,

2007].

5.5.1.1 Controle de Potência de Transmissão sem Coordenação

No Algoritmo 1 descrevemos os procedimentos realizados pelo *HomeNetRescue* para realizar o monitoramento do problema de interferência na rede do serviço. A partir disso objetivamos proporcionar melhorias na qualidade do enlace de um AP_HNR. Nesse caso, a interferência na rede reflete na quantidade de pacotes perdidos na interface de rede sem fio do AP_HNR, pois, quanto maior a interferência presenciada maior a quantidade de pacotes perdidos. Assim, o procedimento recebe os parâmetros `threshold`, `interval` e `time` de controle da aplicação. `Threshold` corresponde ao limiar de perda de pacotes, `interval` corresponde ao intervalo de monitoramento e `time` ao tempo de monitoramento. Em sequência, o serviço aguarda o controlador receber a conexão do AP_HNR (linha 2). Nesse cenário foi considerado somente um AP_HNR. Após o AP_HNR se conectar, uma referência para o seu objeto é armazenada em uma lista de `APs`.

Algoritmo 1 Controle de Potência de Transmissão

```
1:  procedure powerControl(threshold, interval, time)
2:      wait_ap_connections()
3:      APs ← connected_aps()                          ▷ return a set of connected APs
4:      repeat
5:          for each ap ∈ APs do
6:              current_pct_lost ← ap.vaps.statistics          ▷ collects AP statistics
7:              if old_value_dropped_ap = NaN then      ▷ deals with the first reading
8:                  old_value_dropped_ap ← current_pct_lost
9:              diff_lost ← current_pct_lost − old_value_droppeds
10:             old_value_dropped_ap ← current_pct_lost
11:             actionType ← checkThreshold(ap, diff_lost, threshold)
12:                                                    ▷ test condition for event
13:             generateEvent(ap, actionType)      ▷ given condition, generates event
14:         sleep(interval)
15:         time = time − interval
16:     until time <= 0
```

Repetidamente, enquanto o tempo de monitoramento não expirar, a aplicação requisita ao AP_HNR estatísticas sobre a quantidade de pacotes perdidos (linha 6) em sua interface sem fio. A cada iteração de monitoramento, ou seja, a cada leitura realizada, o serviço realiza o cálculo da quantidade de pacotes perdidos através do valor da leitura anterior menos a quantidade de pacotes perdidos da leitura atual (linha 9). O condicional das linhas 7-8 garante que as variáveis utilizadas sejam zeradas no início

da execução do serviço. Isso foi realizado devido à leitura dessa métrica ser realizada diretamente do *driver* da placa de rede do AP_HNR. O *driver* armazena os valores absolutos desde a inicialização da interface de rede, isto é, enquanto o dispositivo estiver ativo ele contabiliza incrementalmente a quantidade de pacotes perdidos. Desta forma ao efeituar a subtração da linha 9 obtemos a perda no intervalo entre as medições.

Como referência para a próxima leitura a ser utilizada nos cálculos realizados pelo serviço, na linha 10, a quantidade de pacotes perdidos atual é atribuída à variável `old_value_dropped` para que esta seja utilizada na iteração seguinte. A função `checkThreshold()` realiza a verificação da quantidade de pacotes perdidos (`diff_lost`) e se ela encontra-se fora do limiar aceitável. O resultado dessa verificação é retornado em `actionType` que corresponde ao tipo de ação a ser realizada. Posteriormente, a função `generateEvent()` registra e gera o evento para a atuação do serviço. Para esse cenário a função `executaAcao()`, omitida no algoritmo, altera a potência de transmissão do AP_HNR para o seu valor máximo de modo a diminuir a quantidade de pacotes perdidos. Por fim, o serviço cumpre o intervalo de monitoramento especificado na linha 15.

5.5.1.2 Controle de Potência de Transmissão com Coordenação

Embora seja baseado no Algoritmo 1, no Algoritmo 2 apresentamos outra abordagem para lidar com o problema de interferências. O objetivo deste cenário se concentrou em gerenciar coordenadamente e dinamicamente a potência de transmissão de vários HNR_APs. A solução consiste em receber os parâmetros `threshold`, `interval`, `time` e `granularity` de controle da aplicação. Os três primeiros correspondem à mesma descrição apresentada no cenário anterior enquanto o último corresponde ao parâmetro para a alteração gradual da potência, empiricamente definido em 3 dBm. Em sequência, a aplicação aguarda o controlador receber as conexões dos APs_HNR (linha 2). Após os APs_HNR se conectarem criam-se os objetos de cada AP_HNR que são armazenados em uma lista de `APs`.

Antes de começar o monitoramento da rede, primeiramente, a potência de transmissão de todos os APs_HNR foi configurada para a menor capacidade (linhas 4 e 5). Esse procedimento foi adotado para minimizar a interferência exercida pela aplicação do *HomeNetRescue* nas redes vizinhas inicialmente. A aplicação foi implementada para melhor se ajustar ao ambiente, preocupando-se em garantir a qualidade de sua rede sem desconsiderar as redes vizinhas. Logo, iniciar os APs_HNR com a potência mínima proporciona que a aplicação ajuste dinamicamente a potência de transmissão de cada AP_HNR conforme a sua localidade, pois podem existir APs_HNR que

presenciam diferentes níveis de interferência, consequentemente, que exigem diferentes potências.

Algoritmo 2 Coordenação de Potência de Transmissão

```
1:  procedure TPC(threshold, interval, time, granularity)
2:      wait_ap_connections()
3:      APs ← connected_aps()                          ▷ return connected APs list
4:      for ap ∈ APs do
5:          ap.vaps.txpower ← 1                ▷ initialize APs with mininimum txpower
6:      repeat
7:          for ap ∈ APs do                    ▷ scroll through a list of connected APs
8:              throughput ← calculate_throughput(ap)
9:              tx_p ← ap.vaps.txpower                      ▷ return current txpower
10:                                  ▷ test condition, and returns type of action
11:             if throughtput > threshold then
12:                 actionType ← tx_p − granularity
13:             else if throughtput < threshold then
14:                 actionType ← tx_p + granularity
15:             else
16:                 actionType ← 0
17:             generateEvent(ap, actionType)              ▷ trigger event to action
18:             sleep(interval)
19:         time = time − interval
20:     until time <= 0
```

A aplicação apresentada objetiva encontrar uma solução global viável em uma rede. Para tal, a aplicação monitora todos os pontos de acesso da sua topologia repetidas vezes enquanto o tempo de monitoramento não expirar (linha 20). A partir da linha 7, para cada AP_HNR, o serviço realiza um conjunto de procedimentos e a coordenação dos APs_HNR se dá através desses procedimentos. Desta maneira, para cada AP_HNR calcula-se primeiramente a sua vazão. Essa é a métrica utilizada nesse cenário. Sua escolha se deu por ela refletir diretamente na qualidade da transmissão de *streaming* de vídeos em uma estação, na taxa de *download*, no atraso, entre outras métricas de rede. Por conseguinte, por afetar as métricas mencionadas ela também reflete negativamente na qualidade de experiência dos clientes.

O cálculo de vazão é realizado passivamente obtendo-se dos APs_HNR informações sobre a quantidade de *bytes* transmitidos em suas respectivas interfaces sem fio. O resultado da operação é retornado em `throughput`. Posteriormente, são realizadas as verificações sobre qual situação a vazão recebida se enquadra. São verificadas três possibilidades para a vazão de forma que ela seja superior, idêntica ou inferior ao limiar. Para cada verificação existe um tipo de ação diferente, sendo esta definida na

variável `typeAction`. Após as verificações na função `generateEvent()` é registrado e gerado um evento de atuação na rede.

Em casos de eventos onde a vazão encontra-se inferior ao limiar, o evento a ser gerado corresponde ao aumento da potência de transmissão em 3 dBm. Em casos onde o limiar seja superior, o evento corresponde à diminuição da potência em 3 dBm. Por fim, para eventos com equidade entre vazão e limiar não são executadas alterações, pois a ação é nula. Esse processo permite a alteração dinâmica e gradual da potência de transmissão dos APs_HNR, ou seja, aumento, diminuição ou permanência da mesma. Ressaltamos que cada AP_HNR possui um limite especificado pelo *driver* da placa de rede sem fio para a alteração da potência. Em geral, para os componentes usados em nossa rede, foi possível alterar a potência de transmissão em um intervalo variando entre 1 e 27 dBm.

Por fim, a aplicação aguarda o tempo definido na variável `interval` para realizar o monitoramento do próximo AP_HNR e esse ciclo se repete para todos os pontos de acesso até o tempo de monitoramento expirar.

5.5.2 Seleção Dinâmica e Coordenada de Canais

Os cenários anteriores nos permitiram concluir que a alocação de um mesmo canal para os APs_HNR colabora para exercer interferências adicionais em roteadores de um mesmo ambiente ou próximos entre si. Como as redes domésticas não são imunes a isso e possuem diversos dispositivos interferentes em seu contexto, estas também são susceptíveis a sofrer desse problema. Para viabilizar melhorias da qualidade de transmissão sem fio entre os APs_HNR, nesse cenários propomos uma solução onde o serviço busca otimizar a distribuição de canais entre os APs_HNR de forma dinâmica e coordenada.

O problema clássico de k-coloração de grafos pode ser reduzido a este problema de seleção de canais, portanto se $P \neq NP$, não existe solução polinomial para o problema. Desta forma optamos por propor uma abordagem gulosa para a seleção dos canais. Nosso algoritmo prioriza os APs com maior tráfego selecionando para estes o canal que está menos ocupado. Nessa abordagem o algoritmo não encontra uma sub-estrutura ótima para o problema por demandar de uma enumeração de todas as possibilidades para encontrar a solução ótima e pela complexidade deste algoritmo.

O Algoritmo 3 corresponde à solução implementada para a seleção coordenada de canais dos APs_HNR. O objetivo consiste em prover uma solução global onde o canal dos roteadores da rede sejam configurados de forma a não se interferirem ou se interferirem minimamente. Dessa forma, inicialmente definimos uma lista para armazenar

a carga dos APs_HNR (linha 2). Para cada AP_HNR são coletadas informações sobre seus respectivos canais (linha 4). Entre as informações de cada canal estão o tempo de recebimento (`receive_time`), o tempo de transmissão (`transmit_time`) e o tempo ativo (`active_time`) do referido AP. Estas informações possibilitam à aplicação calcular a carga vigente do AP_HNR (linha 8) para então armazená-la em `aps_load` (linha 9). Na linha 10, utilizamos uma lista de prioridade máxima com heap (`heap<value,key>`) onde `value` representa o objeto a ser recuperado do heap e `key` o peso para a sua ordenação. Na linha 11, o algoritmo preenche o heap considerando o percentual de ocupação de carga dos APs_HNR. Desta maneira, a raiz do heap conterá o AP_HNR com a maior ocupação de carga, isto é, maior tempo de utilização do canal.

