

Universidade Federal de Minas Gerais  
Programa de Pós-Graduação em Engenharia Elétrica

# Uma Arquitetura de Seleção de Redes Móveis Para Ambientes Heterogêneos

Douglas Chagas da Silva

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

*Orientador:*  
Lucas S. Batista

*Co-orientador:*  
Claudio C. Monteiro

Belo Horizonte, Fevereiro de 2015.

# Uma Arquitetura de Seleção de Redes Móveis Para Ambientes Heterogêneos

## Resumo

A tarefa de selecionar redes em um ambiente de tecnologias heterogêneas constitui um problema difícil, e não existe ainda uma solução ou técnica totalmente aceita para esse fim. Desta maneira, a implementação de novas técnicas de seleção se tornam necessárias, inclusive pela demanda na crescente utilização em redes veiculares, monitoramento de pacientes, cidades inteligentes, Internet das Coisas, dentre outras tecnologias e cenários envolvendo convergência de redes. Nesse contexto, este trabalho apresenta uma solução integrada para o problema de seleção de rede, que fornece um modelo de aplicabilidade real em ambientes envolvendo redes que já estejam em produção, realizando mínimas alterações em suas estruturas. Para isto, são usados protocolos e ferramentas computacionais para classificação de redes, através da utilização de métodos de auxílio à tomada de decisão em uma arquitetura tipo cliente/servidor. A solução proposta provê mecanismos que tornam a seleção de rede transparente para os usuários, exigindo baixo consumo de recursos computacionais dos dispositivos móveis e oferecendo total independência da operadora da rede. Testes de bancada foram realizados visando validar a arquitetura proposta. Os resultados apontam para a possibilidade real de oferecimento de uma solução para o problema de seleção de rede, sem custo computacional incremental para o dispositivo móvel e que permite ao usuário perceber uma qualidade de experiência adequada ao consumo de seus serviços de rede.

# A Mobile Network Selection Architecture for Heterogeneous Environments

## Abstract

The network selection task into heterogeneous technology environments is a difficult problem, and there is not a fully accepted solution or technique employed to reach this goal. In this way, the implementation of new network selection strategies represents a very important development topic, mainly due to its growing demand in vehicular networks, patient monitoring, smart cities, Internet of Things, among other technologies and scenarios involving network convergence. In this context, this work presents an integrated solution for the network selection problem, which provides a real applicable model for environments involving production networks, making minimal changes in their structures. To this end, protocols and software tools are used for network classification, employing decision-making aid methods in a client/server architecture. The proposed solution provides mechanisms that make the network selection transparent to users, requiring low computational resource consumption from mobile devices and offering a completely independence of the network operator. Testbeds have been conducted in order to validate the proposed architecture. The results indicate a real possibility of offering a solution to the network selection problem without incremental computational cost from the mobile device, allowing the user to observe an adequate experience quality concerning the consumption of its network services.

## Agradecimentos

Agradeço primeiramente a Deus, pelo dom da vida, saúde e oportunidades que tem me propiciado.

Aos meus pais, João e Josefa pelo exemplo de carácter e apoio incondicional em todos os projetos que tracei para minha vida.

A minha querida avó Dona Rosa, por tudo que representa na minha vida, pelo exemplo de discernimento e sabedoria.

A minha amada esposa e eterna namorada Danielle Lima. Seu amor e apoio foram fundamentais nesta jornada.

Ao meu orientador Prof. Lucas Batista, pela ajuda e orientação sempre enriquecedora em todas as fases do mestrado.

Ao meu co-orientador e amigo Prof. Claudio Monteiro, que tem me acompanhado e orientado durante toda a minha vida acadêmica. Do curso técnico no IFTO ao mestrado na UFMG.

Aos professores do PPGEE, em especial ao Prof. Rodney Saldanha, por ser um exemplo de profissional, e pela prontidão em ajudar a todos que o procuram.

Aos amigos que fiz na UFMG (em ordem alfabética), Alcy Monteiro, Carlos Santos, Cristiano Tavares, Diana Álbán, Diego Tami, Elder Fenili, Ernesto Lana, Estevão Modolo, Fredy Salazar, Heitor Salvino, Lianny Sanchez, Marcos Felipe, Maria Africano e Natália Alcântara.

A equipe do GREDES/IFTO, em especial aos amigos Fábio Lima, Manoel Campos, Pablo Moreira, Rodolfo Medeiros e Vinicius Rios. A troca de informações, discussões e contribuições técnicas foram sempre muito proveitosas.

Ao CNPq e a UFT pelo apoio financeiro, sem o qual esse trabalho não teria sido possível.

# Lista de Figuras

2.1. Fases do Handover . . . . .	19
2.2. Requisitos de algumas aplicações . . . . .	22
2.3. Métricas agrupadas de <i>handover</i> . . . . .	23
3.1. Visão geral da Arquitetura Proposta . . . . .	29
3.2. Esquema de seleção da arquitetura proposta . . . . .	31
3.3. Funcionalidades da aplicação Android . . . . .	39
3.4. Diagrama de Pacotes – UML . . . . .	41
3.5. Diagrama de Caso de Uso - UML . . . . .	42
3.6. Diagrama de estados - UML . . . . .	42
3.7. Diagrama de atividades - UML . . . . .	43
3.8. Diagrama de Classes - UML . . . . .	45
4.1. Cenário dos Experimentos . . . . .	50
4.2. Cenário dos Experimentos com Mobilidade . . . . .	57
5.1. Seleção realizada pelo SP e Móvel . . . . .	62
5.2. Variação Média Delay . . . . .	68
5.3. Variação Média Jitter . . . . .	68
5.4. Máxima perda de pacotes . . . . .	70
5.5. Variação média RSSI . . . . .	70

---

5.6. Resultado teste com 50 requisições . . . . .	74
5.7. Resultado teste com 100 requisições . . . . .	74
5.8. Resultado teste com 200 requisições . . . . .	75
A.1. Funções de Preferência . . . . .	90

# Lista de Tabelas

3.1. Percentual de vendas – Sistema Operacional Smartphone . . . . .	39
4.1. Configuração de <i>hardware</i> dos APs . . . . .	51
4.2. Parâmetros de configuração dos algoritmos . . . . .	53
4.3. Tráfego injetado nos conjuntos de testes . . . . .	55
5.1. Seleção de Rede – Site Parceiro . . . . .	60
5.2. Seleção de Rede – Móvel . . . . .	61
5.3. Valores do critério Delay: SP . . . . .	63
5.4. Valores do critério Delay: Móvel . . . . .	64
5.5. Valores do critério Jitter: SP . . . . .	65
5.6. Valores do critério Jitter: Móvel . . . . .	66
5.7. Valores do critério Perda de Pacotes: SP . . . . .	67
5.8. Valores do critério Perda de Pacotes: Móvel . . . . .	69
5.9. Valores do critério RSSI: SP . . . . .	71
5.10. Valores do critério RSSI: Móvel . . . . .	72
5.11. Resultados numéricos teste com mobilidade . . . . .	73
A.1. Escala Fundamental de Saaty . . . . .	87
A.2. Comparação par a par entre os critérios - AHP . . . . .	87
A.3. Somatório de cada elemento das colunas dos critérios - AHP . . . . .	87

---

A.4. Divisão de cada elemento pelo somatório da coluna - AHP . . . . .	88
A.5. Geração das PML - AHP . . . . .	88
A.6. Matriz de Critérios e Alternativas - Promethee II . . . . .	89
A.7. Características dos critérios para o decisor . . . . .	89
A.8. Matriz de comparação à luz do critério delay . . . . .	91
A.9. Matriz de comparação à luz do critério jitter . . . . .	92
A.10. Matriz de comparação à luz do critério perda de pacotes . . . . .	92
A.11. Matriz de comparação à luz do critério RSSI . . . . .	92
A.12. Matriz de comparação delay considerando o peso do critério . . . . .	92
A.13. Matriz de comparação jitter considerando o peso do critério . . . . .	93
A.14. Matriz de comparação perda de pacotes considerando o peso do critério . . . . .	93
A.15. Matriz de comparação RSSI considerando o peso do critério . . . . .	93
A.16. Matrizes de preferência positiva e negativa . . . . .	94
A.17. Matriz resultante Promethee II . . . . .	94



# Lista de Algoritmos

2.1. Pseudo-código processo de handover . . . . .	20
3.1. Obtenção dos parâmetros de QoS . . . . .	33
3.2. Medição do nível de sinal - RSSI . . . . .	34
3.3. Seletor de redes com o método Promethee II . . . . .	35

# Sumário

<b>Lista de Figuras</b>	<b>5</b>
<b>Lista de Tabelas</b>	<b>7</b>
<b>Lista de Algoritmos</b>	<b>9</b>
<b>1. Introdução</b>	<b>12</b>
1.1. Apresentação . . . . .	12
1.2. Motivação . . . . .	14
1.3. Objetivos do Trabalho . . . . .	15
1.3.1. Objetivo Geral . . . . .	15
1.3.2. Objetivos Específicos . . . . .	15
1.4. Contribuições . . . . .	15
1.5. Estrutura do Trabalho . . . . .	16
<b>2. Seleção de Redes</b>	<b>17</b>
2.1. Introdução . . . . .	17
2.2. Problema de Seleção de Redes . . . . .	18
2.3. Critérios de Seleção de Redes . . . . .	21
2.4. Técnicas de Seleção de Redes . . . . .	23
2.5. Decisão Multicritério . . . . .	24
2.6. Conclusão . . . . .	25
<b>3. Arquitetura Proposta</b>	<b>27</b>
3.1. Introdução . . . . .	27
3.2. Entidades da Arquitetura . . . . .	28
3.3. Esquema da Arquitetura Proposta . . . . .	30
3.3.1. Parte Servidora . . . . .	31
3.3.2. Parte Cliente . . . . .	38
3.4. Principais Vantagens da Arquitetura Proposta . . . . .	46

---

3.5. Escalabilidade da Arquitetura Proposta . . . . .	46
3.6. Restrições da Arquitetura Proposta . . . . .	47
3.7. Conclusão . . . . .	48
<b>4. Metodologia</b>	<b>49</b>
4.1. Introdução . . . . .	49
4.2. Cenário de testes sem mobilidade . . . . .	50
4.2.1. Definição dos parâmetros dos algoritmos . . . . .	52
4.2.2. Estratégias adotadas na validação . . . . .	53
4.2.3. Injeção de tráfego . . . . .	55
4.3. Cenário de testes com mobilidade . . . . .	57
4.4. Conclusão . . . . .	58
<b>5. Resultados</b>	<b>59</b>
5.1. Introdução . . . . .	59
5.2. Resultados dos testes sem mobilidade . . . . .	59
5.3. Análise para o cenário de testes com mobilidade . . . . .	73
5.4. Conclusão . . . . .	75
<b>6. Conclusões</b>	<b>76</b>
6.1. Trabalhos Futuros . . . . .	78
<b>Referências</b>	<b>80</b>
<b>A. Detalhamento dos Métodos MADM</b>	<b>86</b>
A.1. Aplicação do Método AHP . . . . .	86
A.2. Aplicação do Método Promethee II . . . . .	88

# Capítulo 1.

## Introdução

### 1.1. Apresentação

A convergência entre redes de tecnologias distintas tem se tornado cada vez mais uma realidade nos dias de hoje. A capacidade de processamento dos dispositivos móveis e a diversidade de serviços que podem ser utilizados com os mesmos têm impulsionado a maneira como as tecnologias de infraestrutura de acesso são modeladas. Neste contexto, o termo convergência refere-se a possibilidade de prover ao usuário a melhor qualidade de experiência possível no consumo de serviços que exijam sempre uma boa conexão. Ao consumir um serviço arbitrário, um usuário em seu dispositivo móvel usa protocolos de comunicação de dados bem conhecidos da pilha TCP/IP (Kurose & Ross 2013). Esta condição impõe ao dispositivo algumas restrições, sendo a mais grave retratada pelo fato de cada conexão ser tratada como uma conexão fim-a-fim e estar associada a um endereço de rede de origem específico. Desta forma, não é raro que um usuário, estando em uma conexão de voz (skype, por exemplo), se desloque com seu dispositivo e experimente uma alteração em sua conexão, gerada por este deslocamento. Cada alteração em sua conexão, que caracterize mudança de rede, gera alteração em seu endereço de rede de origem, fazendo com que sua conexão seja perdida (Monteiro 2012),(Ricardo 2009).

Trabalhos que propõem soluções para a questão relacionada a re-conexão de rede são encontrados na literatura desde a proposta do MIP (*Mobile IP*) (Perkins 1998), (Hsieh et al. 2003), (Xiang & Kong 2012). No entanto, estas soluções não funcionam sozinhas. Elas precisam de um “gatilho”. É fundamental saber o momento certo de trocar de rede, acionando assim os mecanismos MIP. Este momento está relacionado ao problema abordado nesta dissertação: a seleção de redes.

As principais questões que surgem neste processo são: *a)* Como escolher a melhor rede? *b)* A rede escolhida provê os requisitos necessários para o usuário? *c)* Os critérios de escolha são modelados de forma genérica e independente da tecnologia de acesso? Ainda sobre esta prerrogativa, *d)* a mudança do dispositivo móvel é assistida? *e)* Em algum momento nesta transição entre redes o serviço em uso é interrompido?

Estas questões têm sido alvo de diversas pesquisas, que se comprometem em resolver parte do problema, propondo soluções pontuais para determinados aspectos. Do ponto de vista do problema de seleção de rede, de maneira geral, a literatura propõe abordagens que contemplam o controle de admissão, a alocação de recursos e a seleção da melhor rede. As duas primeiras abordagens são propostas centradas exclusivamente na rede e a terceira apresenta propostas centradas na rede, no móvel, além de versões distribuídas (rede e móvel).