Algoritmo 3 Seleção de Canais

```
1:  procedure Choose_channels(APs)
2:      aps_load ← ∅
3:      for each ap ∈ APs do
4:          info ← ap.channel_info()              ▷ get current channel information
5:          receive_time = info.receive_time
6:          transmit_time = info.transmit_time
7:          active_time = info.active_time
8:          w ← (receive_time + transmit_time)/active_time
9:          aps_load.add(ap, w)                   ▷ add the AP to a max heap
10:     h ← heap.MaxPriorityQueue()
11:     h.build_heap(APs, aps_load)
12:     channels ← ∅
13:     repeat
14:         ap ← h.popMax()
15:         acs ← ap.radio.acs                    ▷ acs is a factor list indexed by channel
16:         new_channel ← arg min_channel acs[channel]
17:                                               ▷ select the best channel for this AP
18:         generateEvent(ap, new_channel)
19:     until h = ∅
```

Nas linhas 13-19, enquanto houver elementos no heap, o algoritmo remove da raiz da estrutura de dados o AP_HNR com a maior ocupação de carga, ou seja, aquele que mais transmite e recebe pacotes de seus clientes. O processo de seleção do canal ocorre nas linhas 15 e 16. Na linha 15, o algoritmo consulta o tempo de ocupação de cada canal disponível no AP_HNR em foco. Essa informação é obtida do *driver* da interface de rede sem fio que efetua varreduras em cada canal. Ele retorna um fator de interferência. Esse fator representa a taxa do tempo de ocupação observado sobre o tempo de transmissão gasto no canal. Para cada canal, o fator de interferência é

calculado considerando a Equação 5.1. O Hostapd utiliza essa métrica para realizar a *Automatic Channel Selection* (ACS)[6], que permite ao `ap` selecionar o melhor canal ao ser ativado. Assim, empregamos a mesma função na nossa solução.

$$ACS = 10^{\frac{chan\_nf}{5}} + \frac{busy\_time - tx\_time}{active\_time - tx\_time} \times 2^{10^{\frac{chan\_nf}{10}} + 10^{\frac{band\_min\_nf}{10}}} \quad (5.1)$$

Essa equação retorna um valor onde a sua magnitude indica a quantidade de interferência percebida em cada canal. O menor valor retornado corresponde ao melhor canal a ser adotado (linha 16), ou seja, o canal selecionado é aquele com o menor fator, e consequentemente com menor interferência. Por fim, na função `generateEvent()` (linha 18), o evento para a alteração para o canal selecionado é gerado.

5.6 Resumo

Nesse capítulo apresentamos detalhes de projeto, de implementação e de algumas aplicações do *HomeNetRescue*. Citamos as linguagens de programação, as ferramentas e a plataforma de *hardware* utilizada para a sua avaliação. Descrevemos também os diagramas (classes e sequência) propostos. Descrevemos ainda alguns cenários, apresentando os algoritmos que nos permitiram avaliar o serviço em um ambiente real. No próximo capítulo apresentamos os resultados obtidos a partir do protótipo implementado. Através dele foi possível realizar os experimentos e verificar o funcionamento do serviço, assim como identificar os benefícios de sua utilização em ambiente real.

[6]https://wireless.wiki.kernel.org/en/users/documentation/acs

Capítulo 6

Experimentos e Avaliações

Para realizarmos os testes e as avaliações com o *HomeNetRescue* construímos um protótipo que implementa parcialmente as funcionalidades da arquitetura proposta no Capítulo 4. Nesse capítulo discutimos e apresentamos os resultados obtidos através da aplicação do serviço em ambiente real. Os resultados obtidos com o *HomeNetRescue* são comparados com os resultados de uma solução sem a sua utilização. A seguir são apresentados os resultados dos cenários descritos no Capítulo 5, são eles: *a*) Controle de Potência de Transmissão sem Coordenação; *b*) Controle de Potência de Transmissão com Coordenação; e, *c*) Seleção Dinâmica e Coordenada de Canais.

O capítulo está organizado da seguinte maneira. Na Seção 6.1 descrevemos a metodologia e as avaliações realizadas com o *HomeNetRescue*. Nas Seções 6.2, 6.3 e 6.4 discutimos sobre os cenários Controle de Potência de Transmissão sem Coordenação, Controle de Potência de Transmissão com Coordenação e Seleção Dinâmica e Coordenada de Canais, respectivamente. Em cada seção descrevemos os componentes de *hardware* empregados, os resultados e as discussões de cada cenário. Por fim, resumimos o capítulo na Seção 6.5.

6.1 Descrição da Montagem dos Experimentos

O *HomeNetRescue* foi avaliado em um protótipo dividido em três cenários que simulam problemas presenciados em redes domésticas. Embora o *HomeNetRescue* tenha sido modelado para lidar com problemas que podem ocorrer nas camadas 2-5 (modelo TCP/IP), nessa dissertação avaliamos principalmente eventos que podem ocorrer na camada de enlace (camada 2), mais especificamente nos enlaces sem fio. Desta maneira, o foco da avaliação concentrou-se nas anormalidades que podem colaborar para a degradação da qualidade do sinal. Assim, foram apresentadas soluções visando solu-

cionar problemas que provoquem variações em fluxos sem fio devido às interferências, localidade dos clientes (baixa cobertura do sinal), entre outras causas.

O tráfego de dados gerado nos experimentos foi obtido através da ferramenta cliente-servidor *bwping*[1] (ferramenta similar ao *iperf*) [Moura et al., 2017]. A ferramenta utiliza o protocolo de transporte UDP e é capaz de fornecer estatísticas de vazão, atraso, *jitter*, quantidade de pacotes enviados e recebidos, entre outras. Além disso, a ferramenta permite configurar o tamanho dos pacotes de envio e o intervalo entre requisições. Para os experimentos a ferramenta foi configurada para gerar tráfego no limite da capacidade da transmissão sem fio, dados os componentes empregados.

Consideramos como métricas o desempenho das estações, dados pelas métricas vazão, atraso e *jitter*. Nos APs_HNR consideramos a porcentagem de utilização de CPU e memória, o tempo de resposta a eventos, a saber, tempo entre a detecção do problema e a gravação da regra no componente.

Visando a confiabilidade dos resultados, para cada experimento descrito a seguir realizamos leituras periódicas, a cada segundo, das métricas vazão, atraso e *jitter*. Para o cenário 1, analisamos aproximadamente 500 leituras que representam 33 repetições. Os resultados apresentados correspondem à média das repetições. Para os cenários 2 e 3, analisamos aproximadamente 1300 leituras que representam 25 repetições. Devido ao baixo valor do desvio padrão a quantidade de repetições foi suficiente para representar os gráficos das seções seguintes. Além da média das leituras das métricas obtidas das estações, calculamos também o desvio padrão e o intervalo de confiança considerando um coeficiente de confiança $(1-\alpha) = 95\%$.

6.2 Cenário 1: Controle de Potência de Transmissão sem Coordenação

Os detalhes de implementação e o algoritmo para esse cenário foram apresentados na Seção 5.5.1.1. A seguir, descrevemos a configuração do ambiente e os componentes empregados para a avaliação do referido cenário. Na próxima seção apresentamos os gráficos e as discussões sobre seus resultados.

O cenário representa uma rede doméstica com um único ponto de acesso e foi organizado conforme ilustrado na Figura 6.1. Para representar exemplos de interferências presenciadas em redes domésticas, inserimos interferências sintéticas no ambiente com o auxílio de duas fontes interferentes. Uma fonte gera ruídos no ambiente utilizando um *hardware* específico denominado *Universal Software Radio Peripheral* ($USRP$) e a

[1] https://github.com/h3dema/bwping-udp

outra é um AP interferente (INT_{AP}), representando um ponto de acesso instalado em outro domicílio onde o ponto de acesso comunica-se com uma estação no mesmo canal do AP_HNR1.

O objetivo desta disposição consiste em induzir na rede do *HomeNetRescue* um alto nível de interferência, simulando assim ruídos de aparelhos domésticos (por exemplo, aparelhos micro-ondas, telefones sem fio e babás eletrônicas, etc), bem como a interferência entre APs devido à sobreposição de canais do IEEE 802.11.

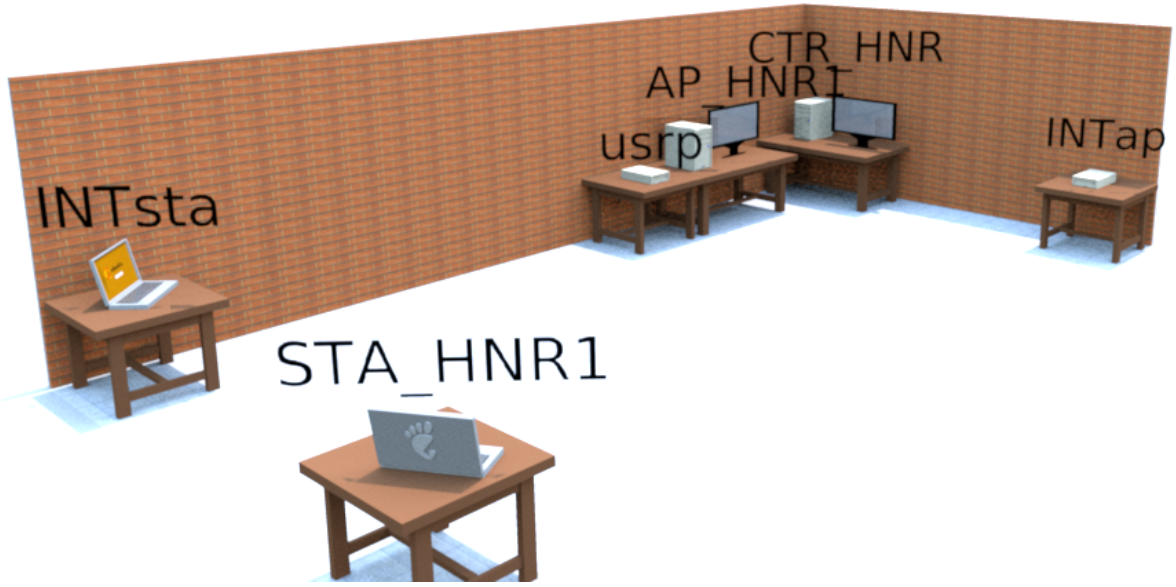


Figura 6.1: Ilustração do cenário para o cenário 1.

Nesse cenário o AP_HNR1 foi configurado com uma potência de transmissão de 1 dBm. O INT_{AP} foi configurado em 18 dBm tendo uma antena acoplada com ganho de 2 dBm. A $USRP$ esteve acoplada com uma antena de 6 dBm e foi configurada para emitir sinais gaussianos com 100% de ganho na mesma frequência do AP_HNR1.

6.2.1 Componentes empregados

Para a execução desse experimento com o *HomeNetRescue* utilizamos a seguinte combinação dos componentes:

- **Ponto de Acesso** AP_HNR1: com processador: Intel Core 2 Duo CPU E7200 @ 2.53GHz; memória: 2 GB; placa de rede: *Qualcomm Atheros AR9285 Wireless Network Adapter 802.11bgn (PCI)*; *driver* de rede: ath9k; e, sistema operacional: Linux Ubuntu versão 14.04 LTS em conjunto ao *daemom* do hostapd versão 2.6;

- **Ponto de Acesso** INT_{AP}: roteador sem fio Linksys WRT54GL com o OpenWrt versão Backfire 10.03.x.

- **Estação** STA_HNR1: com processador: Intel Core 2 Duo CPU E7500 @ 2.93GHz; memória: 2 GB; placa de rede: *Atheros Communications, Inc. AR9170 802.11bgn (USB) driver* de rede: carl9170; e,

- **Estação** INT_{STA}: com processador: Intel Core2 CPU 6300 @ 1.86GHz; memória: 2 GB; placa de rede: *Atheros Communications, Inc. AR9170 802.11bgn (USB) driver* de rede: carl9170;

  As estações executam o Sistema Operacional Linux Ubuntu versão 14.04 LTS.

- $USRP$: representado por uma placa *Universal Software Radio Peripheral* (USRP) B210[2] (GNU Radio) com uma placa filha FE-TX2;

6.2.2 Resultados e Discussões

Nas figuras 6.2, 6.4 e 6.5 utilizamos diagramas de caixa para exibir os resultados. Nos diagramas de caixa o eixo vertical representa a variável a ser analisada e o eixo horizontal os fatores de interesse, indicado nas legendas de cada figura. O diagrama de caixa permite visualizar os quartis (25%, mediana e 75%). Em cada um, a caixa é delimitada na parte superior pelo quartil Q_3 (distribuindo 25% dos dados acima) e na parte inferior pelo Q_1 (distribuindo 25% dos dados abaixo). O traço interno indica a mediana (distribuindo 50% dos dados abaixo e 50% acima). As duas linhas na horizontal que se estendem a partir da caixa são os bigodes. O intervalo interquartílico (II) é dado em função $II = Q_3 - Q_1$. Por fim, o limite superior (LS) é dado em função de $LS = Q_3 + 1,5 * II$, enquanto que o limite inferior (LI), em função de $LI = Q_1 - 1,5 * II$. Ainda, desconsideramos valores discrepantes ou *outliers*.

Na Figura 6.2 exibimos o impacto das interferências no ambiente de testes e como essas afetam as transmissões sem fio. Demonstramos a relação entre o número de pacotes perdidos e a vazão percebida no AP_HNR1 em quatro situações: *a*) o AP_HNR1 não esteve sob influência da interferência induzida; *b*) quando a interferência foi inserida somente pelo tráfego do INT_{AP} ou *c*) somente pela $USRP$; e, *d*) quando ambos geraram interferências simultaneamente. Baseado nisso, conforme esperado, observamos que à medida que acrescentamos novas fontes de interferência a quantidade de pacotes perdidos por vazão aumenta, validando que as interferências comprometem o desempenho da rede.

[2]https://www.ettus.com/product/details/UB210-KIT

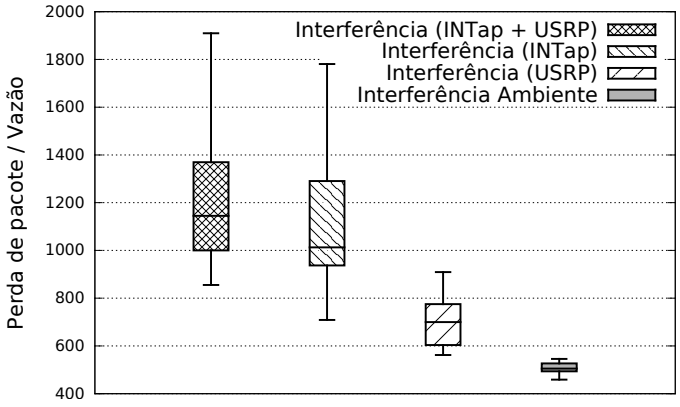


Figura 6.2: Razão entre número de pacotes perdidos pela vazão percebida no AP.

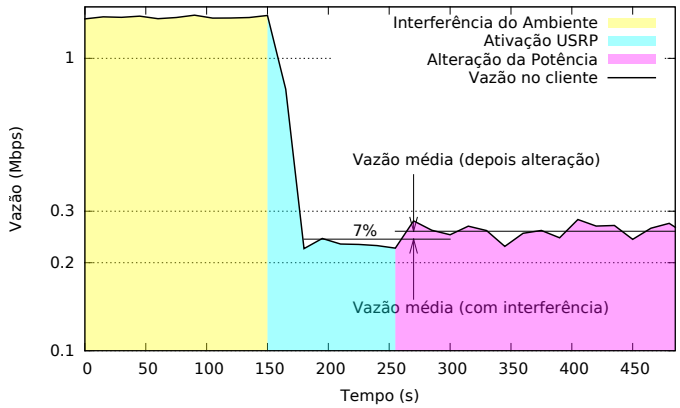


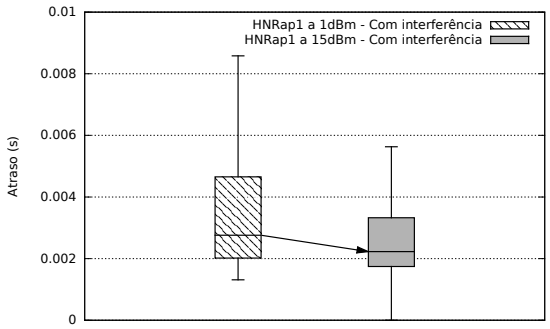
Figura 6.3: Exemplo de execução do *HomeNetRescue* com ganho de 7% para um fluxo frente a interferência sintética.

Na Figura 6.3 ilustramos a variação da vazão durante o experimento. Na fase inicial (em amarelo entre 0 e 150 s), o AP_HNR1 esteve com uma potência de transmissão de 1 dBm. Em 150 s, acrescentamos a interferência do $USRP$. Posteriormente, aguardamos alguns segundos para a vazão estabilizar. A figura mostra a vazão média depois da estabilização, no período de 170 a 250 s. Em 250 s, a aplicação acionou o controlador por ter detectado um problema no fluxo (por atingir o limiar de 248 pacotes perdidos) e aumentou a potência do AP_HNR1 para 15 dBm (capacidade máxima). Neste caso, o HNR conseguiu obter uma melhora de 7% na vazão percebida pelo cliente. Além disso, destacamos que a aplicação do HNR também é capaz de reduzir dinamicamente a potência, assim como utilizar-se de outras métricas, o que viabiliza um gerenciamento da rede com mais justiça. Por exemplo, reduzir a interferência nos APs vizinhos uma vez que a vazão alcançada atenda à necessidade da aplicação.

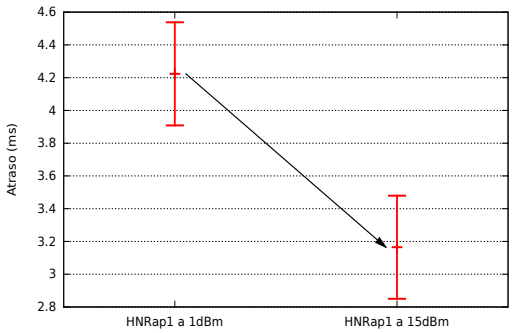
Em termos de desempenho de *hardware*, ao executar o *HomeNetRescue* com a aplicação, o processamento realizado na CPU do AP_HNR1 alcançou no máximo 2%

da capacidade total do dispositivo enquanto a utilização da memória não ultrapassou 3% (cerca de 40 MB). Esses valores indicam que o serviço demanda poucos recursos dos dispositivos e poderia, por exemplo, ser executado em um dispositivo doméstico. A título de exemplo, um roteador modesto para os padrões atuais, como o TP-Link TL-WR2543ND, possui processador 400 MHz, 64 MB de memória RAM e 8 MB de memória *flash*. Adicionalmente, as alterações de potência de transmissão, quando realizadas, gastaram no máximo 29 ms entre a detecção do problema e o envio da regra ao componente.

Na Figura 6.4(a) exibimos os valores de atraso (em segundos) obtidos no experimento, medidos no cliente. Na barra da esquerda é mostrado o atraso observado pelo cliente com o AP_HNR1 configurado para a potência de 1 dBm sob a influência do INT_{AP}. A barra à direita mostra a situação quando a aplicação aumenta a potência do AP_HNR1 para 15 dBm. Pode-se notar na figura (seta) que há uma redução do atraso, resultado da aplicação do serviço. Na Figura 6.4(b), representamos o intervalo de confiança do atraso médio para a situação com o AP_HNR1 funcionando na potência de 1 dBm e na situação depois, da alteração para 15 dBm. Na Tabela 6.1 exibimos o intervalo de confiança da média do atraso e do *jitter*. Dessa maneira, não existe sobreposição entre os intervalos de confiança da média, portanto as médias são independentes. Logo, os valores das médias para 1 dBm são piores dos que os valores para 15 dBm.



(a) Atraso medido pelo cliente



(b) Atraso médio com intervalo de confiança de 95% com interferência

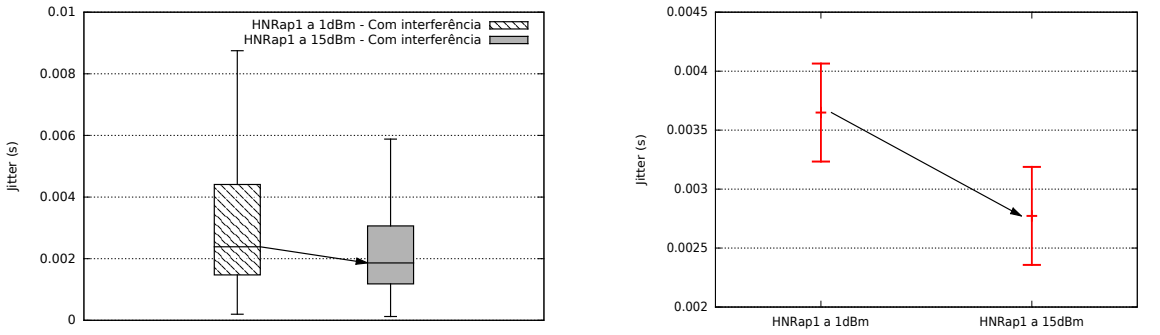
Figura 6.4: Atraso na estação do experimento.

Na Figura 6.5(a), apresentamos os valores de *jitter* obtidos no experimento. Na barra à esquerda exibimos o *jitter* observado pelo cliente com o AP_HNR1 configurado para a potência de 1 dBm sob a influência do INT_{AP}. A barra à direita mostra a situação quando o serviço aumenta a potência do AP_HNR1 para 15 dBm, mantidas as outras condições constantes. Verifica-se que há uma redução do *jitter*. O intervalo

Tabela 6.1: Média dos resultados para o cliente com interferência no cenário 1.

Cenário	Métrica	Valor
HNR @ 1 dBm	atraso	4.218 ± 0.316 ms
	jitter	3.833 ± 0.329 ms
HNR @ 15 dBm	atraso	3.160 ± 0.218 ms
	jitter	2.773 ± 0.229 ms

de confiança do *jitter* médio é mostrado na Figura 6.5(b). Como os intervalos não se sobrepõem, logo as médias são independentes. Dessa forma, vemos que a aplicação conseguiu neste caso melhorar o *jitter* dinamicamente, ao modificar a potência do ponto de acesso, quase obtendo uma situação próxima ao cenário sem interferência. Contudo, como as diferenças são pequenas, acreditamos que este ponto precisa ser estudado mais profundamente em trabalhos futuros, aumentando o número de execuções ou reduzindo o intervalo de confiança.



(a) *Jitter* medido pelo cliente

(b) *Jitter* médio com intervalo de confiança de 95% com interferência

Figura 6.5: *Jitter* na estação do experimento.

6.3 Cenário 2: Controle de Potência de Transmissão com Coordenação

Os detalhes de implementação e o algoritmo para esse cenário foram apresentados na Seção 5.5.1.2. Nessa seção primeiramente descrevemos a configuração do ambiente e os componentes empregados para a avaliação do referido cenário. Em sequência, apresentamos os gráficos e as discussões sobre seus resultados. Esse experimento demonstra também como as interferências no ambiente afetam as transmissões sem fio e os parâmetros de rede.