Desta forma, o problema de seleção de redes pode ser definido assim: Dado um conjunto de critérios ou parâmetros de rede, verificar em um dado momento e entre as redes disponíveis, qual rede melhor se adapta às necessidades do usuário e, por conseguinte, sugerir a mudança de rede (processo de *handover*) pelo dispositivo móvel (Bischoff 2008), (Alkhawlan & Ayesh 2008), (Bari & Leung 2007), (Ciccarese et al. 2009). Nesse caso, o processo de escolha da rede é condicionado a uma série de critérios.

É necessário destacar que o problema de seleção de redes ainda não possui uma solução ótima adotada pela indústria, visto que os dispositivos móveis atuais ainda usam o algoritmo *WLAN-First* (Ting et al. 2013), (Thiagarajah et al. 2013), (Choi et al. 2011). Como é sabido, dados referentes apenas ao nível de sinal das redes ou mesmo a priorização de uma tecnologia de rede em detrimento de outra, apenas baseado em seu custo monetário, não são suficientes para decidir se um móvel deve permanecer ou mudar de rede. Portanto, é necessário medir a QoS (*Quality of Service*) das redes envolvidas para que o dispositivo possa decidir com maior precisão, visando manter a QoE (*Quality of Experience*) do usuário. Esta medida de QoS pode ser feita pela rede ou pelo móvel e, se realizada pelo móvel, esbarra em problemas de consumo de energia.

Neste sentido, esta dissertação propõe uma arquitetura distribuída para seleção de redes, a qual faz uso de um servidor local que fornece informações sobre a qualidade das redes disponíveis, e do móvel que fará o consumo dessas informações. A arquitetura proposta não oferece consumo incremental de energia, é independente de operadora e envolve a integração de soluções de acesso a redes sem fio heterogêneas e estratégias de tomada de decisão multiatributo. Para tal, utiliza-se ferramentas computacionais

para seleção de redes, baseadas nos métodos AHP (*Analytic Hierarchy Process*) e PROMETHEE II (*Preference Ranking Organization Method for Enrichment Evaluations*) (Figueira et al. 2005). Para validar a arquitetura proposta, considerou-se um ambiente de testes (*testbed*) onde os dispositivos móveis estão conectados a redes com domínios administrativos diferentes, visando testar a solução em um ambiente independente de operadora.

## 1.2. Motivação

A convergência entre redes heterogêneas é um problema antigo, mas ainda sem solução definitiva. Pode-se fazer a convergência de redes heterogêneas que estejam acopladas (mesmo domínio administrativo) ou não acopladas (domínios administrativos diferentes). No primeiro caso, questões relacionadas a seleção de redes e gerência de mobilidade são tratadas internamente usando protocolos específicos de camada 2 e roteamento IP.

Por outro lado, redes não acopladas precisam de uma atenção maior. Considerando domínios administrativos diferentes e ausência de SLA (*Service Layer Agreement*), para saber qual a melhor rede, o móvel precisará medi-las e, portanto, consumir recursos fundamentais, como bateria (Trestian et al. 2012). Além disso, será necessário um bom protocolo de gerenciamento de mobilidade para prover a re-conexão IP, visto que, neste caso, roteamento não resolve (Monteiro 2012).

Desta forma, uma arquitetura de seleção independente da rede, que trabalhe em conjunto com o móvel e utilize técnicas de tomada de decisão, apresenta-se como uma possibilidade real para que o usuário possa ter uma boa QoE no consumo de serviços durante seus deslocamentos.

Um fator importante é que uma arquitetura de seleção de rede, independente da tecnologia de acesso, ofereça ao usuário a possibilidade de ter informações importantes sobre o QoS das redes envolvidas, sem a necessidade de contratar serviços com o operador da rede “A” ou da rede “B”. Outra questão que vale destacar é o fato de que uma arquitetura distribuída permite que as atividades contínuas de coleta de dados de QoS, e a computação desses dados pelo método de tomada de decisão, possam ser todos realizados em entidades da rede, retirando assim o consumo incremental de energia do móvel.

## 1.3. Objetivos do Trabalho

### 1.3.1. Objetivo Geral

Propor uma arquitetura distribuída para seleção de redes, sem consumo incremental de energia, independente de operadora, e que envolva estratégias de auxílio à tomada de decisão multiatributo.

### 1.3.2. Objetivos Específicos

Dentre os objetivos específicos são destacados:

- Estudo de modelos de seleção disponíveis na literatura;
- Proposição de uma arquitetura para classificação e seleção de redes, dentre as disponíveis para o usuário em um determinado ambiente;
- Aplicação de técnicas de apoio à tomada de decisão multicritério;
- Realização de testes de bancada e em cenários reais para avaliação e validação da arquitetura proposta.

## 1.4. Contribuições

A tarefa de selecionar redes em um ambiente de tecnologias heterogêneas constitui um problema difícil, visto que ainda não existe uma solução ou técnica totalmente aceita nesse campo, em razão das muitas variáveis e cenários existentes, como é o caso das soluções que consideram ou não o processo de mobilidade inter-redes.

Dessa maneira, a implementação de novas técnicas de seleção de redes se torna bastante viável, inclusive pela demanda na crescente utilização em redes veiculares, cidades inteligentes, monitoramento de pacientes, Internet das Coisas, dentre outras tecnologias e cenários envolvendo convergência de redes.

De maneira específica, acredita-se que os pontos apresentados a seguir são contribuições originais desta dissertação.

- Proposta de uma arquitetura que implementa um serviço de seleção de rede, onde a decisão de *handover* é compartilhada entre a rede e o dispositivo móvel;
- Desenvolvimento de um modelo de seleção de redes para dispositivos com sistema operacional Android;
- Aplicação do método Promethee II no contexto de uma arquitetura cliente/servidor.

## 1.5. Estrutura do Trabalho

Os assuntos discutidos neste trabalho estão organizados conforme mostrado a seguir.

**Capítulo 2 – Seleção de Redes:** Este capítulo discorre acerca dos principais trabalhos dedicados à seleção de redes, suas principais contribuições e as questões deixadas em aberto, bem como as tendências existentes. Visando tornar o texto autocontido, apresenta-se um estudo detalhado considerando-se as técnicas mais citadas na literatura.

**Capítulo 3 – Arquitetura Proposta:** Com o objetivo de propor uma solução eficiente para seleção de redes, apresenta-se uma proposta de arquitetura na qual a decisão de *handover* é compartilhada entre a rede e o dispositivo móvel. Além disso, são apresentadas as principais tecnologias envolvidas na proposta, bem como o funcionamento das principais aplicações envolvidas no *framework* de seleção e na implementação do modelo arquitetural.

**Capítulo 4 – Metodologia:** Neste capítulo são definidos alguns cenários de teste, de maneira a validar a arquitetura proposta. A principal questão envolvida consiste em verificar a similaridade entre o que é observado pela rede e pelo dispositivo móvel.

**Capítulo 5 – Resultados:** Inicialmente são descritos os experimentos utilizados para a validação da estratégia proposta. Feito isso, discorre-se sobre as métricas de avaliação de desempenho e o ajuste de parâmetros dos métodos. Por fim, apresenta-se o mecanismo usado para a realização da análise estatística dos resultados.

**Capítulo 6 – Conclusões:** Nesse capítulo são ressaltados os resultados obtidos para os cenários avaliados e a eficiência da arquitetura proposta, bem como as perspectivas de trabalhos futuros.



# Capítulo 2.

## Seleção de Redes

### 2.1. Introdução

Neste capítulo são apresentados os principais aspectos que envolvem o problema de seleção de redes e as principais abordagens apresentadas na literatura para resolvê-lo. No entanto, o objetivo principal aqui não é descrever as soluções apresentadas por relevantes trabalhos desde 2006 (Rodrigues Jr & Dias 2006), quando o problema começou a receber atenção de grupos de pesquisa e pesquisadores de todo o mundo. Estes trabalhos já foram bastante estudados e abordados em teses (Monteiro 2012) e dissertações (Rios 2012), (Bischoff 2008). Desde então seus resultados já foram usados e/ou comparados por outros trabalhos conduzidos por grandes centros de pesquisa.

O objetivo principal deste capítulo é contextualizar o problema de seleção de redes no cenário tecnológico atual, visando evidenciar a contribuição da abordagem utilizada para propor a solução apresentada nesta dissertação.

Nesta direção, diversas soluções têm sido propostas no sentido de avaliar e selecionar a melhor rede disponível, considerando um conjunto de critérios. Algumas destas soluções são computacionalmente viáveis, outras impõem modelos restritivos, que requerem um esforço computacional maior, inviabilizando a solução, visto que boa parte dos dispositivos que farão uso de tais aplicações estão limitados a poucos recursos de *hardware* (dispositivos móveis), quando comparados aos computadores tradicionais. A maioria das soluções comerciais são baseadas apenas no nível de sinal (*Received Signal Strength Indicator* - RSSI), ou no custo (preço/tarifação) no processo de escolha. As soluções embarcadas nos *smartphones* atuais, em sua maioria, baseiam-se no algoritmo *WLAN FIRST* (Ting et al. 2013) e (Thiagarajah et al. 2013). Neste algoritmo a opção do móvel é

sempre a rede *wireless* disponível em um determinado ambiente, mesmo que as condições de tráfego na rede de tecnologia WAN (*Wide Area Network*) estejam mais apropriadas, não avaliando, portanto, os requisitos necessários para a continuidade do serviço em uso por estes usuários (Mendes Valenzuela et al. 2011), (Valenzuela et al. 2012), (Quental & Gonçalves 2013).

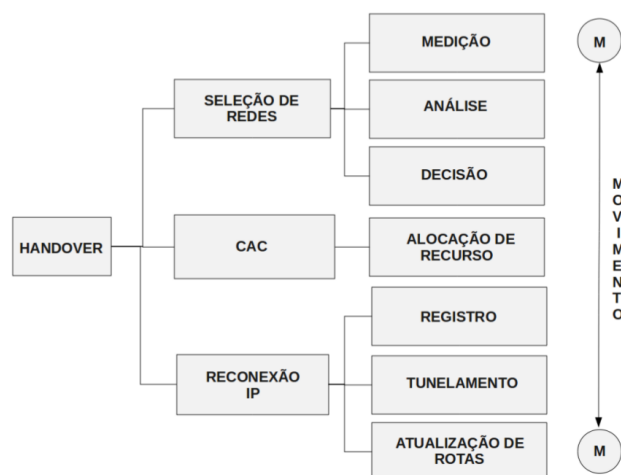
Alguns trabalhos têm proposto a utilização de algoritmos genéticos (Mohamed et al. 2012), (Alkhawlan & Ayesh 2008), (Bischoff 2008), como ferramenta para avaliar a capacidade das redes disponíveis e então sugerir a mudança. Outros trabalhos concentram-se em abordagens híbridas, os quais fazem uso de lógica *fuzzy* aliado com outras técnicas, como é o caso de (Chamodrakas & Martakos 2012), (Bari & Leung 2007), (Zhang 2004), (Radhika & Reddy 2011), (Lahby et al. 2013), (Alkhawlan & Ayesh 2008), onde foram propostas soluções utilizando a combinação de lógica nebulosa com algoritmos genéticos e métodos de tomada de decisão. Tem-se ainda trabalhos que utilizam redes neurais, teoria dos jogos, funções utilidade e processo de Markov (Trestian et al. 2012), (Ibrahim et al. 2009), (Chen 2010), (Khan et al. 2010), (Nguyen-Vuong 2008), (Shen & Zeng 2008).

De maneira geral, os trabalhos envolvendo seleção de redes variam de acordo com as inúmeras técnicas que são empregadas, considerando diferentes parâmetros de rede, mecanismos de *handover* distintos, tecnologias de acesso (e.g., WLAN e WMAN) heterogêneas e cenários de utilização diversos (Bari & Leung 2007), (Chamodrakas & Martakos 2012), (Kassar et al. 2008), (Niyato & Hossain 2009), (Lee et al. 2009), (Shen & Zeng 2008).

## 2.2. Problema de Seleção de Redes

De acordo com (Alkhawlan & Ayesh 2008), o processo de seleção de redes pode ser dividido em três fases ou blocos, que são: a coleta de dados, o processamento e a classificação das redes de acesso. Dessa forma, o problema de seleção de redes é parte integrante do conhecido mecanismo de *handover* que, de maneira geral, pode ser explicado como o processo que leva um dispositivo móvel a realizar a transição de uma rede *A* para uma rede *B* (Monteiro 2012). Convém salientar que esta transição poderá ocorrer entre tecnologias de acesso semelhantes (e.g., WLAN para WLAN) ou entre tecnologias de acesso distintas, como é o caso de uma transição de uma rede 3G (UMTS - *Universal Mobile Telecommunications System*) para uma rede WiMax (*Worldwide Interoperability for Microwave Access*) ou WLAN (*Wireless Local Area Network*) (Dreibholz

& Pulinthanath 2014), (Omheni et al. 2014), (Edwin Winston et al. 2014), (Kumaran & Shaji 2014), (Ramirez & Ramos 2009).



**Figura 2.1.:** Fases do Handover, obtido em (Monteiro 2012).

A Fig. 2.1 ilustra esse contexto, onde a seleção de redes se apresenta como parte do processo de mobilidade geral. Note que a seleção de redes assume três momentos. A medição compreende a obtenção dos parâmetros ou critérios pelos quais as redes disponíveis são comparadas. A análise refere-se a aplicação de procedimentos estatísticos ou funções de utilidade para avaliação do comportamento desses parâmetros. Por fim, a decisão refere-se a aplicação de alguma estratégia que mensure o quanto uma rede é melhor que a outra, baseado nas preferências do usuário e permitindo verificar se compensa executar o *handover*. A sequência de passos pode ser resumida pelo Algoritmo 2.1.

Um determinado dispositivo em movimento e fazendo uso de um protocolo de mobilidade, tal como o MIP (*Mobile IP*) (Perkins 1998), necessita fazer a troca para uma outra rede. Entretanto, o dispositivo precisa “perceber” o momento em que precisa efetuar a migração entre as redes. Isso só será possível após o dispositivo avaliar as redes disponíveis, verificar a qualidade ofertada pelas mesmas, para então decidir realizar a mudança. Assim, a seleção de redes deve funcionar como o gatilho para a mudança entre redes.