Este cenário é um complemento da solução apresentada no cenário descrito anteriormente. Seu objetivo concentrou em gerenciar coordenadamente e dinamicamente a potência de transmissão de vários APs_HNR. Empregar esse tipo de solução proporciona benefícios como aumento no desempenho e redução dos níveis de interferência gerados entre APs_HNR se comparado com uma solução sem coordenação. Na abordagem com coordenação buscamos encontrar a melhor solução global de potência para os dispositivos. Na abordagem sem coordenação busca-se a melhor solução local. Podem existir casos onde a melhor solução local prejudique outros APs no mesmo alcance, sendo a solução global mais adequada para a rede onde vários APs podem ser controlados.

A disposição do cenário foi configurada conforme apresentamos na Figura 6.6. Nesse cenário a interferência também colabora para a degradação da qualidade do sinal sem fio. Para tal, utilizamos duas USRPs como agentes interferentes, indicadas como $USRPs(01-02)$, quatro $APs\_HNR(1-4)$, e quatro $STAs\_HNR(1-4)$ na referida figura. Em adição, no ambiente continham também variadas fontes interferentes e obstáculos, pois além da própria interferência dos APs_HNR (configurados no mesmo canal), ao redor existiam mais de 35 roteadores transmitindo pacotes, além da presença de paredes, metais (*racks* de servidores) e movimentação intensa de pessoas. Dado o cenário, o cenário se concentrou em realizar a melhor escolha do parâmetro potência de transmissão dos APs, através do *HomeNetRescue*, visando melhorar dinamicamente a vazão nos APs por ele controlados.

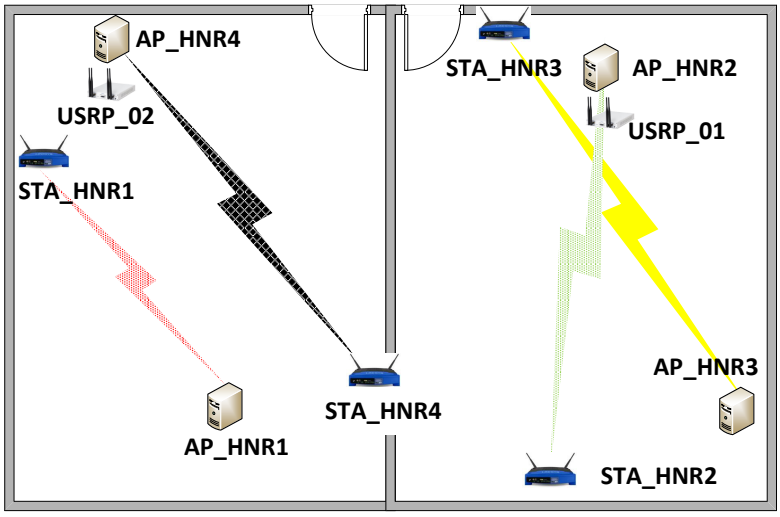


Figura 6.6: Configuração do cenário para os cenários 2 e 3.

6.3.1 Componentes empregados

Para a execução desse experimento com o *HomeNetRescue* utilizamos outra combinação de componentes conforme descrição a seguir:

- **Ponto de Acesso** AP_HNR1: com processador: Intel Pentium CPU 3825U @ 1.90GHz; memória: 4 GB; placa de rede: *Intel Corporation Wireless 3160 802.11abgn (PCI)*; *driver* de rede: iwlwifi;

- **Ponto de Acesso** AP_HNR2: com processador: Intel Core2 CPU 6300 @ 1.86GHz; memória: 2 GB; placa de rede: *Atheros Communications, Inc. AR9170 802.11bgn (USB) driver* de rede: carl9170;

- **Ponto de Acesso** AP_HNR3: com processador: Intel Core2 Duo CPU E7200 @ 2.53GHz; memória: 2 GB; placa de rede: *Qualcomm Atheros AR9285 Wireless Network Adapter 802.11bgn (PCI)*; *driver* de rede: ath9k;

- **Ponto de Acesso** AP_HNR4: com processador: Intel Core i7-4500U CPU @ 1.80GHz; memória: 8 GB; placa de rede: *Qualcomm Atheros AR9485 Wireless Network Adapter 802.11bgn (PCI)*; *driver* de rede: ath9k;

  Todos os pontos de acesso executam o Sistema Operacional Linux Ubuntu versão 14.04 LTS. A implementação do ponto de acesso 802.11 é feita pelo *daemon* do hostapd versão 2.6.

- **Estações** $STA\_HNR1 - 4$: quatro roteadores sem fio Linksys WRT54GL com o OpenWrt versão Backfire 10.03.x atuando como estações.

- $USRP\_01 - 02$: duas placas *Universal Software Radio Peripheral* (USRP) B210 (GNU Radio) com uma placa filha FE-TX2;

6.3.2 Resultados e Discussões

Nesta seção discutimos sobre dois tipos de avaliações realizadas com a execução da aplicação do cenário com o *HomeNetRescue*. A primeira avaliação apresenta os benefícios sobre as métricas consideradas em cada estação em função da execução da aplicação para a alteração dinâmica e coordenada da potência de transmissão do *HomeNetRescue*. A segunda parte mostra o benefício global para a rede da aplicação do *HomeNetRescue*. Com isto queremos mostrar que a utilização do *HomeNetRescue* de forma coordenada melhora não somente os valores para cada rede controlada (par

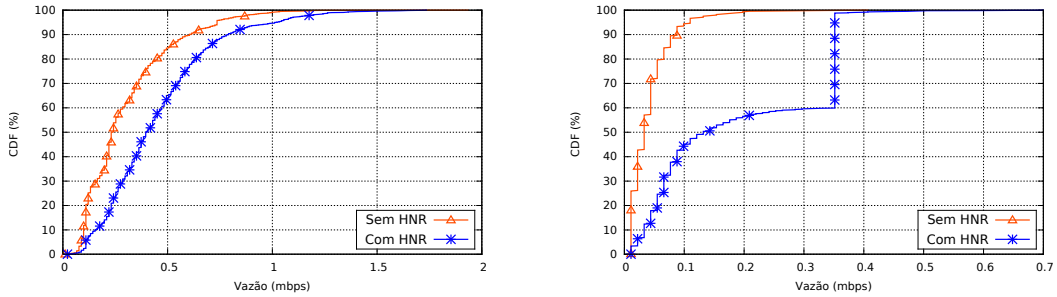
AP–estação) como também para toda a rede, uma vez que a visão global do controlador permite evitar (ou pelo menos minimizar) a interferência cruzada entre as redes controladas pelo *HomeNetRescue*. Em ambas as avaliações comparamos nossos resultados com uma solução `sem HNR`, onde os APs foram configurados em sua capacidade máxima de transmissão.

6.3.2.1 Vazão

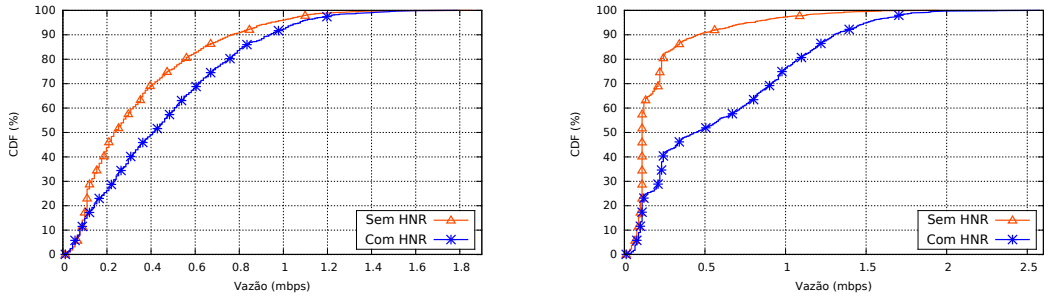
A primeira métrica analisada nesse cenário foi a vazão. Na Figura 6.7 exibimos tal métrica, em $Mbps$, medida nas estações do experimento. Para tal, comparamos a solução `sem HNR`, com potência estática, com a aplicação do HNR (`com HNR`) com potência dinâmica e coordenada. Plotamos um gráfico para cada estação utilizada. Neles, o eixo vertical representa a probabilidade de o valor medido corresponder à métrica demonstrada no eixo horizontal. As estações foram nomeadas como $STA\_HNR(1-4)$. No cenário do experimento, Figura 6.6, inserimos o tráfego com o auxílio da ferramenta *bwping* utilizando o meio de transmissão sem fio. Primeiramente, o serviço foi iniciado com a aplicação de controle coordenado de potência de transmissão dos APs_HNR. A partir da execução da aplicação, iniciamos também o tráfego nas estações e a ativação das $USRPs$. O mesmo procedimento foi efetuado para a aplicação `sem HNR` em momentos distintos.

Durante a execução do experimento a nossa aplicação permaneceu monitorando os quatro APs_HNR. De acordo com os níveis de interferências presenciados ao redor de cada AP_HNR a aplicação alterou dinamicamente e coordenadamente a sua potência de transmissão. A interferência presenciada em cada AP_HNR é a responsável por alterar o valor da métrica configurada na aplicação permitindo a atuação na rede. Na execução houveram APs_HNR que tiveram a sua potência aumentada, outros diminuída e outros a potência flutuou buscando um equilíbrio para atender ao limiar configurado. O limiar para a vazão foi definido empiricamente para os APs_HNR.

Nas Figuras 6.7(a), 6.7(b), 6.7(c) e 6.7(d) apresentamos as CDFs das vazões nas estações STA_HNR1, STA_HNR2, STA_HNR3 e STA_HNR4, respectivamente e estas representam a primeira avaliação realizada. Nota-se nos gráficos a superioridade e os benefícios de nossa solução quando a comparamos com a solução `sem HNR`. Atribuímos a discrepância entre a vazão aferida em cada estação à sua localidade no cenário configurado, à diferença entre o *hardware* das interfaces de rede sem fio dos APs_HNR e aos diferentes níveis de SNR de cada estação. Acreditamos que o comportamento da curva apresentada para a vazão, com o HNR, na Figura 6.7(b), ocorreu devido à limitações de parâmetros configurados no sistema operacional.



(a) STA_HNR1 (b) STA_HNR2



(c) STA_HNR3 (d) STA_HNR4

Figura 6.7: Vazão das estações

Apresentamos o resultado da segunda avaliação realizada na Figura 6.8. Para gerar a figura plotamos os dados de todas as estações em conjunto na CDF, ou seja, a Figura 6.8 corresponde ao agrupamento dos dados da Figura 6.7 – o mesmo foi realizado para as próximas duas métricas. A partir dessa apresentação percebe-se que no geral a aplicação com o *HomeNetRescue* também supera a solução `sem HNR`. Desta maneira, na Tabela 6.2 apresentamos que o ganho médio na vazão com a utilização do HNR foi de aproximadamente 66%.

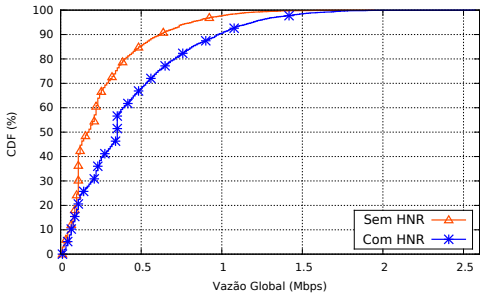


Figura 6.8: Vazão geral das estações em conjunto.