A seleção de redes depende, portanto, do processo de medição, isto é, necessita-se que o dispositivo móvel descubra as redes disponíveis no ambiente no qual ele está inserido, para que seja possível realizar alguma avaliação destas (a partir de um conjunto de parâmetros ou critérios), de maneira a inferir qual das redes possui os requisitos necessários

---

**Algoritmo 2.1:** Pseudo-código do processo de handover.

---

**Input:** nível de sinal, redes disponíveis, dados QoS das redes disponíveis

```
1 begin
2   Móvel na rede A;
3   Móvel percebe atenuação do sinal da rede A;
4   Coleta dados sobre as redes disponíveis (no caso, redes B, C e D);
5   Aplica-se a estratégia de seleção de redes (decide-se pela rede B);
6   if rede A usa mesma tecnologia da rede B then
7     Handover horizontal – desconecta da rede A e conecta na rede B;
8     Aciona protocolo de gerência de mobilidade na camada 3 (MIP);
9   else
10    Handover vertical – aciona interface e conecta na rede B sem desconectar
11    da rede A (soft handover);
12    Aciona protocolo de gerência de mobilidade na camada 3 (MIP);
13  end
14 end
```

**Output:** rede selecionada

---

para continuação ou melhoria do serviço em uso pelo dispositivo. Neste sentido, a avaliação das redes disponíveis é a principal atividade a ser tratada no processo de seleção. Nota-se, portanto, que este problema é multivariável e de características estocásticas, dada a aleatoriedade dos valores das variáveis que precisam ser consideradas.

Um exemplo disso pode ser encontrado no mecanismo de seleção de redes proposto por (Monteiro 2012), que precisou compor uma nova métrica de qualidade de vídeo (QoV), mapeando valores de qualidade de serviço (QoS) das redes. Nesse caso, o autor considerou apenas 03 (três) variáveis de QoS: atraso, *jitter* e perda de pacotes.

Além disso, não adianta nada propor soluções de seleção de rede, usando técnicas sofisticadas e boas métricas, se a infraestrutura de rede (camada 3) não for apta a suportar as consequências do processo de seleção. Isso significa que a última tarefa do processo de seleção de redes deve ser o acionamento dos protocolos de camada 3, destinados a recompor a conexão do móvel no menor tempo possível (Masri et al. 2014), (Hsiao & Lin 2014).

Assim, o processo de seleção de redes é fundamental na realização do procedimento de *handover*, cujo o papel é selecionar a melhor rede de acesso disponível ao usuário móvel, considerando critérios, parâmetros, regras e políticas bem definidos, de forma a

caracterizar e favorecer as vantagens de cada uma das tecnologias em um determinado momento, e onde haja heterogeneidade de tecnologias de acesso.

De maneira geral, a maioria dos trabalhos ou propostas de estratégias de seleção de rede são avaliadas via simulação, utilizando geralmente o *Network Simulator 2* (NS-2), *Network Simulator 3* (NS-3), OPNET, *Mobility Simulator* (MobiSim). A grande dificuldade de partir para estas abordagens é que apenas os métodos são validados, seja por comparação com outros métodos ou através da aplicação em cenários extremamente controlados, i.e., que na maioria das vezes não conseguem representar as particularidades do padrão da tecnologia de acesso, do meio de transmissão e do comportamento das variáveis consideradas.

Assim, a aplicabilidade prática destes modelos geralmente é comprometida, tendo em vista os inúmeros fatores que não são necessariamente controlados. Estes fatores apresentam comportamento dinâmico e variantes no tempo, principalmente quando há ocorrência de injeção de tráfego concorrente, tipo de tratamento de *buffer* implementado pelos dispositivos, além de considerações acerca de variação de sinal percebida pelas entidades envolvidas, geralmente *Access Point* (AP) e dispositivos móveis.

Desta forma, parte destes trabalhos estão apenas preocupados com a comparação do método pelo método, o que tem dificultado a adoção de modelos que realmente contemplem uma solução aplicável no que tange à seleção eficiente de redes. Assim, discute-se na seção seguinte algumas estratégias adotadas por grupos de pesquisa, os principais pontos considerados, e como foram implementadas as metodologias adotadas pelos mesmos.

### 2.3. Critérios de Seleção de Redes

Fundamentalmente, os critérios de seleção de redes estão intimamente relacionados às demandas ou aplicações em uso. Desta forma, parâmetros que mensuram a Qualidade de Serviço (QoS) das redes, bem como métricas objetivas de qualidade para aplicações específicas como QoVNS (*Quality of Video for Network Selection*) (Monteiro 2012), PSNR (*Peak signal-to-noise ratio*) (Silva, Almeida & Fonseca 2014), (Na & Kim 2014), além de métricas subjetivas, baseadas na experiência do usuário (QoE), como AVQ (*Automatic Video Quality*) (Suresh et al. 2007), (Suresh et al. 2009) e MOS (*Mean Opinion Score*),

podem ser consideradas no momento de realizar o processo de seleção de redes (Rios et al. 2012), (Monteiro 2012).

Apesar de existirem métricas de qualidade para tipos específicos de aplicações (áudio, vídeo e dados), este trabalho considerou o fato de que o processo de seleção de redes precisa ser ágil e consumir pouco recurso computacional do móvel, devido às restrições de energia. Desta forma, descartou-se o uso de processamento adicional no dispositivo, visando não gerar custo energético adicional. Portanto, considerou-se apenas variáveis de qualidade de serviço das redes como possíveis parâmetros de avaliação das mesmas.

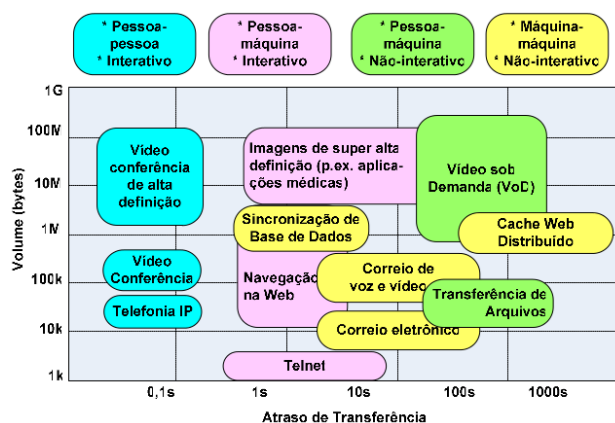


Figura 2.2.: Requisitos de algumas aplicações, obtido em (Bischoff 2008).

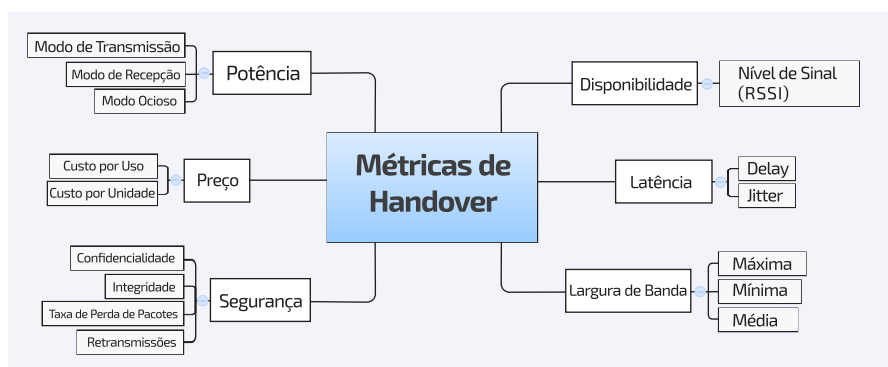
A Fig. 2.2 ilustra, de maneira genérica, estas demandas através da relação volume de dados por atraso de transferência. Nota-se que diferentes aplicações impõem restrições específicas de requisitos de rede. Considere, por exemplo, uma aplicação de videoconferência de alta definição. A mesma exigirá requisitos altos, do ponto de vista de desempenho da rede, para um bom funcionamento. Entretanto, se comparada aos requisitos de uma transferência de arquivos, ou navegação web, os requisitos são mais conservadores. Desta maneira, as redes disponíveis devem proporcionar a utilização dessas aplicações, proporcionando sempre que possível a melhor experiência para os seus usuários (conceito de redes ABC – *Always Best Connected*) (Bischoff 2008), (Bi et al. 2013).

Em suma, diversos são os critérios que podem ser utilizados para avaliação, classificação e seleção das redes disponíveis em um determinado ambiente. Entre eles podemos citar (Charilas & Panagopoulos 2010), (Braun & Abolhasan 2011):

- *Throughput* (Vazão);
- *Delay* (Atraso ou latência);

- *Jitter* (Variação do atraso);
- Taxa de perda de pacotes e taxas de erro;
- RSSI (Nível de sinal);
- Disponibilidade;
- Caracterização do tráfego;
- Utilização (quantidade de clientes);
- Custo monetário (preço/tarifa);
- Confiabilidade de serviço.

Esquemáticamente, estes critérios podem ser organizados em grupos, conforme Fig. 2.3, nos quais podem ser priorizados determinados subcritérios de acordo com o perfil de cada usuário, de maneira a subsidiar o procedimento de *handover* e propiciar sempre a melhor conexão. Essas prioridades são refletidas frequentemente na definição dos pesos de cada critério, no momento da avaliação e mudança de rede.



**Figura 2.3.:** Métricas agrupadas de *handover*, adaptado de (Bischoff 2008).

## 2.4. Técnicas de Seleção de Redes

Conforme comentado anteriormente, as propostas para solucionar o problema de seleção de redes envolvem vários métodos e técnicas. Entretanto, os mais comuns reportados na literatura compreendem o uso de: Lógica *Fuzzy*, Métodos MADM (*Multiple Attribute Decision Making*), Algoritmos Genéticos, Redes Neurais, Teoria dos Jogos, Funções Utilidade e Processo de Decisão de Markov. Entre os métodos MADM mais utilizados,

têm-se: AHP (*Analytic Hierarchy Process*), SAW (*Simple Additive Weighting*), TOPSIS (*Technique for Order Preference by Similarity to Ideal Solution*), MEW (*Multiplicative Exponential Weight*), SMART (*Simple Multiattribute Rating Technique*) e GRA (*Grey Relation Analysis*) (Rios 2012), (Wang & Binet 2009), (Figueira et al. 2005).

No que tange aos modelos que consideram soluções híbridas, uma vertente que tem conseguido resultados relevantes são as técnicas que compreendem o uso de Lógica *Fuzzy* e Métodos MADM (Junior 2013), (Rios 2012). De maneira geral, estes modelos funcionam de forma parecida, ou seja, após o processo de coleta dos dados, de acordo com os critérios delineados na seção anterior, ocorre o tratamento pela lógica nebulosa para em seguida ser realizada a classificação e seleção pelo método de decisão. Neste caso, para cada critério é dado um determinado peso, de forma a priorizar alguns serviços em detrimento dos outros, direcionando assim a escolha da nova rede de acordo com a aplicação em uso.

Em suma, os autores têm apresentado propostas que mostram o desenvolvimento e a aplicação de técnicas e tecnologias que consideram ambientes controlados, onde muitas das variáveis envolvidas na mobilidade de terminal são desconsideradas. Isto pode ser evidenciado com as soluções que consideram ou não o processo de mobilidade inter-redes.

## 2.5. Decisão Multicritério

Os métodos de tomada de decisão multicritério compreendem inúmeras técnicas que visam auxiliar na resolução de problemas, nas quais há ocorrências de incertezas, conflito de informações e disputa entre os critérios que serão considerados para avaliar as alternativas. Essas características são partes inerentes ao próprio problema que se deseja solucionar e devem portanto ser mapeadas pelo tomador de decisão (pessoa ou sistema) (Parreiras 2006). Objetiva-se desta forma avaliar um conjunto de alternativas viáveis, mediante diversos critérios, considerando para cada um deles os seus respectivos pesos, e.g., graus de importância ou preferências do decisor.

Neste trabalho considerou-se os métodos AHP e Promethee II. O primeiro utiliza níveis hierárquicos para a resolução do problema: em sua versão mais simples, no topo da hierarquia encontra-se o objetivo principal do problema; no nível imediatamente inferior são listados os critérios utilizados para se alcançar o objetivo principal; e no nível mais baixo as alternativas viáveis a serem avaliadas. Neste método faz-se uso de



uma escala fundamental para expressar a importância relativa (grau de preferência) entre as alternativas do projeto, considerando-se cada um dos critérios (Costa 2004). Tendo em vista os julgamentos fornecidos pela unidade de decisor e algumas operações básicas implementadas pelo método, a alternativa que apresentar o maior peso, após considerar a prioridade de cada critério, é escolhida como sendo a melhor. É possível verificar ainda se os julgamentos realizados apresentam consistências lógicas.

O método Promethee II utiliza o conceito de sobreclassificação, contribuição proposta pelos métodos de tomada de decisão multiatributo da Escola Européia (Figueira et al. 2005). A partir desse conceito, as alternativas são comparadas aos pares, de maneira a inferir se uma determinada alternativa é tão boa quanto outra. Caso uma alternativa  $a$  seja melhor que uma alternativa  $b$  considerando um dado critério, diz-se que  $a$  *sobreclassifica*  $b$ . Ao final de todas as comparações tem-se a classificação final (*ranking*) de todas as alternativas de um determinado conjunto, da melhor para a pior. Neste trabalho optou-se pela segunda versão deste método, pois o Promethee I admite situações onde ocorrem incomparabilidade entre alternativas. O Promethee II, por sua vez, inibe julgamentos nos quais as alternativas são incomparáveis, possibilitando sempre uma classificação completa das ações (Parreiras 2006).

A utilização do método AHP em conjunto com a aplicação do Promethee II, para o problema de seleção de redes, é apresentado no Apêndice A. Neste trabalho, utilizou-se a escala fundamental e o mecanismo de cálculo das prioridades fornecidos pelo AHP apenas para geração dos pesos dos critérios. O processo de tomada de decisão propriamente é feito pelo Promethee II. Por fim, uma comparação mais detalhada entre esses e outros métodos para o mesmo problema é apresentada em (Silva, Monteiro & Batista 2014), (Pradesh et al. 2014) e (Rios 2012).

## 2.6. Conclusão

Embora o problema de seleção de rede venha sendo estudado desde 2006, ainda existem questões não totalmente resolvidas. Estas questões podem ser divididas em três grupos:

- a. Modelagem matemática do problema: Como trata-se de um problema estocástico e multi-variável, ainda não existe um modelo ótimo que possa representar o comportamento real das variáveis de QoS de rede, dificultando assim o uso de métodos de otimização.

- b. Utilização de técnicas computacionais: Algumas abordagens já foram amplamente usadas, e.g., Algoritmo Genético, Rede Neural, Métodos MADM, Função de Custo, dentre outras. Ainda não foi comprovado qual das técnicas é mais eficiente para resolver o problema.
- c. Questões operacionais: Estas questões envolvem uma relação entre o que é ótimo e o que é possível fazer. O que está em jogo aqui é a possibilidade de que as soluções possam, de fato, virar protótipo e padrão para a indústria. Como mencionado neste capítulo, o assunto é estudado desde 2006 e até hoje, depois de muita evolução nas tecnologias de redes de acesso e de infra-estrutura de redes, é usado o algoritmo de seleção de redes *WLAN-First* nos dispositivos mais modernos. Até hoje, embora muitas soluções acadêmicas tenham apontado caminhos através de propostas validadas, não é possível a um usuário manter sua conexão skype ou youtube, por exemplo, durante sua movimentação entre redes.

A solução proposta neste trabalho foca nas questões (b) e (c). A arquitetura proposta foi concebida para que, de fato, possa ser utilizada, tendo em vista a mínima requisição de custo computacional para o móvel, além de poder ser escalável, distribuída e totalmente independente de operadora. Este último fato faz com que a solução possa ser oferecida como um serviço de rede, o qual não precisa ser implantado dentro do núcleo das redes das operadoras, o que contribui para a sua escalabilidade.

# Capítulo 3.

## Arquitetura Proposta

### 3.1. Introdução

A ideia de dividir o processamento no momento de seleção da rede vem sendo discutido há algum tempo. Entretanto, a maioria das propostas têm se concentrado em propor condições nas quais os dispositivos necessitam realizar medições localmente, bem como mecanismos de sinalização, para decidir em qual momento o *handover* deve ocorrer. Desta maneira, a necessidade de propor uma solução que utilize a menor quantidade de sinalização para auxiliar neste processo sempre se fez necessária. Parte destas propostas fazem uso do MIH (*Media Independent Handover*), que é um padrão (IEEE 802.21) que possibilita o *handover* entre tecnologias distintas, mantendo a continuidade do serviço em uso, ou ainda de outros protocolos de mobilidade (Trestian et al. 2014), (Bi et al. 2013), (Monteiro 2012).

Conforme comentado no capítulo 2, seleção de redes ainda é um problema em aberto, dada a ampla variedade de técnicas e cenários que podem ser explorados. Para um destes muitos cenários, em um ambiente de redes heterogêneas, cabe ao dispositivo móvel escolher a melhor rede, seja através de medições e processamento local, ou por meio de alguma premissa, tal como: realize a mudança para a rede de maior sinal, ou ainda, realize a mudança se a rede é gratuita. Esta última premissa basicamente compreende o funcionamento dos algoritmos que vem por padrão nos sistemas operacionais mais populares para *smarthphones*, como Android, iOS e Windows Phone.

Contudo, com a exigência cada vez maior das aplicações, demandando mais requisitos de rede, de forma a permitir a utilização de determinados serviços, a implementação de técnicas que possam fazer escolhas inteligentes passou a ser de fundamental importância.

Assim, inúmeras técnicas começaram a ser utilizadas para esta finalidade, conforme reportado no capítulo 2.

Neste capítulo, descreve-se a arquitetura proposta para o problema de seleção de rede, fazendo-se uso de técnicas de auxílio à tomada de decisão em um contexto de processamento distribuído, utilizando uma típica arquitetura cliente/servidor. Nesse contexto, conforme comentado inicialmente, é apresentada uma solução integrada para o problema de seleção de rede, na qual o processo de tomada de decisão é compartilhado entre o dispositivo móvel e a rede.

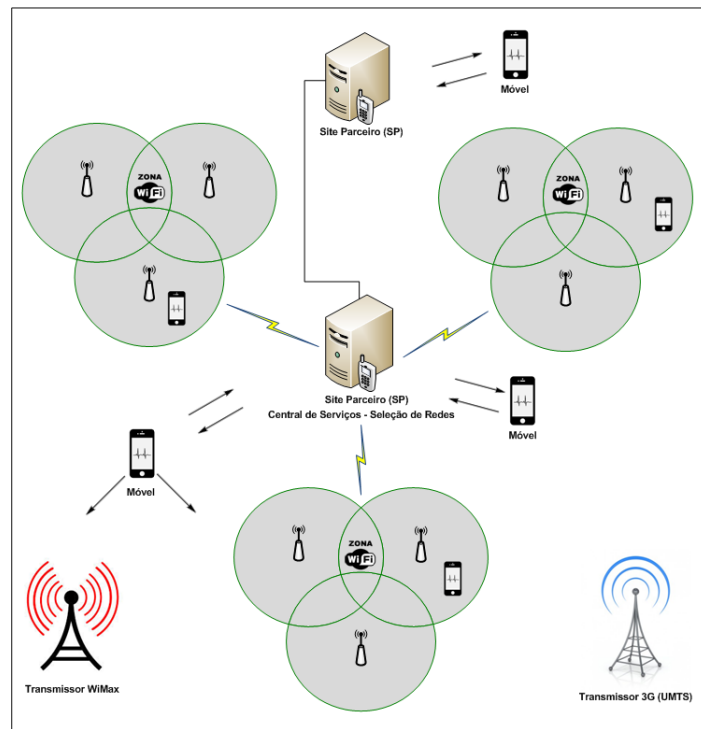
Desta forma, a arquitetura proposta deverá prover mecanismos que tornam a seleção de rede transparente para os usuários, oferecendo baixo consumo de recursos computacionais para os dispositivos móveis. A principal característica da arquitetura é então disponibilizar aos clientes/usuários sempre a referência da melhor conexão *wireless* disponível em um determinado ambiente. Assim, seja em locais públicos ou privados, fornecer informações sobre a qualidade das redes disponíveis aumenta a experiência dos usuários (QoE), influenciando diretamente na qualidade percebida e nos tipos de serviços (e.g., VoIP, VoD) que poderão ser utilizados.

## 3.2. Entidades da Arquitetura

A estrutura pela qual a arquitetura foi viabilizada necessita da existência de algumas entidades que estão envolvidas no processo de seleção de rede. Essas entidades compreendem neste estudo:

- Servidor: denominado Site Parceiro (SP). Essa entidade tem a função de disponibilizar as informações a respeito da qualidade das redes;
- Dispositivos móveis: irão consumir as informações do SP e tomar sua decisão baseada nas informações reportadas;
- Infraestrutura de acesso sem fio: composta pela presença de *Access Points* (APs) para o caso de enlaces 802.11b/g/n (WLAN), que compõem as chamadas Zona *Wifi*, ou enlace sem fio WAN (*Wide Area Network*) ou ainda enlaces do tipo WMAN (*Wireless Metropolitan Area Network*), tais como WiMAX (*Worldwide Interoperability for Microwave Access*), UMTS (*Universal Mobile Telecommunications System*) ou

a co-existência de todas essas tecnologias nas chamadas redes de quarta geração (4G) (Kejuruteraan et al. 2008).



**Figura 3.1.:** Visão geral da Arquitetura Proposta.

A arquitetura proposta é apresentada na Fig. 3.1. Conforme pode ser observado, a disposição do SP poderá ocorrer por domínio administrativo de rede ou conjunto de domínios distintos, funcionando como uma central de serviços. Isto significa que, em um determinado ambiente que se tem disponíveis diversas redes sem fio, seria necessário ao menos um servidor com a finalidade de fornecer informações sobre a qualidade dessas redes. Essa estrutura é similar ao que ocorre com outros serviços de rede, tais como um servidor DHCP (*Dynamic Host Configuration Protocol*) ou DNS (*Domain Name System*) por domínio administrativo.

A existência de vários sites parceiros em uma estrutura WAN permite a troca de informações sobre o histórico das redes que apresentaram melhores pontuações em um determinado período de tempo, de maneira a possibilitar a alimentação de uma rede de informações que descreva a qualidade das redes disponíveis em uma determinada região (bairro, cidade, etc). Nesse caso, por exemplo, para o domínio UFMG, poderia ser disponibilizado um mapa com as redes presentes e a qualidade ofertada pelas mesmas a partir das informações fornecidas pelo SP, em tempo real.

Observe ainda na Fig. 3.1, que os dispositivos móveis, compreendendo *smartphones*, notebooks, netbooks, relógios inteligentes, dentre outros, podem estar em uma determinada zona wifi, e requerer a melhor rede de acesso. É possível ainda que o móvel tenha a opção de escolher entre conectar-se em uma rede que compõe a zona *wireless* ou se manter na rede de tecnologia WAN, provida, por exemplo, por uma operadora de celular. Basta, para isso, que o SP “conheça” todas as redes que serão monitoradas, possuindo interfaces de conexão a estas redes.

### 3.3. Esquema da Arquitetura Proposta

A arquitetura proposta nesta dissertação adota uma metodologia similar à descrita em (Rios et al. 2012) para a implementação da parte servidora. O serviço de seleção de redes ofertado pelo SP utiliza os seguintes parâmetros de QoS: *delay*, *jitter*, perda de pacotes e RSSI (*Received Signal Strength Indication*). Além disso, foram adicionados os módulos para persistência dos dados de seleção e histórico das redes, utilizando um Sistema Gerenciador de Banco de Dados (SGBD) e a implementação de um serviço de disponibilização das informações, através do uso de *webservices*. Diferentemente da proposta de (Rios et al. 2012) e (Monteiro 2012), este trabalho concentra esforços em avaliar a percepção da qualidade de rede pelo dispositivo móvel e a verificada pela própria rede, a partir de um servidor local.

O *framework* de seleção de redes, parte integrante do SP, é constituído dos seguintes módulos: módulo de coleta de dados, módulo de processamento ou seleção, módulo de armazenamento e recuperação e, por fim, o módulo de disponibilização de informações e comunicação. No que tange a parte cliente da arquitetura, utilizou-se recursos dos dispositivos móveis, fazendo uso do sistema operacional Android<sup>1</sup>. Uma visão geral do esquema de seleção da arquitetura proposta, que compreende o *framework* desenvolvido para o SP e a aplicação móvel para os *smartphones*, é apresentado na Fig. 3.2.

Dessa forma, a arquitetura compreende uma série de requisitos (R) funcionais e não funcionais, os quais podem ser resumidos da seguinte forma:

- R1: Ser possível de implantação em servidores com diversas distribuições Linux;
- R2: Facilitar a implantação do site parceiro em redes em produção, de forma transparente;

---

<sup>1</sup><https://www.android.com>

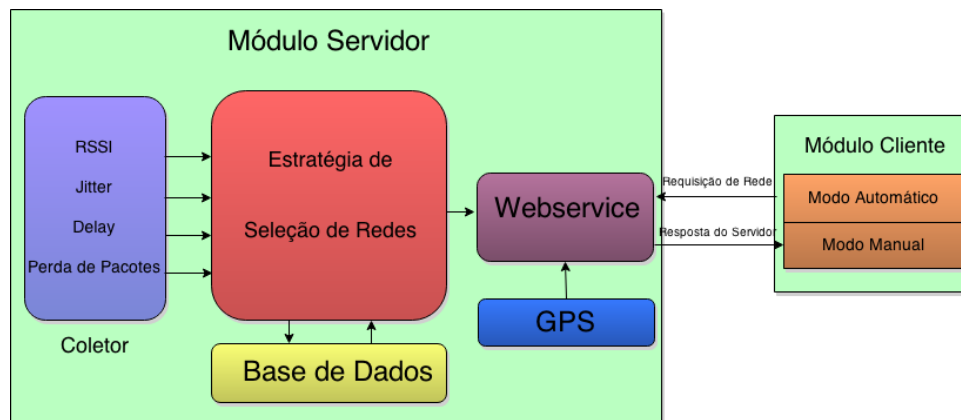


Figura 3.2.: Esquema de seleção da arquitetura proposta.

- R3: Facilitar a integração de módulos posteriores, adicionando novas funcionalidades, através da documentação disponível;
- R4: Estar descrita a partir do paradigma de orientação a objetos e padrões de projeto (aplicação móvel) (Sommerville 2011);
- R5: Ser independente de fabricantes e modelos específicos de hardware;
- R6: Utilizar preferencialmente softwares *open source*;
- R7: Tornar a aplicação móvel disponível em repositórios e lojas virtuais de forma gratuita.

Cada um dos módulos, tanto para a parte servidora quanto para a parte cliente, são detalhados a seguir.

### 3.3.1. Parte Servidora

A parte servidora da arquitetura (site parceiro) é composta por um conjunto de módulos responsáveis por funções específicas. Estas funcionalidades são comentadas de forma detalhada, entretanto uma visão geral é apresentada inicialmente. A partir do módulo de coleta obtém-se os valores referentes aos parâmetros de QoS utilizados no processo de seleção da melhor rede.

O módulo com a estratégia de seleção é responsável por fazer a escolha da melhor rede, a partir da relação entre os pesos dos critérios e os valores obtidos no processo

de coleta. Neste trabalho, o método utilizado para fazer a seleção é o Promethee II (Figueira et al. 2005).