Tabela 6.2: Vazão média considerando os dados das estações no cenário 2.

Estações	Valor Médio	Intervalo de confiança 1 - α = 95%		Desvio Padrão	Ganho (%)
		Mínimo	Máximo		
Sem HNR	0.261	0.254	0.269	0.25	**65.77**
Com HNR	**0.433**	0.424	0.444	0.36	

6.3.2.2 Atraso

O atraso na entrega de pacotes utilizando transmissões sem fio foi outra métrica de rede analisada nesse cenário, pois ele pode interferir na QoE dos usuários ao assistir um vídeo, participar de uma teleconferência, entre outras aplicações. Na Figura 6.9 exibimos o atraso, em segundos, medido nas estações do experimento. Da mesma maneira que avaliamos a vazão, para o atraso também comparamos a solução `sem HNR` com a aplicação `com HNR`.

Nas Figuras 6.9(a), 6.9(b), 6.9(c) e 6.9(d) apresentamos as CDFs dos atrasos nas estações. A partir delas percebemos que assim como os ganhos de vazão proporcionados pela aplicação `com HNR`, nessa avaliação nota-se também que para todas as estações, quando comparamos a nossa solução com a solução *baseline* (`sem HNR`), que a nossa apresentou vantagens e benefícios para a rede, mostrando-se ser superior à solução comparada devido à redução no atraso proporcionada. Para esse critério o HNR também se mostrou ser superior à solução comparada, demonstrando a sua viabilidade de utilização e que o mesmo consegue proporcionar melhorias individualmente em cada ponto acesso de sua rede.

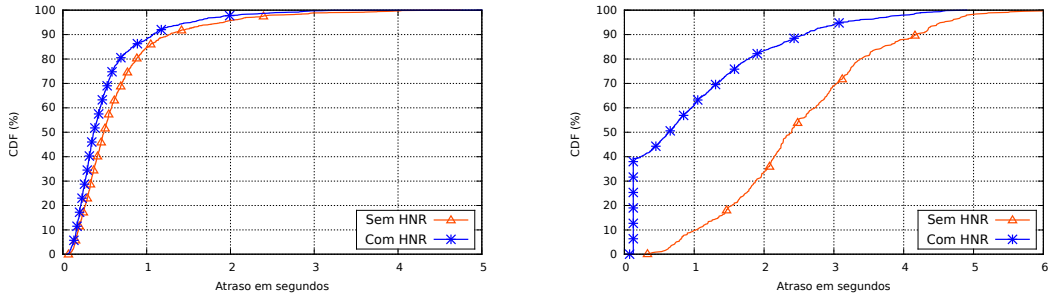
Os dados de atraso das estações em conjunto foram apresentados na Figura 6.10. A partir dessa apresentação percebe-se que no geral a aplicação com o *HomeNetRescue* também supera a solução sem o HNR. Na Tabela 6.3 apresentamos que a redução média do atraso com a utilização do HNR foi de aproximadamente 36%.

Tabela 6.3: Atraso médio considerando os dados das estações no cenário 2.

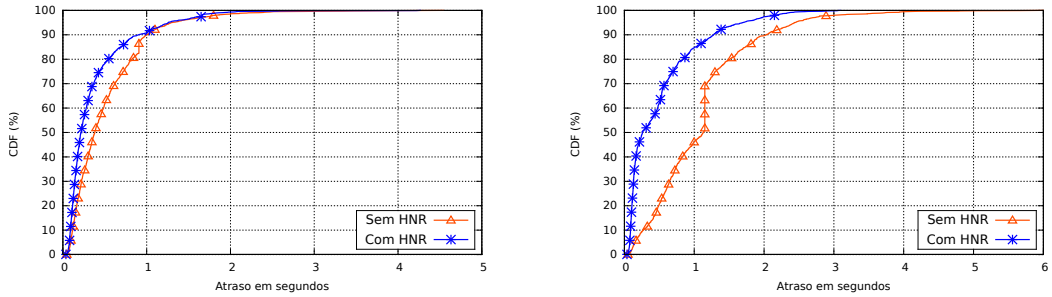
Estações	Valor Médio	Intervalo de confiança 1 - α = 95%		Desvio Padrão	Redução (%)
		Mínimo	Máximo		
Sem HNR	0.926	0.900	0.951	0.88	**36.17**
Com HNR	**0.591**	0.572	0.610	0.69	

6.3.2.3 *Jitter*

O último parâmetro de rede analisado nesse cenário foi a variação estatística do atraso na entrega de pacotes, *jitter*. Na Figura 6.11 exibimos o *jitter*, em segundos, medido nas estações. Nas Figuras 6.11(a), 6.11(b), 6.11(c) e 6.11(d) apresentamos as CDFs dos



(a) STA_HNR1 (b) STA_HNR2



(c) STA_HNR3 (d) STA_HNR4

Figura 6.9: Atraso das estações.

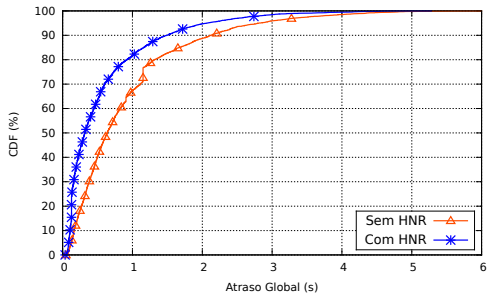
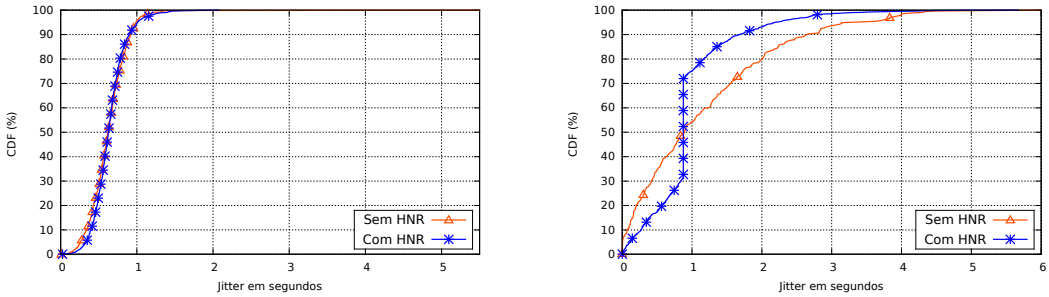


Figura 6.10: Atraso geral de todas as estações em conjunto.

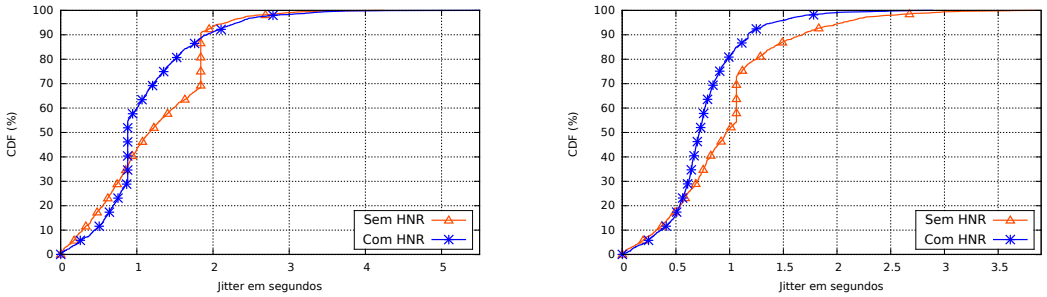
jitters em cada uma das estações. Nessa avaliação também houve uma redução média no *jitter* para todas as estações quando comparamos a nossa solução com a solução *baseline*.

Os dados de *jitter* das estações em conjunto foram apresentados na Figura 6.12. A partir dessa apresentação percebe-se que no geral a aplicação com o *HomeNetRescue* também supera a solução sem o HNR, mas para esse parâmetro, de maneira mais discreta. Desta maneira, na Tabela 6.4 apresentamos que a redução média do *jitter* nas estações da rede com a utilização do HNR foi de aproximadamente 15%.

Ao analisarmos os resultados referentes ao *jitter* concluímos que apesar da aplicação não necessitar de QoS, o controle realizado no HNR poderia ser benéfico para



(a) STA_HNR1 (b) STA_HNR2



(c) STA_HNR3 (d) STA_HNR4

Figura 6.11: *Jitter* geral de todas as estações em conjunto.

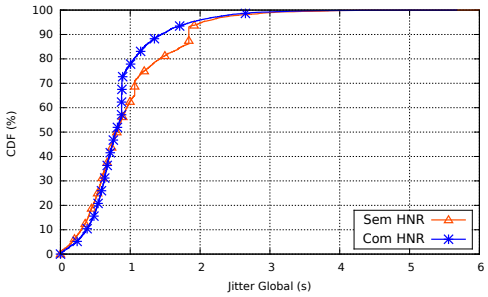


Figura 6.12: *Jitter* de todas as estações em conjunto

Tabela 6.4: *Jitter* médio considerando os dados das estações no cenário 2.

Estações	**Valor Médio**	**Intervalo de confiança 1 - α = 95%**		**Desvio Padrão**	**Redução (%)**
		Mínimo	**Máximo**		
Sem HNR	1.052	1.028	1.077	0.83	**15.46**
Com HNR	**0.890**	0.874	0.906	0.60	

aplicações que demandem baixos valores de *jitter*, como aplicações interativas (jogos, telecontrole) e transmissão de vídeos. O objetivo da aplicação não foi reduzir o *jitter*, mas isso foi obtido como consequência de uma melhor alocação da potência de transmissão. Em complemento, acreditamos que com a nossa abordagem houve uma redução na quantidade de colisões no meio.

Ressaltamos que os altos valores de atraso e *jitter* observados nas Tabelas 6.3 e

6.4 se justificam devido ao ambiente ao redor do laboratório conter várias redes sem fio concorrentes. Em análises realizadas no ambiente contabilizamos cerca de 80 pontos de acesso por 35 redes sem fio (SSIDs distintos). Em um dia calmo medimos valores em torno de 300 ms, com picos de 3-4 segundos, sem a execução do nosso experimento. Para atraso, as medianas são menores que as médias - 638 ms e 317 ms sem e com *HNR*, enquanto que para *jitter*, são 828 ms e 794 ms para sem e com *HNR*, respectivamente.

Por fim, após as avaliações realizadas para esse cenário, ao compararmos a nossa solução com uma solução sem a utilização do serviço, percebemos que SDN em nossa abordagem proporciona diversas vantagens a uma rede doméstica. Atualmente, o cliente ou o administrador dessas redes podem configurar a potência de transmissão dos pontos de acesso de sua rede em sua capacidade máxima caso queiram priorizar sua rede. Contudo, essa solução apresenta limitações por ser estática e por não considerar a vizinhança no processo de tomada de decisão para ajuste dos parâmetros. Realizar o controle coordenado e dinâmico da potência de transmissão mostra-se interessante, pois além de proporcionar benefícios locais, também proporciona benefícios globais, ou seja, a rede tem a capacidade de se auto ajustar em distintas maneiras para lidar com diferentes níveis de interferência e SNR dada a localização dos componentes.