A escolha do método Promethee II se deu principalmente pela sua eficiência e simplicidade quando comparado com outros métodos da literatura (Silva, Monteiro & Batista 2014), (Pradesh et al. 2014), (Karim et al. 2011). Entretanto a arquitetura pode utilizar qualquer outra estratégia, como redes neurais, algoritmos genéticos, dentre outros.

Após o processamento das informações coletadas e da escolha da melhor rede, ocorre a persistência dos dados no SGBD (Sistema Gerenciador de Banco de Dados) e, conseqüentemente, essas informações são disponibilizadas para a parte cliente, através do *webservice*. Além disso, informações sobre a localização dos SPs são providas pelo módulo GPS.

## Módulo de Coleta dos Dados

Para a realização das coletas desenvolveu-se um conjunto de *daemons* utilizando a linguagem de programação Python<sup>2</sup>, versão 2.7.5+, e Shell Script, além dos recursos do sistema operacional linux Ubuntu Server 12.04 LTS<sup>3</sup>. Utilizou-se ainda a biblioteca *libpcap*, através da ferramenta *tcpdump*<sup>4</sup>, para a obtenção dos dados de QoS da rede, neste caso, *delay*, *jitter* e perda de pacotes. Tais dados foram obtidos através da técnica de captura passiva de tráfego (Silva & Cancian 2008).

O módulo permite ainda a possibilidade de uso de captura a partir de medições ativas, e para tal fez-se uso de requisições ICMP (*Internet Control Message Protocol*). A utilização dos diferentes métodos de captura variam de acordo com o ambiente que se deseja monitorar; detalhes do processo são comentados no capítulo 4, que trata da validação da arquitetura.

O Algoritmo 3.1 apresenta a estrutura básica de funcionamento das rotinas responsáveis pela realização das coletas e medições dos parâmetros de rede. A função *checkInterfaces* verifica se as redes que serão avaliadas estão disponíveis e configuradas corretamente. Os arquivos de *logs* das medições são tratados pela função *writeTrace*. Esta função realiza a formatação dos dados, utilizando expressões regulares, permitindo a extração somente dos dados de interesse. A função *corData* correlaciona os *traces* gerados ao

---

<sup>2</sup><https://www.python.org>

<sup>3</sup><http://www.ubuntu.com/server>

<sup>4</sup><http://www.tcpdump.org>



vetor contendo as SSID das redes, enquanto a função *getQoS* calcula os valores dos parâmetros. Para cada um dos critérios obtidos nas medições, são calculados os valores de média, intervalos de confiança, desvio padrão e variância, através da função *getStatistics*. Por fim, os dados das coletas são persistidos no SGBD.

---

**Algoritmo 3.1:** Obtenção dos parâmetros de QoS.
 

---

**Input:** Quantidade de pacotes ICMP  $N$ , quantidade de redes monitoradas  $X$ , vetor SSID redes  $\mathbb{V}$ , vetor gateways redes  $\mathbb{Y}$ , ip SGBD  $P$ , passwd SGBD  $W$ , tipo de medição  $K$

**Output:** Trace delay  $\mathcal{A}$ , trace jitter  $\mathcal{B}$ , trace perda  $\mathcal{C}$

```

1 begin
2   checkInterfaces(X);           /* verifica as interfaces de rede */
3   sleep(2);
4   flag = K;
5   while true do
6     sleep(20);
7     if flag then
8       sniffer(X,Y);           /* faz medições passivas - tcpdump */
9       writeTrace(A,B,C)
10    else
11      icmp(X,Y)(N);           /* faz medições ativas - ICMP */
12      writeTrace(A,B,C)
13    end
14    data = corData((A,B,C)(V));
15    tab = getQoS(data);       /* calcula os parâmetros de QoS */
16    results = getStatistics(tab); /* calcula AVG, SD, VAR... */
17    connection(P,W);
18    persistence(results);     /* salva os resultados no SGBD */
19    closeConnection(P);
20  end
21 end

```

---

No que tange a obtenção dos dados referentes ao nível de sinal, implementou-se uma aplicação que utiliza o retorno do comando *iwlist*<sup>5</sup>, conforme ilustrado no Algoritmo 3.2 na função *execScan*. Convém salientar que nas interfaces de rede utilizadas na máquina com a função servidora, o valor do sinal medido encontra-se na escala entre 0 e 70, sendo 0 quando não existe enlace físico, e 70 o nível máximo de sinal percebido. Essas especificidades são detalhadas no capítulo 4, visto que variam de acordo com o *driver* de alguns fabricantes. A função *addSignal* realiza a formatação e adiciona os dados em uma estrutura do tipo lista para ser persistida no SGBD.

<sup>5</sup><http://wireless.kernel.org/en/users/Documentation/iw>

**Algoritmo 3.2:** Medição do nível de sinal - RSSI.

**Input:** Quantidade de redes monitoradas  $X$ , vetor SSID redes  $\mathbb{V}$ , result SCAN interfaces  $\mathbb{S}$ , ip SGBD  $P$ , passwd SGBD  $W$

**Output:** Lista resultante  $\mathcal{L}$

```

1 begin
2   checkInterfaces(X);           /* verifica as interfaces de rede */
3   sleep(2);
4   while true do
5     sleep(30);
6      $\mathcal{S} \leftarrow \text{execScan}(\mathbb{V});$            /* comando iwlist, awk, sed... */
7      $\mathcal{L} \leftarrow \text{addSignal}(\mathcal{S});$            /* adiciona em uma lista */
8     connection(P,W);
9     persistence( $\mathcal{L}$ );           /* salva os dados no SGBD */
10    closeConnection(P);
11  end
12 end

```

Ao final de cada *round* de coleta, os valores são persistidos em tabelas específicas do SGBD, através do módulo de armazenamento e recuperação de dados.

### Módulo de Processamento

Conforme comentado anteriormente, após a realização da coleta dos parâmetros de rede, executa-se a aplicação que faz a seleção da rede propriamente. Para tal, desenvolveu-se um módulo de processamento, que implementa um algoritmo em linguagem C (gcc versão 4.8.1), que utiliza os métodos AHP e Promethee II (Figueira et al. 2005). O primeiro é utilizado na definição do vetor de pesos para os critérios; e o segundo implementa o processo de tomada de decisão, avaliando as alternativas em função dos pesos de cada critério.

O Algoritmo 3.3 é o responsável pela seleção de rede, constituindo a parte mais importante ou núcleo da arquitetura proposta. Note que a estrutura básica do algoritmo contempla várias funções. Inicialmente utilizou-se a função *readMatrix* para ler as preferências do usuário (operador do sistema), em relação a importância de cada critério em comparação ao outro. Os critérios são fixos, conforme ilustrado na Fig. 3.2. Os julgamentos par-a-par entre os critérios são indicados segundo uma escala de 1 a 9, e refletem o grau da importância relativa entre os mesmos. Assim, calcula-se os valores dos pesos de cada critério baseado na matriz de preferência/julgamentos informada, utilizando a função *calculateAhpWeights*.

Nesta dissertação, utilizou-se apenas a escala fundamental e o mecanismo de cálculo das prioridades fornecidos pelo AHP. Alguns trabalhos dedicados à seleção de redes utilizam todas as funções do método (Lassoued et al. 2008), (Ramirez & Ramos 2009), (Song & Jamalipour 2005).

---

**Algoritmo 3.3:** Seletor de redes com o método Promethee II.
 

---

**Input:** Quantidade de redes monitoradas  $X$ , quantidade de critérios  $Q$ , estrutura dinâmica Lista com os dados das redes  $\mathcal{L}(t)$ , trace consulta  $C$ , nome base de dados  $D$ , nome da View relacionando QoS e RSSI  $V$ , nome tabela rede selecionada  $F$ , nome tabela ranking  $K$ , matriz de importância dos critérios  $M$ , vetor de pesos AHP  $H$ , ip SGBD  $P$ , passwd SGBD  $W$

**Output:** Rede selecionada  $\mathcal{M}$ , ranking redes  $\mathcal{R}(t)$

```

1 begin
2   connection(P,W);           /* conexão persistente com o SGBD */
3   sleep(2);
4   t = X;
5   i = Q;
6   MI(i) ← readMatrix (M(i));      /* importância dos criterios */
7   H(i) ← calculateAhpWeights (MI(i)); /* pesos dos criterios */
8   while true do
9     sleep(25);
10    C ← consultData (D, V);        /* recupera dados das medições */
11    L(t) ← readTrace (C);
12    AX(t) ← (L(t));
13    (M, R(t)) ← applyPromethee (AX(t), H(i)); /* tomada de decisão
    */
14    (F, K(t)) ← saveData (M, R(t)); /* dados para o webservice */
15  end
16  closeConnection(P);
17 end
  
```

---

As demais funções apresentadas no Algoritmo 3.3, como *consultData* e *readTrace* são responsáveis pela manipulação dos dados obtidos pelo módulo de coleta. Para a realização da persistência dos dados, após o tratamento realizado pelo método de tomada decisão, fez-se uso da biblioteca *libpq-fe*.<sup>6</sup> na função *saveData*. A biblioteca supracitada é formada por um conjunto de métodos, que permitem ao desenvolvedor (utilizando a linguagem de programação C) fazer uso dos recursos do SGBD PostgreSQL. Essa biblioteca encontra-se embutida em outros pacotes que fornecem as mesmas funcionalidades em outras linguagens como Python, C++, Perl, dentre outros.

---

<sup>6</sup><http://pgdocptbr.sourceforge.net/pg80/libpq.html>

Por fim, o Algoritmo 3.3 utiliza a função *applyPromethee*, que aplica o método Promethee II, responsável pela tomada de decisão, que corresponde a estratégia utilizada no núcleo da arquitetura, conforme Fig. 3.2. O método utiliza-se da curva de critério usual para os critérios *delay*, *jitter* e perda de pacotes. Para o critério RSSI, por outro lado, podem ser utilizadas no algoritmo tanto a curva com critério de indiferença quanto a curva de critério usual. Detalhes inerentes às curvas que podem ser utilizadas pelo método para diferenciação entre as alternativas quando da decisão, podem ser obtidas em (Figueira et al. 2005), (Neves et al. 2002).

Convém salientar que foram realizados testes utilizando outras curvas no método Promethee II, tais como linear, Gaussiana e trapezoidal. Contudo, em virtude da dinamicidade dos parâmetros, em razão das condições de tráfego das redes mudarem brusca-mente em períodos de tempo curtos, era necessário realizar constantemente alterações nos parâmetros adicionais, utilizados por essas curvas. Essa característica do sistema influenciava na qualidade de decisão do método, uma vez que a capacidade de classificação era comprometida, e.g., os valores coletados ficavam fora dos intervalos outrora definidos, mesmo usando intervalos relativamente grandes para as variáveis.

Um exemplo didático da execução do Algoritmo 3.3, no que tange a utilização do método AHP, de forma conjunta com a aplicação do Promethee II para o problema de seleção de redes, é apresentado no Apêndice A. O enfoque é dado na organização dos dados e nas operações de classificação e ranqueamento das alternativas em função dos critérios e pesos.

## Módulo de Armazenamento e Recuperação

Para o armazenamento dos dados coletados nos processos de medições, bem como a persistência de resultados referentes ao módulo de processamento, utilizou-se o SGBD PostgreSQL<sup>7</sup>, na versão 9.3. A organização dos dados utilizou a estrutura de tabelas necessárias ao método Promethee, assim como tabelas para armazenamento do histórico das redes ou para tabulação dos resultados dos testes detalhados no capítulo 4.

De modo a facilitar a manipulação dos dados persistidos, utilizou-se ainda recursos do tipo *view* e *functions* para tabulação dos testes e agregação dos valores obtidos dentro do próprio SGBD (Elmasri & Navathe 2005). É importante salientar que a arquitetura suporta acesso a qualquer outro banco de dados.

---

<sup>7</sup><http://www.postgresql.org>

## Módulo de Informações e Comunicação

O módulo de comunicação compreende um conjunto de métodos remotos, para informar a parte cliente sobre as melhores redes disponíveis em um determinado ambiente. O módulo também é responsável por fornecer informações de localização a respeito de outros SPs, bem como a posição do cliente em relação ao site parceiro mais próximo.

Para este trabalho fez-se uso de *Webservices* do tipo REST (*Representational State Transfer*), que consiste em um padrão utilizado na integração de aplicações distribuídas implementadas através de tecnologias distintas. Embora não seja um padrão W3C (*World Wide Web Consortium*), seu modelo arquitetural tem sido amplamente utilizado, em virtude principalmente da sua simplicidade, não exigindo formatação específica para respostas as requisições (Filho 2011).

O retorno das consultas são disponibilizadas em formato JSON (*JavaScript Object Notation*)<sup>8</sup>, que é uma notação de texto utilizada para troca de dados. Dada a notação extremamente simples, a representação por ele propiciada possibilita às linguagens de programação tratá-los como estrutura de dados já conhecidas (*arrays*, *structs*, *hash tables*, dentre outros). A adoção do formato JSON tem ganhado grande ênfase em virtude da facilidade de manipulação, em detrimento do uso de arquivos XML (*eXtensible Markup Language*) barganhados pelo consórcio W3C, que nem sempre são facilmente manipuláveis. Outra vantagem refere-se a diminuição considerável da quantidade de pacotes necessários para trafegar os dados, o que diminui o “*overhead*” na rede.

Para este trabalho, o uso do formato JSON na arquitetura proposta se deu como saída do *framework* de seleção, que apresenta para a aplicação cliente a informação com a melhor rede disponível para o usuário em um determinado ambiente ou, para alguns cenários, o *ranking* ordenado das melhores redes disponíveis.

Um exemplo da representação fornecida pelo JSON neste trabalho pode ser visualizado a seguir:

```
{“cod”：“10”,“idrede”：“5”,“nome”：“PPGEE”}
```

Nesta representação, os dados retornados da consulta ao SGBD exibem uma notação bastante simples (“campo:valor”). Esses campos (cód; idrede e nome) fornecem informações de uma determinada rede e podem ser recuperados por tipos nativos das linguagens de programação.

---

<sup>8</sup><http://json.org>

Para a implementação do *webservice* utilizou-se a linguagem de programação PHP<sup>9</sup>, versão 5.3.10 com suporte a acesso ao PostgreSQL (biblioteca pgsq) e o servidor Web Apache<sup>10</sup>.

### Módulo de Informações Geográficas

O módulo de informações geográficas compreende o uso de recursos do tipo GPS (*Global Positioning System*). Para tal, é necessário apenas que o dispositivo móvel consulte, no SP pré-cadastrado na aplicação android, a localização (latitude/longitude) do SP mais próximo. Neste cenário, é necessário que haja cooperação entre os SPs, conforme ilustrado na Fig. 3.1, ou que o usuário já tenha cadastrado os SPs no seu dispositivo.

O módulo permite ainda a apresentação de um mapa com a localização (distribuição) de todos os SPs que compõem a rede WAN, “coberta” pelo serviço de seleção de redes. Para isto, utiliza-se a API (*Application Programming Interface*) JavaScript do Google MAPS<sup>11</sup>, versão 3.

Tal módulo possibilita ainda a identificação das zonas *wireless* e os seus limites geográficos.

#### 3.3.2. Parte Cliente

No que se refere a parte cliente da arquitetura, implementou-se uma aplicação para o sistema operacional Android. De maneira resumida, pode-se dizer que o android é o sistema operacional para dispositivos móveis desenvolvido pela empresa Google, que vem embarcado em grande parte dos celulares e tablets comerciais. Baseado no núcleo e na ideologia do sistema operacional Linux, a plataforma possui código aberto, o que facilita a aceitação do mesmo por parte de desenvolvedores de software e fabricantes, que tenham interesse em utilizá-lo em seus aparelhos de forma customizada. Segundo a pesquisa realizada pela empresa de consultoria IDC (*International Data Corporation*)<sup>12</sup>, no segundo trimestre de 2014, 84.4% dos *smartphones* vendidos possuíam o sistema operacional do Google. Segundo a mesma empresa, esse número vem crescendo desde o ano de 2011, conforme ilustrado na Tabela 3.1.

---

<sup>9</sup><http://php.net>

<sup>10</sup><http://www.apache.org>

<sup>11</sup><https://developers.google.com/maps/?hl=pt-br>

<sup>12</sup>Acessado em dezembro de 2014, <http://www.idc.com/prodserv/smartphone-os-market-share.jsp>

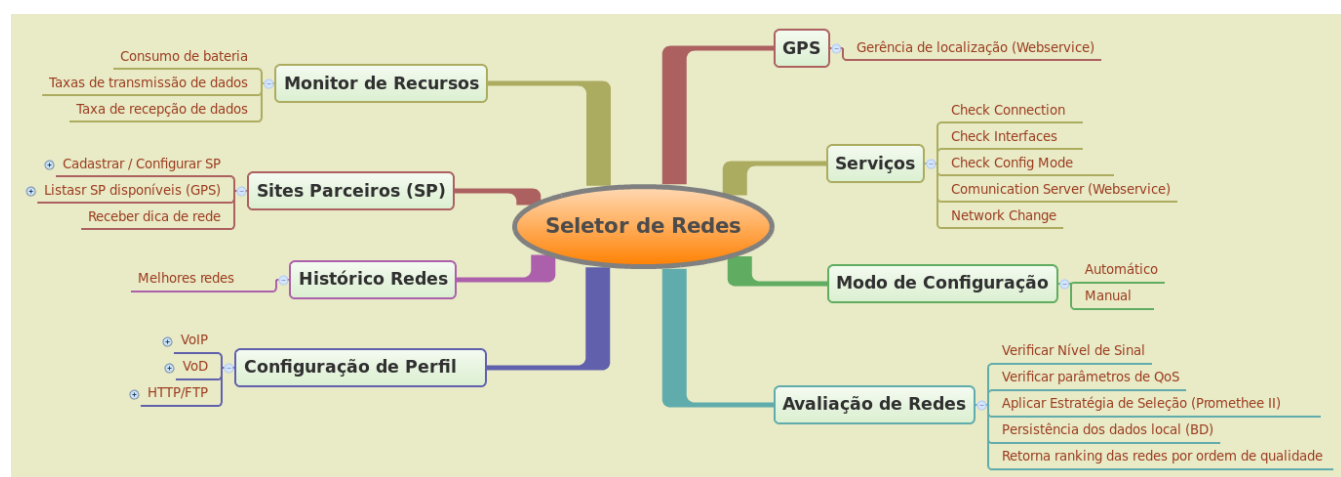
**Tabela 3.1.:** Percentual de vendas – SO Smartphones (IDC 2014).

Período	Android	iOS	Windows Phone	BlackBerry OS	Outros
Q3 2014	84.4%	11.7%	2.9%	0.5%	0.6%
Q3 2013	81.2%	12.8%	3.6%	1.7%	0.6%
Q3 2012	74.9%	14.4%	2.0%	4.1%	4.5%
Q3 2011	57.4%	13.8%	1.2%	9.6%	18%

Para o desenvolvimento da aplicação cliente, utilizou-se a API do Google para dispositivos móveis, chamada Android SDK (*Software Development Kit*)<sup>13</sup>, na sua versão 19. Além disso, utilizou-se a IDE (*Integrated Development Environment*) Eclipse e o plugin ADT (*Android Developer Tools*). Utilizou-se ainda a ferramenta tcpdump<sup>14</sup> embarcada para dispositivos androids. Esta última utilizada no processo de medições passivas.

Para o consumo do *webservice*, referente ao módulo de comunicação da parte servidora, fez-se uso da biblioteca Google GSON<sup>15</sup>. Esta biblioteca permite a leitura dos dados em formato JSON, possibilitando a conversão da representação lida em um objeto Java nativo, como por exemplo *ArrayList*, e vice-versa. Todas as ferramentas são de uso livre e gratuito.

O modelo com as funcionalidades providas pela aplicação android é apresentado na Fig. 3.3.

**Figura 3.3.:** Funcionalidades da aplicação Android.

<sup>13</sup><https://developer.android.com/sdk/index.html?hl=i>

<sup>14</sup><http://www.kandroid.org/online-pdk/guide/tcpdump.html>

<sup>15</sup><https://code.google.com/p/google-gson>

Note que a aplicação possui alguns requisitos (*RM*) funcionais e não funcionais, conforme ilustrado na Fig. 3.3 e descritos a seguir:

- RM1: Ser capaz de buscar as redes disponíveis (local e remoto), exibindo um *ranking* conforme sua qualidade;
- RM2: Solicitar ao site parceiro a opção de melhor rede, através de requisições HTTP e consumo a um *webservice*;
- RM3: Fazer a avaliação das redes disponíveis processando-as localmente (se o usuário desejar);
- RM4: Executar o *handover* para a melhor rede disponível;
- RM5: Permitir a configuração do perfil de consumo (preferências) do usuário, podendo ser: vídeo, áudio e dados;
- RM6: Executar em modo automático (serviço) ou manual, de acordo com o interesse do usuário.

Os módulos disponíveis na aplicação móvel se propõem a atender os requisitos citados anteriormente. Parte das funcionalidades ilustradas na Fig. 3.3, como Serviços, Avaliação de Redes, Configuração de Perfil e GPS, constituem partes funcionais do esquema proposto pela arquitetura, conforme apresentado na Fig. 3.2. Os demais módulos são para auxiliar os usuários finais, propiciando a utilização real da solução.

## Modelagem UML

A aplicação android (*app*) desenvolvida utiliza o padrão de arquitetura MVC (*Model View Controller*), e o padrão de projetos DAO (*Data Access Object*) (Freeman 2007). A mesma está dividida em seis pacotes principais, conforme ilustrado na Fig. 3.4. O pacote *view* depende das classes do pacote *controller*, e este por sua vez acessa as classes contidas nos pacotes *networkselector*, *model* e *service*. Dentro do pacote *model* existem os subpacotes *dao* e *entities*, sendo que o pacote *dao* depende das classes do pacote *entities*.

No pacote *view* estão contidas todas as telas do aplicativo, que são a interface de comunicação entre a *app* e o usuário. No pacote *controller* é realizado todo o processamento, como verificação de status de interface de rede, consulta por redes sem fio disponíveis, recuperação da rede atual selecionada, dentre outros. O pacote *network-*



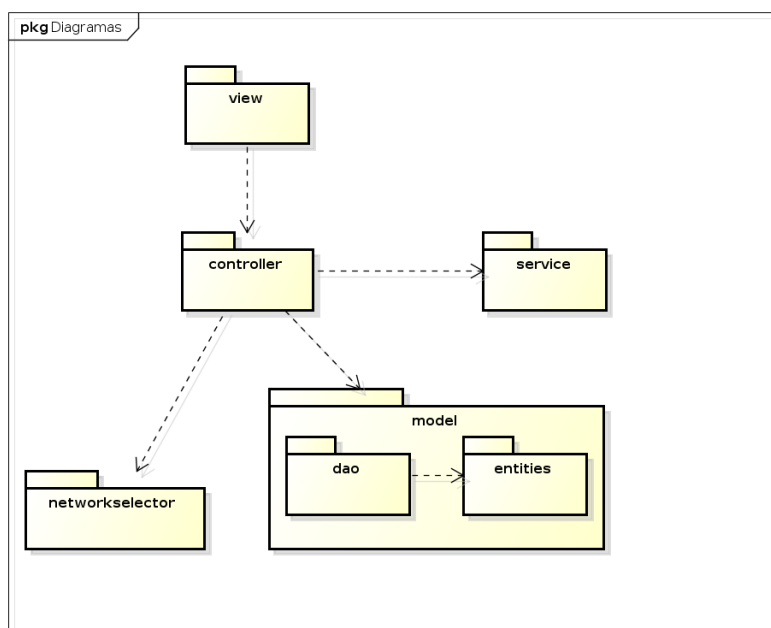


Figura 3.4.: Diagrama de Pacotes.

*selector* guarda as classes que implementam as métricas de seleção de redes, enquanto que os serviços realizados pela *app* estão contidas no pacote *service*, tais como serviço de consulta ao *webservice*, que recupera as informações sobre as melhores redes disponíveis informadas pelo site parceiro. No pacote *model* estão os subpacotes *dao* e *entities*, onde o *entities* guarda as entidades do sistema, enquanto o pacote *dao* utiliza as classes *entities* para persistir as mesmas em banco de dados (Shalloway & Trott 2004).

Para a demonstração do funcionamento dos requisitos outrora definidos para a *app*, utilizou-se recursos da linguagem UML (*Unified Modeling Language*) (Sommerville 2011). Na Fig. 3.5 apresenta-se o diagrama de casos de uso, que ilustra o funcionamento geral da aplicação no contexto da arquitetura proposta.

De maneira a permitir um detalhamento maior, outros diagramas UML foram utilizados. Na Fig. 3.6 é ilustrado o diagrama de estados, na qual a *app* desenvolvida tem dois estados de execução do serviço que faz a requisição ao *webservice*, sendo eles os modos automático e parado (manual).

Quando a aplicação é instalada no dispositivo móvel, a mesma é inicializada e entra no modo “automático”, onde são realizadas requisições a cada 30 segundos ao site parceiro, entrando no estado “parado” somente quando o usuário deseja executar o serviço manualmente. O serviço também pode ir ao estado “automático” quando ele estiver parado e o usuário desejar retorná-lo ao modo automático.

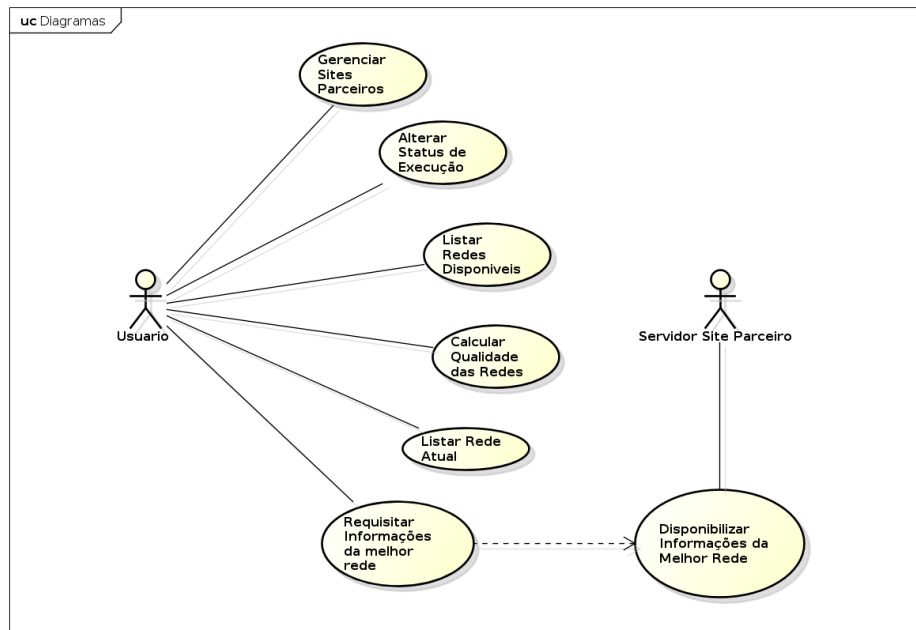


Figura 3.5.: Diagrama de Caso de Uso.

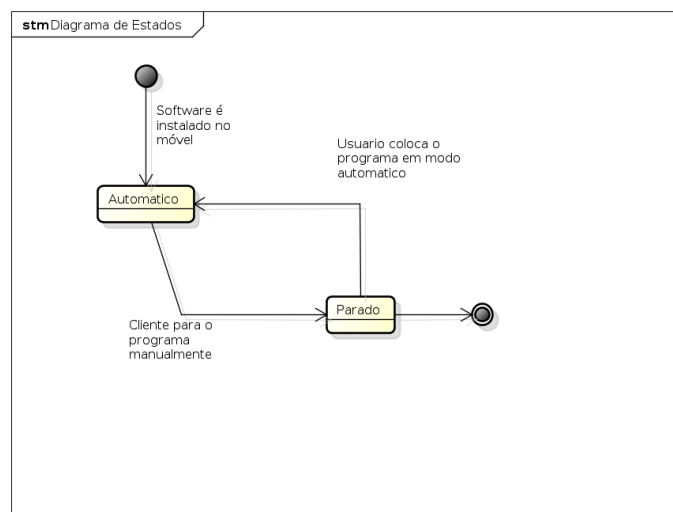


Figura 3.6.: Diagrama de estados.