6.4 Cenário 3: Seleção Dinâmica e Coordenada de Canais

Nesse cenário avaliamos a solução onde o *HomeNetRescue* busca otimizar a distribuição de canais entre os APs_HNR de forma dinâmica e coordenada. Os detalhes de implementação e o algoritmo foram apresentados na Seção 5.5.1.2. A seguir, descrevemos a configuração do ambiente e os componentes empregados para a sua avaliação. Na próxima seção apresentamos os gráficos e as discussões sobre seus resultados.

6.4.1 Componentes empregados

Para a execução desse experimento com o *HomeNetRescue* utilizamos outra combinação de componentes. Nesse cenário padronizamos parte do *hardware* utilizado. Utilizamos três computadores pessoais e quatro estações com as mesmas características de *hardware* e *drivers* de rede. Isso só foi possível pela liberação dos referidos componentes dada a demanda prévia dos mesmos no laboratório. Assim, a combinação utilizada para esse cenário foi composta de:

- **Pontos de Acesso** AP_HNR1**,** AP_HNR3 **e** AP_HNR4: com processador Intel Core i7-4500U CPU @ 1.80GHz; memória 8 GB; placa de rede *Qualcomm Atheros AR9485 Wireless Network Adapter 802.11bgn (PCI)*; *driver* de rede ath9k; e, sistema operacional: Linux Ubuntu versão 14.04 LTS em conjunto ao *daemom* do hostapd versão 2.6;

- **Ponto de Acesso 6** AP_HNR2: com processador Intel Core 2 Duo CPU E7500 @ 2.93GHz; memória 4GB; placa de rede *Qualcomm Atheros AR5416 Wireless Network Adapter 802.11bgn Atheros Communications 802.11n (PCI)*; *driver* de rede ath9k; e, sistema operacional Linux Ubuntu versão 14.04 LTS em conjunto ao *daemom* do hostapd versão 2.6;

- **Clientes**: quatro roteadores sem fio Linksys WRT54GL com o OpenWrt versão Backfire 10.03.x atuando como estações.

6.4.2 Resultados e Discussões

Configuramos esse cenário similar ao anterior (Figura 6.6), porém, sem a presença das $USRPs$ devido à densidade do ambiente e por termos apresentado, no cenário anterior, o impacto da contenção por elas geradas. Assim como no outro ambiente, nesse também continham variadas fontes interferentes e obstáculos. Dado o cenário, a solução buscou selecionar e alocar o canal menos ocupado ao AP_HNR em análise. As métricas avaliadas nesse cenário são as mesmas do anterior. Nesta, discutimos também sobre os dois tipos de avaliações realizadas, demonstrando os benefícios individuais e o benefício global da nossa solução. Em ambas avaliações comparamos nossos resultados com uma solução `sem HNR`.

Os pontos de acesso utilizados durante a execução das aplicações `com HNR` e `sem HNR` foram configurados para utilizarem o canal 1 (frequência 2.412 GHz) enquanto a potência de transmissão de cada um se manteve fixa, porém, conforme seu *hardware*, variou entre 14 e 24 dBm (capacidade máxima). Esse canal foi escolhido por, no momento do experimento, possuir a maior quantidade de roteadores vizinhos utilizando-o, consequentemente, por ser o que continha a maior carga em relação aos outros. Isto, devido aos SSIDs `DCC-usuarios` e `UFMG` representarem as principais redes utilizadas pelos alunos da universidade no prédio do Departamento de Ciência da Computação, local do experimento. Ainda, contabilizamos aproximadamente 35 redes (aproximadamente 80 roteadores) na vizinhança das salas utilizadas para o experimento conforme retratado na Figura 6.13, figura esta, obtida no dia 07/08/2017.

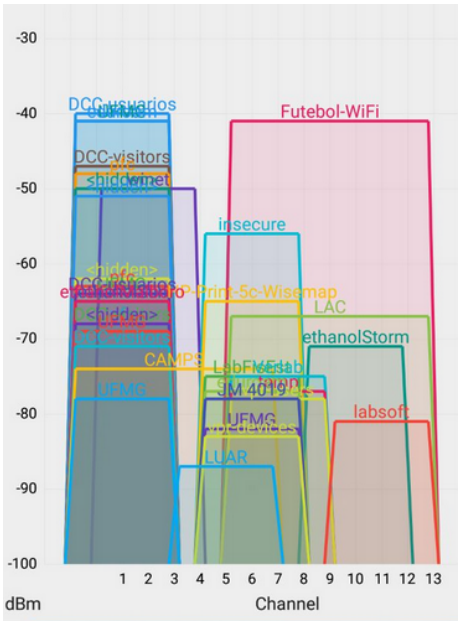


Figura 6.13: Redes vizinhas ao ambiente utilizado para a execução do experimento.

O cenário foi configurado para verificar a eficiência do *HomeNetRescue* em ambientes densos ou com diversas fontes interferentes. Ressaltamos que apesar de inicialmente a solução `com HNR` ter sido configurada com o mesmo canal da solução `sem HNR`, durante o experimento a nossa abordagem buscou atribuir dinamicamente e coordenadamente aos APs_HNR o canal com a menor carga e tempo de utilização.

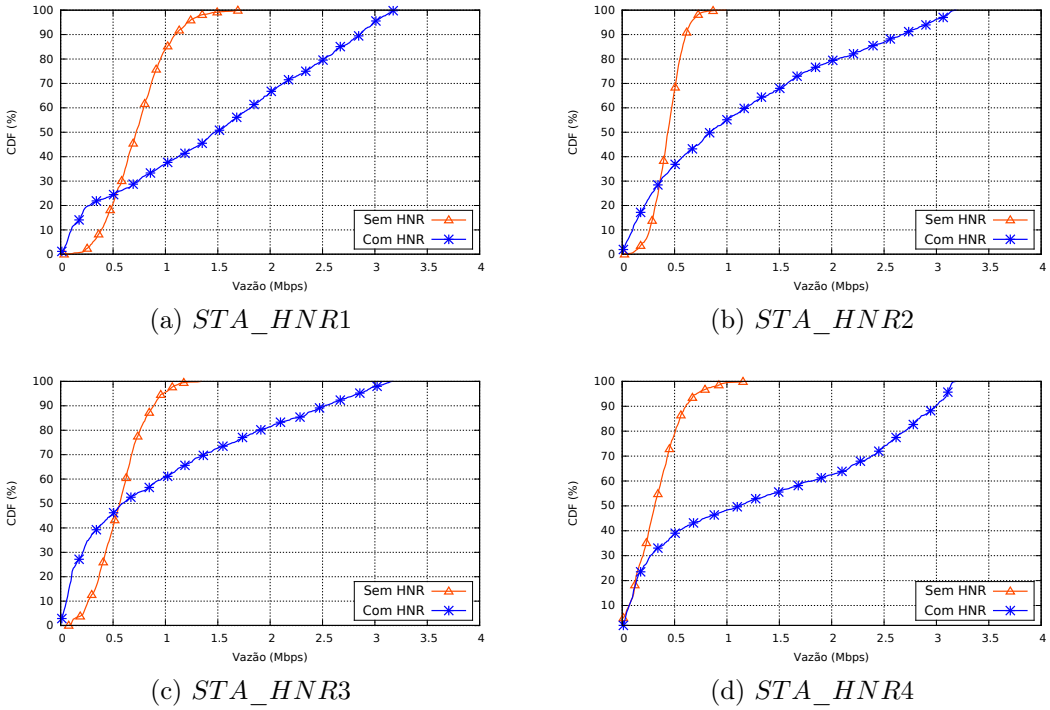
A seguir apresentamos os resultados obtidos para cada métrica avaliada.

6.4.2.1 Vazão

Na Figura 6.14 exibimos a vazão medida nas estações do experimento. Nota-se nos gráficos que a solução `com HNR` proporcionou ganhos na vazão da rede, mostrando-se ser superior à solução comparada. Para esse cenário, os resultados apresentaram um comportamento mais uniforme em cada estação, reflexo da padronização do *hardware* utilizado. Na Figura 6.15 plotamos os dados agregados de todas as estações. A partir dessa apresentação percebe-se que a aplicação com o *HomeNetRescue* também supera a solução sem o HNR. Desta maneira, na Tabela 6.5 apresentamos que o ganho médio na vazão com a utilização do HNR foi de aproximadamente 131%.

Tabela 6.5: Vazão média considerando os dados das estações no cenário 3.

Estações	**Valor Médio**	**Intervalo de confiança 1 - α = 95%**		**Desvio Padrão**	**Ganho (%)**
		Mínimo	**Máximo**		
Sem HNR	0.536	0.534	0.539	0.26	**131.39**
Com HNR	**1.242**	1.232	1.252	1.03	



(a) STA_HNR1 (b) STA_HNR2

(c) STA_HNR3 (d) STA_HNR4

Figura 6.14: Vazão das estações.

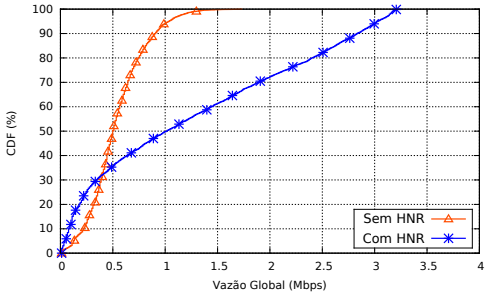
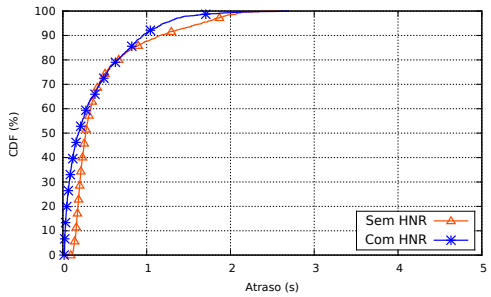


Figura 6.15: Vazão geral das estações em conjunto.

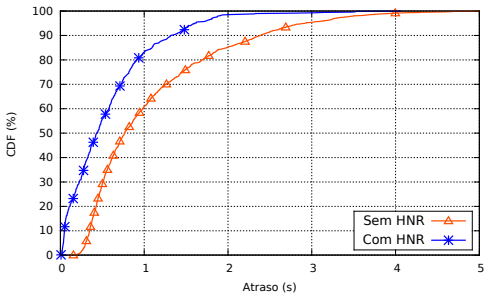
6.4.2.2 Atraso

Apresentamos as CDFs dos atrasos nas estações nas Figuras 6.16(a), 6.16(b), 6.16(c) e 6.16(d). Novamente notamos uma redução no atraso em todas as estações. Para esse critério o HNR também se mostrou ser superior à solução comparada. Isso demonstra a viabilidade de utilização do serviço e que o mesmo consegue proporcionar melhorias em cada ponto acesso de sua rede devido à sua capacidade de selecionar dinamicamente o canal com menor tráfego, permitindo então que os clientes se beneficiem com mais *"air time"*. Os dados de atraso global das estações foram apresentados na Figura 6.17. A referida figura também evidencia a redução proporcionada pelo serviço. Assim, na Tabela 6.6 indicamos que a redução média do atraso com a utilização do HNR foi de

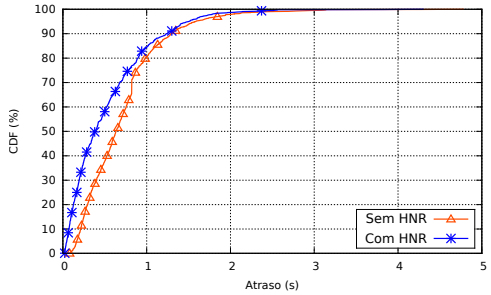
aproximadamente 47%.