O processo de requisição ao *webservice* para obtenção da melhor rede disponível ou do *ranking* das melhores redes, bem como o procedimento de *handover* realizado pelo móvel, é ilustrado na Fig. 3.7. Note que no momento em que o dispositivo móvel é ligado, o android lança internamente uma sinalização pública avisando que o processo de *boot* está completo. Nesse ponto, o aplicativo Seletor de Redes recebe essa sinalização e inicia o serviço de requisição.

Em seguida, a *app* solicita a melhor rede ao SP. Caso a rede indicada como a melhor já esteja em uso pelo dispositivo móvel, uma nova requisição é realizada após 30 segundos. Do contrário, a *app* verifica se o nível de sinal da rede indicada é superior ao limiar mínimo definido. Se for, a *app* realiza o *handover* para a rede indicada. Aguarda-se 30s e o processo é repetido.

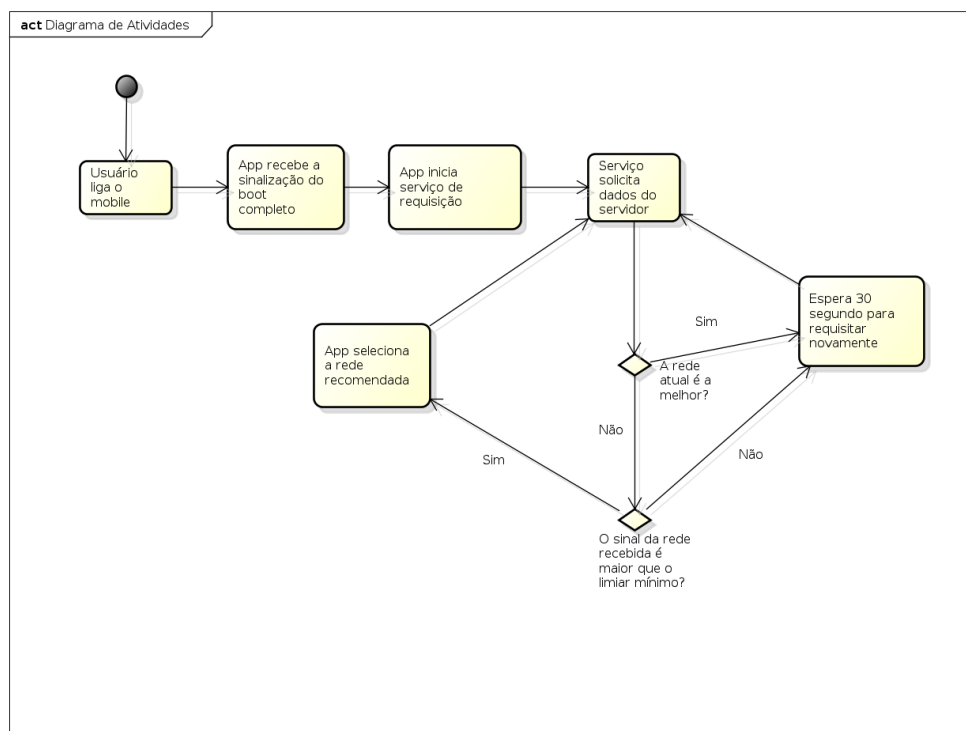


Figura 3.7.: Diagrama de atividades.

O diagrama de classes implementado no desenvolvimento da *app* é ilustrado na Fig. 3.8. Dessa forma é possível verificar como estão dispostos os pacotes, e como se dá a interação entre as classes. Para a confecção dos diagramas utilizou-se o software *Astah*

*Community*<sup>16</sup>, versão 6.9. No que tange ao SGBB utilizado na persistência de dados localmente na aplicação, utilizou-se o SQLite<sup>17</sup>, versão 3.4.0.

---

<sup>16</sup><http://astah.net/editions/community>

<sup>17</sup><http://developer.android.com/reference/android/database/sqlite/package-summary.html>



### 3.4. Principais Vantagens da Arquitetura Proposta

Uma correta seleção de redes pode ser utilizada para diversas finalidades. Como exemplo, poderia ser embarcada em um VANT (Veículo Aéreo Não-Tripulado), para que o mesmo possa se descolar a partir de um ponto inicial, realizar uma trajetória sem perder a conexão com a central de comandos, e fazer o retorno à base. Desta forma o VANT “passearia” por várias redes sem fio sem perder a conexão. Note que isso teoricamente já estaria resolvido utilizando uma rede de celular convencional, entretanto o *throughput* fornecido por essas redes costumam ser bem menores que os de uma rede WLAN, o que poderia inviabilizar, por exemplo, uma exibição *streaming* de vídeo realizada pelo VANT.

A aplicação na robótica também poderia ser de grande utilidade, em função da existência de problemas na navegação de robôs, quando da necessidade de grandes deslocamentos por ambientes *indoor* e *outdoor*.

Outro ponto bastante importante refere-se ao monitoramento de pacientes, no qual seria possível fazer o acompanhamento remoto das condições físicas de uma determinada pessoa, empregando equipamentos que façam uso de redes celulares de longa distância (3G e WiMax) e redes WLAN, garantindo sempre a melhor escolha para transmissão de dados, isto é, a rede que apresentar sempre as melhores condições de tráfego.

### 3.5. Escalabilidade da Arquitetura Proposta

A arquitetura proposta é escalável e pode ser utilizada com novos critérios para seleção da rede, bem como outras tecnologias de acesso, tais como 3G ou WiMax. A utilização de vários SPs, e a cooperação entre os mesmos, permite ainda que haja uma rede de informações sobre a qualidade das redes disponíveis em determinadas regiões (e.g., bairros, universidades, cidades).

A possibilidade de adição de novos módulos na arquitetura, como a implementação de algoritmos para seleção ou distribuição de APs em um dado ambiente, seria de fácil implantação. Além disso, é possível utilizar ferramentas para tornar os SPs tolerantes a falhas. Para isso, pode-se fazer uso de técnicas de redundância e alta disponibilidade *open source* diretamente nos sistemas operacionais dos SPs.

A aplicação móvel pode ser implementada facilmente para *smartphones* iOS ou *Windows Phone*. Basta para isso conhecimento sobre o funcionamento desses sistemas operacionais no que tange a gerência de interfaces de redes, visto que apenas o consumo das informações do *webservice* são necessárias.

Uma proposta bastante interessante é a utilização da aplicação cliente em *Smart TVs* e relógios inteligentes, que possuam o sistema operacional android. Entretanto, faz-se necessário portar a aplicação para estes dispositivos. A própria mantenedora do android disponibiliza a API para desenvolvimento para estes dispositivos.

### 3.6. Restrições da Arquitetura Proposta

As principais restrições da arquitetura referem-se a necessidade, mas não obrigatoriedade, de se utilizar uma placa *wireless*, padrão 802.11 b/g/n, contendo *chipset* da fabricante Atheros. Essa premissa se faz necessária em virtude desse *driver* possibilitar o uso de *interfaces* virtuais sobre a mesma *interface* física. Dessa forma, não é necessário, no momento de implantação do SP, a instalação de uma placa física por rede a ser monitorada.

Imagine que o intuito seja monitorar a qualidade de 20 redes distintas: com o dispositivo da fabricante supracitada e fazendo uso da solução proposta na parte servidora, faz-se necessário utilizar apenas uma placa física. Contudo, a solução é flexível e pode ser empregada usando fabricantes de placas de redes diferentes. Entretanto, é necessário uma placa *wireless* por rede a ser monitorada.

A parte servidora possibilita ainda o uso de apenas uma interface física, não *atheros*. No entanto, os dados coletados (valores dos critérios) não serão totalmente em tempo real, visto que ao “associar” a interface em cada uma das SSID (*Service Set Identification*) de todas as redes avaliadas, as condições de tráfego medidas na primeira rede do *round* de coletas podem estar defasadas, isto é, a rede pode ter sofrido alterações bruscas na sua qualidade no decorrer do tempo, e o que outrora fora coletado não representaria a condição real, ocasionando erro no processo de escolha.

Outra restrição, nesse caso do sistema operacional Android, e não da solução proposta, é a questão de permitir o *handover* via aplicação de terceiros somente para redes já visitadas. Assim, para que o móvel migre de uma rede para outra, tomando essa decisão via aplicação, é necessário que o sistema operacional em algum momento já tenha tido

acesso àquela rede, independente da rede ser aberta ou protegida via mecanismos de autenticação.

### 3.7. Conclusão

Neste capítulo apresentou-se a contribuição mais relevante deste trabalho. De maneira geral, propôs-se uma arquitetura composta por uma parte servidora (SP) e outra cliente (aplicação móvel) para o problema de seleção de redes. A arquitetura proposta neste capítulo permite sua aplicação em ambientes reais e em redes que já estejam em produção, oferecendo baixo consumo de recursos para os dispositivos móveis.

Os requisitos definidos tanto para o *framework* de seleção no SP, quanto para a aplicação móvel, foram atendidos durante o desenvolvimento deste trabalho.

Com o objetivo de validar a arquitetura proposta, o capítulo seguinte dedica-se ao detalhamento dos testes realizados e descrição dos cenários avaliados. O objetivo é medir a eficiência da solução proposta.



# Capítulo 4.

## Metodologia

### 4.1. Introdução

Após a implementação da arquitetura detalhada no capítulo anterior, partiu-se para a definição de alguns cenários de teste, de maneira a validar a arquitetura. Neste momento, o objetivo é verificar a similaridade entre o observado pelo site parceiro e pelo dispositivo móvel. Conforme comentado anteriormente, sabe-se que já é reconhecido na literatura que a decisão sobre a melhor rede tem que partir do dispositivo móvel, contudo o custo computacional, principalmente referente ao consumo de energia, direcionou os esforços deste trabalho para evitar a realização de medições constantes no móvel. Diante disto, buscou-se então mensurar a diferença entre as observações feitas no móvel e no site parceiro, na tentativa de validar a solução proposta.

Assim, propôs-se um cenário de teste incluindo injeção de diferentes taxas de tráfego nas redes avaliadas. O propósito é criar um ambiente de *stress* em alguns momentos, visando observar a resiliência (tempo de resposta) da arquitetura sob estas condições.

Por fim, os experimentos realizados permitiram i) avaliar a eficiência da arquitetura, no que tange ao tempo de resposta ao usuário, possibilitando que o móvel ficasse sem conexão de dados pelo menor tempo possível, e ii) verificar a quantidade de *handovers* para cada uma das redes, do ponto de vista do móvel e do site parceiro, determinando os valores dos parâmetros no momento da mudança.

## 4.2. Cenário de testes sem mobilidade

A montagem e realização dos cenários de experimentos sem mobilidade envolveu a utilização de três pontos de acesso WLAN IEEE 802.11g; um *host* com a função do Site Parceiro (SP), com dois processadores AMD Phenom II X2 550, 4GB de memória RAM e disco rígido de 500GB; sistema operacional linux Ubuntu Server 12.04 LTS<sup>1</sup>; três placas de rede sem fio USB TP-Link, modelo TL-WN722N com *throughput* de até 150Mbps e antena externa acoplada de 4dBi. Nesta interface que contém *chipset* da atheros (*AR9271*), a intensidade do nível de sinal é medido na escala entre 0 e 70. A Fig. 4.1 apresenta o cenário utilizado na realização dos experimentos.

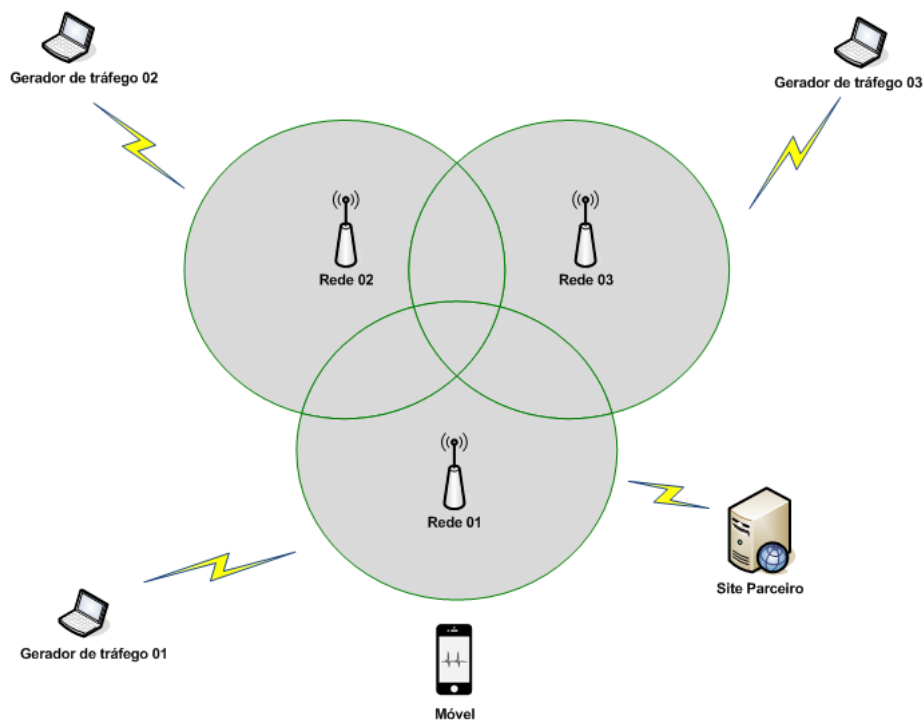


Figura 4.1.: Cenário dos Experimentos.