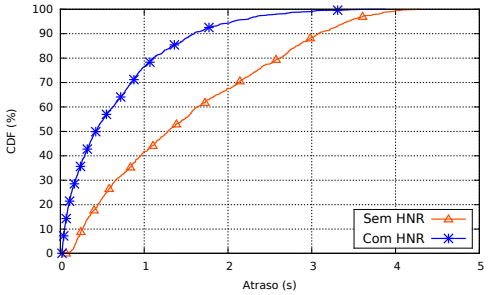
(a) STA_HNR1



(b) STA_HNR2



(c) STA_HNR3



(d) STA_HNR4

Figura 6.16: Atraso das estações.

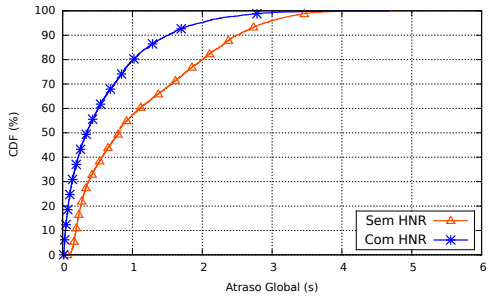


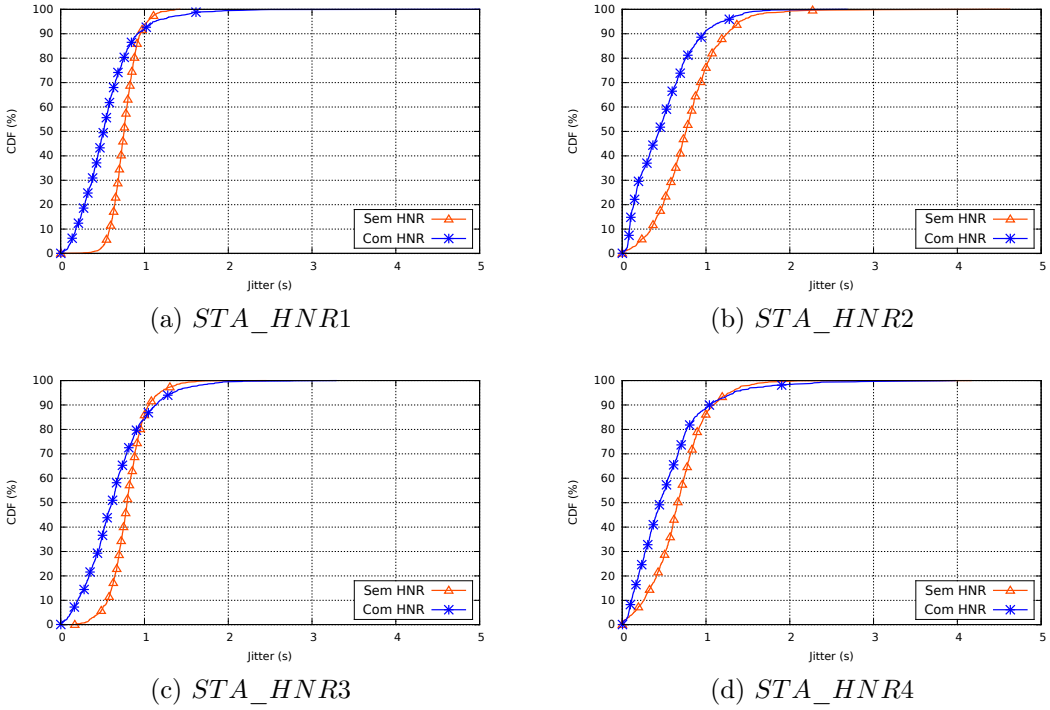
Figura 6.17: Atraso geral das estações em conjunto.

Tabela 6.6: Atraso médio considerando os dados das estações no cenário 3.

Estações	**Valor Médio**	**Intervalo de confiança 1 - α = 95%**		**Desvio Padrão**	**Redução (%)**
		Mínimo	**Máximo**		
Sem HNR	1.10	1.09	1.11	0.91	**46.77**
Com HNR	**0.58**	0.58	0.59	0.65	

6.4.2.3 *Jitter*

Do mesmo modo que nas avaliações anteriores, na Figura 6.18 apresentamos as CDFs dos *jitters* nas estações. Dados os gráficos, assim como o ganho de vazão e a diminuição do atraso, nessa avaliação nota-se a redução do *jitter* em cada estação, assim como globalmente. Na Figura 6.19 apresentamos o *jitter* global das estações. Na Tabela 6.7 apresentamos que a redução média do *jitter* nas estações da rede com a utilização do HNR foi de aproximadamente 24%.



(a) STA_HNR1 (b) STA_HNR2

(c) STA_HNR3 (d) STA_HNR4

Figura 6.18: *Jitter* das estações.

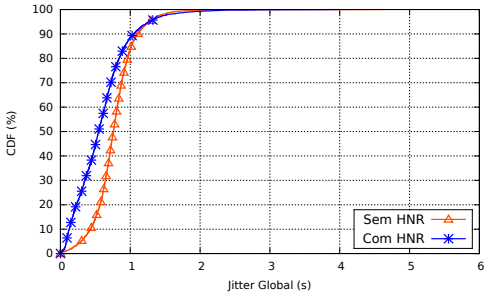


Figura 6.19: *Jitter* das estações em conjunto.

Por fim, embora o *hostapd* apresente a funcionalidade de realizar o ACS, isto é realizado somente uma vez quando o serviço é inicializado no ponto de acesso. Assim,

Tabela 6.7: *Jitter* médio considerando os dados das estações no cenário 3.

Estações	**Valor Médio**	**Intervalo de confiança 1 - α = 95%**		**Desvio Padrão**	**Redução (%)**
		Mínimo	**Máximo**		
Sem HNR	0.77	0.76	0.77	0.30	**24.37**
Com HNR	**0.58**	0.57	0.58	0.39	

uma vez configurado o canal, ele permanece em utilização (e fixo) enquanto o ponto de acesso estiver ativo. Não utilizamos outros pontos de acesso que realizam a seleção automática de canais por não os ter disponíveis para testes, assim como por apresentarem plataformas fechadas que inviabilizariam a comparação. Através desse cenário, concluímos mais uma vez que SDN proporciona vantagens primordiais para uma rede doméstica, pois possibilita que toda a rede seja analisada e controlada verificando o contexto de cada ponto de acesso, e dada a localidade, permite atribuir a ele o melhor canal.

6.5 Resumo

Nesse capítulo descrevemos a metodologia para os experimentos e as avaliações realizadas com o protótipo do *HomeNetRescue* em ambiente real. Comparamos nossa solução a uma solução *baseline* sem o HNR. Para cada cenário apresentado, avaliamos os parâmetros vazão, atraso e *jitter* e listamos a combinação de *hardware* utilizada. No cenário 1, o *HomeNetRescue* proporcionou um ganho de 7% na vazão. No cenário 2, os ganhos foram de 65.77% na vazão e redução de 36.16% no atraso e 15.46% no *jitter*. Por fim, no cenário 3, os ganhos foram de 131.39% na vazão e redução de 46.77% e 24.37% no atraso e *jitter*, respectivamente. No próximo capítulo apresentamos as considerações finais e os trabalhos futuros.

Capítulo 7

Conclusões e Trabalhos Futuros

Nesta dissertação propomos o *HomeNetRescue*, um serviço que emprega o conceito de redes definidas por *software* para realizar o gerenciamento autônomo de redes domésticas. O *HomeNetRescue* é voltado para a detecção, diagnóstico e solução automática de problemas em redes com e sem fio. Sua arquitetura é genérica e modular, o que permite a adição de novas aplicações e dispositivos a serem monitorados. O serviço foi modelado para poder ser empregado pelas provedoras de acesso à Internet de forma que estas gerenciem problemas à distância nas redes domésticas de seus clientes. Ainda, o serviço pode agregar a essas redes funcionalidades de detecção e solução automática de problemas de modo que estas se tornem mais estáveis e confiáveis. Isso pode proporcionar redução de custos para as provedoras, gerar menor demanda por serviços de suporte e menor tempo de recuperação em caso de falhas.

Avaliamos o protótipo do *HomeNetRescue* em um ambiente real utilizando três cenários. A partir das execuções dos experimentos com o protótipo nos cenários configurados, concluímos que o *HomeNetRescue* proporcionou benefícios individuais e globais na rede. Esses benefícios correspondem a ganhos na vazão e reduções no atraso e no *jitter* das transmissões sem fio das estações.

No cenário *Controle de Potência de Transmissão sem Coordenação* o *HomeNetRescue* proporcionou um ganho de 7% na taxa média de transferência (vazão). No cenário *Controle de Potência de Transmissão com Coordenação* o serviço melhorou a vazão em 66%, reduziu o atraso em 36% e o *jitter* em 15% quando comparado a uma rede sem o controle de potência. Por fim, no caso *Seleção Dinâmica e Coordenada de Canais* o serviço realizou a atribuição orquestrada de canais aos pontos de acesso da rede em um ambiente multi-AP. Nesse cenário o *HomeNetRescue* apresentou benefícios melhorando a vazão em 131% e reduzindo em 46% e 24% o atraso e o *jitter*, respectivamente, quando comparado a algoritmos de atribuição de canais de pontos de acesso

comerciais.

7.1 Trabalhos Futuros

O *HomeNetRescue* é uma arquitetura modular e permite incluir vários novos cenários. Sendo assim, recomendamos as seguintes atividades como trabalhos futuros:

- Implementar cenários complementares que demonstrem a capacidade de gerenciamento de problemas do *HomeNetRescue*. Dessa maneira, futuramente objetivamos adicionar alguns recursos ao serviço, entre eles destacamos:

    1. recursos que permitam a mobilidade (*handover*) de estações em caso de problemas. Isso poderia ser útil para gerenciar o ponto de conexão entre os pontos de acesso e as estações, requisitando que as estações reassociem a outros pontos de acesso da rede de forma a obter melhor qualidade de conexão ou uma situação de balanceamento no número de estações por ponto de acesso da rede;
    2. mecanismos para a detecção e identificação das causas de interferência;
    3. mecanismos para a detecção de áreas de sombra (desvanecimento) nas residências.
    4. mecanismos para a detecção de pontos de acesso defeituosos ou com $SSIDs$ inválidos através dos *beacons* obtidos da localidade da rede em análise. Em função disso a provedora poderia sugerir a substituição do aparelho;

- Finalizar a implementação do protótipo do *HomeNetRescue* para disponibilizá-lo em formato *open source* à comunidade;

- Melhorar os algoritmos de decisão adotados nos cenários, proporcionando a combinação de parâmetros distintos em conjunto (alteração de potência e alteração de canal) ou utilizando técnicas diferentes para ajustar os parâmetros controlados nos mesmos, por exemplo: *Mixed Integer Linear Program* (MILP), *machine learning* e teoria de jogos;

- Analisar e melhorar o mecanismo de detecção de conflitos de regras;

- Utilizar o serviço em controladores distribuídos; e,

- Adicionar e utilizar outros protocolos da camada de abstração de protocolos no *HomeNetRescue* para o processo de gerenciamento de problemas.

- Estudar a viabilidade de criação de uma *startup* para a comercialização do *homeNetRescue*.

Referências Bibliográficas

Adya, A.; Bahl, P.; Chandra, R. & Qiu, L. (2004). Architecture and techniques for diagnosing faults in ieee 802.11 infrastructure networks. Em *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking (MobiCom)*, pp. 30–44. ACM.

Alves, A. R.; Moura, H. D.; Borges, J. R. A.; Mota, V. F. S.; Cantelli, L.; Macedo, D. F. & Vieira, M. A. M. (2018). Homenetrescue: An sdn service for troubleshooting home networks. *Proceedings of the IEEE/IFIP 16th Network Operations and Management Symposium (NOMS 2018)*.

Alves, A. R.; Moura, H. D.; Cantelli, L.; Guimaraes, J. C. T.; Borges, J. R. A.; Silva, P. S.; Macedo, D. F. & Vieira, M. A. M. (2017). Um serviço sdn para detecção e solução de problemas em redes domésticas. *Proceedings of the XXII Workshop de Gerência e Operação de Redes e Serviços (WGRS) do SBRC*, pp. 96–109.

Avižienis, A.; Laprie, J.-C.; Randell, B. & Landwehr, C. (2004). Basic concepts and taxonomy of dependable and secure computing. *Journal of IEEE Transactions on Dependable and Secure Computing (TDSC)*, 1(1):11–33.

Berde, P.; Gerola, M.; Hart, J.; Higuchi, Y.; Kobayashi, M.; Koide, T.; Lantz, B.; O'Connor, B.; Radoslavov, P.; Snow, W. & Parulkar, G. (2014). Onos: Towards an open, distributed sdn os. Em *Proceedings of the 3rd Workshop on Hot Topics in Software Defined Networking (HotSDN)*, pp. 1–6. ACM.

Biswas, S.; Bicket, J.; Wong, E.; Musaloiu-E, R.; Bhartia, A. & Aguayo, D. (2015). Large-scale measurements of wireless network behavior. Em *Proceedings of the 15th ACM Conference on Special Interest Group on Data Communication (SIGCOMM)*, pp. 153–165. ACM.

Bouchet, O.; Javaudin, J.-P.; Kortebi, A.; El Abdellaouy, H.; Lebouc, M.; Fontaine, F.; Cochet, F.; Jaffre, P.; Brzozowski, M.; Mengi, A. et al. (2014). Acemind: The smart

integrated home network. Em *Proceedings of the 10th International Conference on Intelligent Environments (IE)*, pp. 1–8. IEEE.

Calvert, K. L.; Edwards, W. K.; Feamster, N.; Grinter, R. E.; Deng, Y. & Zhou, X. (2011). Instrumenting home networks. *Journal of Computer Communication Review (SIGCOMM)*, 41(1):84–89.

CISCO (2015a). Cisco IP Next-Generation Network (NGN). http://www.cisco.com/c/en/us/solutions/collateral/service-provider/ip-ngn-ip-next-generation-network/white_paper_c11-481360.html (acessado em 22 de junho de 2016).

CISCO (2015b). Cisco Visual Networking Index: Forecast and Methodology, 2016–2021. https://goo.gl/UjqVvW (acessado em 12 de outubro de 2017).

Correia, L. H.; Macedo, D. F.; dos Santos, A. L.; Loureiro, A. A. & Nogueira, J. M. S. (2007). Transmission power control techniques for wireless sensor networks. *Journal of Computer Networks*, 51(17):4765–4779.

DiCioccio, L.; Teixeira, R. & Rosenberg, C. (2012). Measuring and characterizing home networks. Em *Journal of ACM SIGMETRICS Performance Evaluation Review*, volume 40, pp. 383–384. ACM.

Dong, C. & Dulay, N. (2011). Argumentation-based fault diagnosis for home networks. Em *Proceedings of the 2nd ACM SIGCOMM 2011 Workshop (HomeNets)*, pp. 1–6.

Feamster, N.; Gao, L. & Rexford, J. (2007). How to lease the internet in your spare time. *Jounal of SIGCOMM Computer Communications Review*, 37(1):61–64.

Feamster, N.; Rexford, J. & Zegura, E. (2014). The road to sdn: an intellectual history of programmable networks. *Journal of ACM SIGCOMM Computer Communication Review*, 44(2):87–98.

Fonseca, P. & Mota, E. (2017). A survey on fault management in software-defined networks. *Journal of IEEE Communications Surveys and Tutorials*, pp. 1–30.

Fratczak, T.; Broadbent, M.; Georgopoulos, P. & Race, N. (2013). Homevisor: Adapting home network environments. Em *Proceedings of the 2nd European Workshop on Software Defined Networks (EWSDN)*, pp. 32–37.

Gheorghe, G.; Avanesov, T.; Palattella, M.-R.; Engel, T. & Popoviciu, C. (2015). Sdn-radar: Network troubleshooting combining user experience and sdn capabilities. Em *Proceedings of the 1st Network Softwarization Conference (NetSoft)*, pp. 1–5. IEEE.

Guedes, D.; Vieira, L. F. M.; Vieira, M. A. M.; Rodrigues, H. & Nunes, R. (2012a). Redes definidas por software: uma abordagem sistêmica para o desenvolvimento de pesquisas em redes de computadores. *Proceedings of the XXX Simpósio Brasileiro de Redes de Computadores (SBRC)*, pp. 160--210.

Guedes, D.; Vieira, L. F. M.; Vieira, M. M.; Rodrigues, H. & Nunes, R. V. (2012b). Redes definidas por software: uma abordagem sistêmica para o desenvolvimento de pesquisas em redes de computadores. *Proceedings of the XXX Simpósio Brasileiro de Redes de Computadores (SBRC)*, pp. 160–210.

IEEE-802.11 (2017). IEEE 802.11. http://www.ieee802.org/11/ (acessado em 27 de Setembro de 2017).

IEEE-802.3 (2017). IEEE 802.3. http://www.ieee802.org (acessado em 27 de Setembro de 2017).

Kanuparthy, P.; Dovrolis, C.; Papagiannaki, K.; Seshan, S. & Steenkiste, P. (2012). Can user-level probing detect and diagnose common home-wlan pathologies. *Journal of SIGCOMM Computer Communication Review*, 42(1):7–15.

Kim, H. & Feamster, N. (2013). Improving network management with software defined networking. *Journal of Communications Magazine*, 51(2):114–119.

Kim, H.; Sundaresan, S.; Chetty, M.; Feamster, N. & Edwards, W. K. (2011). Communicating with caps: managing usage caps in home networks. Em *Journal of SIGCOMM Computer Communications Review*, pp. 470–471. ACM.

Kim, K.-H.; Nam, H. & Schulzrinne, H. (2014a). WiSlow: A Wi-Fi network performance troubleshooting tool for end users. Em *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM)*, pp. 862–870.

Kim, K.-H.; Nam, H.; Singh, V.; Song, D. & Schulzrinne, H. (2014b). Dyswis: crowdsourcing a home network diagnosis. Em *Proceedings of the 23rd International Conference on Computer Communication and Networks (ICCCN)*, pp. 1--10. IEEE.

Koponen, T.; Casado, M.; Gude, N.; Stribling, J.; Poutievski, L.; Zhu, M.; Ramanathan, R.; Iwata, Y.; Inoue, H.; Hama, T. et al. (2010). Onix: A distributed control platform for large-scale production networks. *Proceedings of the 8th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 10:1–6.

Kreutz, D.; Ramos, F. M.; Esteves Verissimo, P.; Esteve Rothenberg, C.; Azodolmolky, S. & Uhlig, S. (2015). Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, 103(1):14--76.

Lee, M.; Kim, Y. & Lee, Y. (2015). A home cloud-based home network auto-configuration using sdn. Em *Proceedings of the 12nd IEEE International Conference on Networking, Sensing and Control (ICNSC)*, pp. 444–449.

Macedo, D. F.; Guedes, D.; Vieira, L. F. M.; Vieira, M. A. M. & Nogueira, M. (2015). Programmable networks: From software-defined radio to software-defined networking. *Journal of IEEE Communications Surveys Tutorials*.

McKeown, N.; Anderson, T.; Balakrishnan, H.; Parulkar, G.; Peterson, L.; Rexford, J.; Shenker, S. & Turner, J. (2008). Openflow: enabling innovation in campus networks. *Proceedings of the ACM SIGCOMM Computer Communication Review*, 38(2):69–74.

Mortier, R.; Rodden, T.; Lodge, T.; McAuley, D.; Rotsos, C.; Moore, A. W.; Koliousis, A. & Sventek, J. (2012). Control and understanding: Owning your home network. Em *Proceedings of the 4th International Conference on Communication Systems and Networks (COMSNETS)*, pp. 1–10. IEEE.

Mota, V. F.; Macedo, D. F.; Ghamri-Doudane, Y. & Nogueira, J. M. S. (2013). On the feasibility of wifi offloading in urban areas: The paris case study. Em *Proceedings of the 2013 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pp. 1–6. IEEE.

Moura, H.; Bessa, G. V.; Vieira, M. A. & Macedo, D. F. (2015a). Ethanol: Software Defined Networking for 802.11 Wireless Networks. *Proceedings of the 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pp. 388–396.

Moura, H.; Silva, E. & Macedo, D. F. (2017). Bwping-udp: Avaliando o desempenho de redes sem fio. *Proceedings of the XXXV Simpósio Brasileiro de Redes de Computadores (SBRC)*, pp. 1118–1125.

Moura, H.; Vieira, M. A. M. & Macedo, D. F. (2015b). *Ethanol: uma plataforma SDN para redes wi-fi*. dissertation, Universidade Federal de Minas Gerais.

Nadeau, T. D. & Gray, K. (2013). *SDN: Software Defined Networks*. O'Reilly, Sebastopol. ISBN 978-1-449-34230-2.

ONF (2017). Open Network Foundation. https://www.opennetworking.org/ (acessado em 27 de Setembro de 2017).

OpenFlow (2017). OpenFlow. https://www.opennetworking.org/sdn-resources/openflow (acessado em 16 de Novembro de 2015).

Pelle, I.; Lévai, T.; Németh, F. & Gulyás, A. (2015). One tool to rule them all: a modular troubleshooting framework for sdn (and other) networks. Em *Proceedings of the 1st ACM SIGCOMM Symposium on Software Defined Networking Research*, p. 24. ACM.

Perera, C.; Liu, C.; Jayawardena, S. & Chen, M. (2014). A survey on internet of things from industrial market perspective. *Journal of IEEE Access*, pp. 1660–1679.

Steinder, M. & Sethi, A. S. (2004). A survey of fault localization techniques in computer networks. *Journal of Science of Computer Programming*, 53(2):165 – 194.

Sundaresan, S.; Grunenberger, Y.; Feamster, N.; Papagiannaki, D.; Levin, D. & Teixeira, R. (2013). Wtf? locating performance problems in home networks.

Tanenbaum, A. S. & Wetherall, D. J. (2013). *Computer Networks: Pearson New International Edition: University of Hertfordshire*. Pearson Higher Ed.

Yiakoumis, Y.; Yap, K.-K.; Katti, S.; Parulkar, G. & McKeown, N. (2011). Slicing home networks. Em *Proceedings of the 2nd ACM SIGCOMM Workshop on Home Networks*, pp. 1–6. ACM.

Yishan, G.; Chen, D. & Jia, H. (2009). The bayesian network about computer network failure diagnosis under osi. Em *Proceedings of the International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, volume 1, pp. 44–46.