<sup>1</sup><http://www.ubuntu.com/server>

No cenário ilustrado na Fig. 4.1, o site parceiro e o dispositivo móvel estão posicionados em distâncias distintas em relação aos APs. Os APs da Rede 1 e Rede 3 encontram-se na mesma bancada física, enquanto que o AP da Rede 2 encontra-se em outra bancada, localizada a uma distância de 4.5 metros, separados por uma divisória de gesso acartonado.

Com intuito de prover maior capacidade de vazão para as redes avaliadas, bem como maior poder de processamento aos *Access Points* durante a saturação dos seus *links*, utilizou-se três *hosts* para desenvolver as funções dos APs, possuindo a configuração detalhada na Tabela 4.1.

**Tabela 4.1.:** Configuração de *hardware* dos APs.

Configuração	AP Rede1	AP Rede2	AP Rede3
Processador	AMD Athlon 64 X2 5200+	AMD Phenom II X2 550	Intel Atom N450
Memória RAM	2GB	4GB	2GB
Disco rígido	150GB	300GB	150GB
Sistema Operacional	Ubuntu Server 12.04 LTS	Ubuntu Server 12.04 LTS	Ubuntu Server 12.04 LTS
Interface <i>Wireless</i>	TP-Link USB TL-WN422G	D-Link N150 RT3060	Encore USB N300

O serviço de rede sem fio, provido pelos APs, fez uso da aplicação *hostapd*<sup>2</sup>, que de maneira geral pode ser explicado como sendo um conjunto de *drivers* que provê as funcionalidades de acesso sem fio, implementando várias especificações do padrão IEEE 802.11 (b/g/n). O *hostpad* suporta diversos serviços de autenticação e criptografia, tais como: WEP (*Wired Equivalent Privacy*), WPA/WPA2 (*Wi-fi Protected Access*), EAP (*Extensible Authentication Protocol*) e RADIUS (*Remote Authentication Dial In User Service*). A configuração de *hardware* provida pelos *hosts* é superior a várias marcas e modelos de APs comerciais para redes de médio e pequeno porte. Convém salientar que, no ambiente montado, foram respeitadas questões inerentes a sobreposição de canais, por parte das redes implementadas. Além disto, não foram realizadas alterações nos limites de *buffer* de transmissão e recepção nos sistemas operacionais dos *hosts* com as funções de APs, bem como implementação de políticas de priorização de tráfego ou classes de serviços, tais como HTB (*Hierarchy Token Bucket*), CBQ (*Class Based Queueing*), PRIO (*Priority Token Bucket*) e ALTQ (*Alternate Queueing*).

Por questões de limitação na quantidade de interfaces de rede 802.11 nos *smartphones* e *tablets*, para a validação da proposta a entidade móvel utilizada foi um *netbook* Asus

<sup>2</sup><http://wireless.kernel.org/en/users/Documentation/hostapd>

Eee PC, com processador Intel Atom de 1.66GHz, 2GB de memória RAM e disco rígido de 160GB, usando o sistema operacional Ubuntu Desktop 12.04 LTS, possuindo também três placas de rede USB TP-Link, modelo TL-WN722N. Essa mudança se fez necessária para que fosse possível avaliar em tempo real e de maneira simultânea o comportamento das redes.

#### 4.2.1. Definição dos parâmetros dos algoritmos

Foram realizadas medições nos turnos matutino, vespertino e noturno. Cada medição ou teste possui 99 iterações, sendo que cada iteração utiliza um tempo de captura dos parâmetros de QoS (*delay*, *jitter* e perda de pacotes) de 20 segundos, intervalos de medição do nível de sinal a cada 30 segundos, e intervalo de decisão ou aplicação do método Promethee II a cada 25 segundos.

Esses valores são configuráveis nos algoritmos do *framework* desenvolvido na parte servidora e cliente da arquitetura. Conforme comentado no capítulo 3, após a captura dos dados os mesmos são processados e a decisão da melhor rede é tomada, totalizando um tempo arredondado de 41 minutos por teste (conjunto de 99 iterações).

Para cada iteração foram coletados os valores de *delay*, *jitter*, perda de pacotes e RSSI, para em seguida serem calculados a média, desvio padrão, variância e os intervalos de confiança de cada parâmetro considerado em cada uma das redes avaliadas. Ao final de cada iteração a melhor rede é escolhida do ponto de vista do site parceiro e do móvel.

Os vetores de peso considerados para os critérios de *delay*, *jitter*, perda de pacotes e nível de sinal foram os mesmos, sendo 0.25 para cada um deles, conforme parametrização necessária para a utilização do método Promethee II, isto é, o somatório dos pesos precisa ser igual a 1. A ideia é realizar uma comparação sem tendenciosidade entre os critérios, visto que neste momento não existem requisitos de qualidade de rede exigidos por uma aplicação específica, diferente do que é apresentado em (Monteiro 2012) e (Rios et al. 2012), onde foram priorizados um tráfego de vídeo sob demanda (VoD) e de áudio (VoIP), respetivamente.

Em resumo, a configuração dos parâmetros definidos nos algoritmos para a realização dos testes é apresentada na Tabela 4.2.

**Tabela 4.2.:** Parâmetros de configuração dos algoritmos.

Descrição	Parâmetros
Quantidade de avaliações	99
Intervalo de cada avaliação (Decisão)	25s
Intervalo de coleta QoS	20s
Intervalo de coleta RSSI	30s
Vetor de pesos	[0.25,0.25,0.25,0.25]
Tráfego injetado nas redes	Tabela 4.3
Tempo do teste	41 min

O intervalo de verificação do parâmetro RSSI, fixado em 30 segundos, foi escolhido considerando o fato de que o uso repetitivo e sucessivo do método *scan* nas interfaces de rede tornava a *interface* “ocupada” em alguns momentos. Em virtude disso, o módulo do *framework* que utiliza o comando *iw*<sup>3</sup> era executado sem sucesso.

Em (Han et al. 2012) é realizado um estudo que avalia, para um modelo específico de dispositivo móvel, o consumo de bateria quando a *interface wireless* permanece em modo *scanning*. Os autores, que também utilizam o comando *iwlist*, mostram que o consumo de bateria no móvel com verificações a cada 1s ou 30s é praticamente o mesmo, ocasionando redução do tempo de carga da bateria de 310 horas, para apenas 5 horas no móvel. Essa característica serviu de base para a escolha do intervalo de medição do nível de sinal para validação da proposta.

Desta forma, todos os parâmetros podem ser reconfigurados, conforme pode ser visualizado nos Algoritmos 3.1, 3.2 e 3.3.

#### 4.2.2. Estratégias adotadas na validação

Nos testes iniciais, para verificação do funcionamento da arquitetura proposta, os critérios utilizados no processo de decisão foram obtidos utilizando técnicas de medições passivas, fazendo uso de analisadores de protocolos (*sniffers*). Entretanto, a implementação da arquitetura ficou bastante onerosa, pois havia a necessidade da configuração de uma instância *sniffer* em cada uma das interfaces das entidades envolvidas, ou seja, cada interface de rede dos dispositivos precisaria ser monitorada via uma instância *snif-*

<sup>3</sup><http://wireless.kernel.org/en/users/Documentation/iw>

*fer*, usando ferramentas que faziam uso da biblioteca *libpcap* (*wireshark*<sup>4</sup> e *tcpdump*<sup>5</sup>), conforme descrito no módulo de coletas, detalhado no capítulo 3.

A grande dificuldade no momento de consolidar os dados das medições envolve a leitura dos extensos *traces* e arquivos de *logs* gerados, mesmo utilizando filtros para detalhamento somente do tráfego desejado. Por exemplo, para medições dos parâmetros *delay* e *jitter* é necessário trabalhar com algoritmos de sincronização de relógios nas várias entidades (móvel, site parceiro, APs), a fim de se obter, com alguma precisão, os atrasos na recepção e na transmissão dos dados em cada uma das *interfaces*. Esta situação fica ainda mais delicada quando existe injeção de tráfego concorrente nas redes avaliadas. Toda essa problemática faz com que o processo de tomada de decisão não seja realizável em tempo real.

Como o propósito inicial deste trabalho é construir uma arquitetura aplicável em redes que já estejam em produção, optou-se por substituir o módulo de medição passiva pelo módulo de medição ativa, que usa requisições ICMP (*Internet Control Message Protocol*) através do uso de informações do tipo RTT (*Round-trip time*). Estas informações podem ser úteis para verificação de *overhead* em *links*, desde que sejam observados o contexto das redes que se deseja avaliar.

Para o cenário de teste neste trabalho, as *interfaces* de rede tanto do site parceiro quanto do móvel estão diretamente associadas aos APs provedores das redes sem fio IEEE 802.11, ou seja, é realizado apenas um *hop* (salto) do móvel para o AP ou do site parceiro para o AP. Esse esquema é representado na Fig. 4.1.

Para os casos onde a arquitetura for utilizada para monitorar redes de abrangência WMAN, é recomendável, porém não obrigatório, que exista, na infra-estrutura de rede, uma política de priorização de tráfego, onde os datagramas ICMP, vindos dos sites parceiros, possam ter a mesma prioridade, visando não inibir as medições necessárias.

Outro ponto bastante importante, que se deve considerar ao usar o módulo de medição ativa, refere-se a necessidade de liberação nos APs para aceitarem requisições ICMP. Por questões de segurança, o gestor da rede poderá optar em fazer a liberação apenas para o site parceiro, nos seus equipamentos de segurança (*firewall*, *proxy*, etc) ou de enlace (*switches*).

---

<sup>4</sup><https://www.wireshark.org>

<sup>5</sup><http://www.tcpdump.org>

Assim, embora as medições ativas gerem tráfego desnecessário na rede (Tanenbaum 2003), esta abordagem torna a arquitetura mais flexível, visto que não é necessário o acréscimo de *sniffers* em cada uma das entidades geradoras e receptoras de tráfego. Contudo, o módulo passivo está disponível na arquitetura e pode ser utilizado pelo gestor da rede que implantará.

### 4.2.3. Injeção de tráfego

De maneira a possibilitar a avaliação do funcionamento da arquitetura sob diversas situações de *stress*, nas redes que compõem o cenário da Fig. 4.1, utilizou-se a ferramenta Iperf<sup>6</sup>, versão 2.0.5, para geração de tráfego em duas das redes consideradas, deixando a terceira rede em produção. O *iperf* funciona em modo cliente/servidor, sendo que a parte cliente da aplicação é desempenhada pelos *hosts* geradores de tráfego, e a servidora pelos APs das redes implementadas. Nesse ponto, o intuito é verificar a resiliência da arquitetura sob diferentes tráfegos. A taxa de transmissão para cada conjunto de testes é apresentado na Tabela 4.3.

**Tabela 4.3.:** Tráfego injetado nos conjuntos testes.

Conjunto	Cliente <i>iperf</i>	Servidor <i>iperf</i>	Taxas de transmissão
01	Gerador de tráfego 01, 02 e 03	Rede 01 e 02	200Kbit/s
02	Gerador de tráfego 01, 02 e 03	Rede 01 e 02	400Kbit/s
03	Gerador de tráfego 01, 02 e 03	Rede 01 e 02	600Kbit/s
04	Gerador de tráfego 01, 02 e 03	Rede 01 e 02	800Kbit/s
05	Gerador de tráfego 01, 02 e 03	Rede 01 e 02	1Mbit/s
06	Gerador de tráfego 01	Rede 01 e 02	2Mbit/s
07	Gerador de tráfego 01	Rede 01 e 02	5Mbit/s
08	Gerador de tráfego 01	Rede 01 e 02	8Mbit/s
09	Gerador de tráfego 01	Rede 01 e 02	11Mbit/s

O tráfego injetado nas Redes 01 e 02 é do tipo CBR (*Constant Bit Rate*) (Kurose & Ross 2013), (Costa 2009). Além disto, foram realizadas medições da largura de banda de forma bidirecional e simultânea, ou seja, o cliente *iperf* (gerador de tráfego 01, 02 e 03) utilizou portas diferentes para envio e recebimento dos dados para o servidor *iperf* (AP WLAN), simulando assim uma comunicação do tipo “*duplex*” (Silva & Junior 2014).

<sup>6</sup><https://iperf.fr>

O objetivo é impor uma situação de *stress* nos APs, visto que em condições de tráfego normais a geração e consumo de dados em uma determinada *interface* não se mantém fixo ao longo do tempo. Convém salientar que, em virtude do UDP não possuir controle de fluxo ou de congestionamento, assim como não fornecer transmissão confiável de dados, não implementou-se nenhum mecanismo adicional de checagem de recepção dos pacotes nos APs. Neste contexto, todos os valores dos parâmetros de QoS utilizados foram obtidos diretamente pela resposta nativa da pilha TCP/IP de cada rede avaliada.

A partir da Tabela 4.3 observa-se ainda que, para taxas de transmissões superiores a 1Mbit/s, optou-se por utilizar apenas um gerador de tráfego, no qual injetava fluxos nas duas redes. Esse procedimento se fez necessário em virtude do padrão IEEE 802.11, através da função DCF (*Distributed Coordination Function*) e do método CSMA/CA (*Carrier Sense Multiple Access with Collision Avoidance*), permitir que as estações tenham condições iguais de acesso ao meio (Kurose & Ross 2013). Entretanto, devido ao contexto de *stress* dos APs (tratamento de *buffer*), em razão do fluxo contínuo UDP nos dois sentidos, a taxa de transmissão é multiplicada por dois em cada enlace. Nesse sentido, caso três geradores estivessem injetando tráfego a uma taxa de 5Mbit/s (ida e volta 10Mbit/s) para uma mesma rede, o tráfego total injetado seria de 30Mbit/s. Esse valor é superior ao *throughput* de 20Mbit/s alcançados por uma rede 802.11g, mesmo com um nível de sinal acima de 81% (Siris & Anagnostopoulou 2013).

Outro problema de se ter inúmeras estações transmitindo fluxos contínuos ao mesmo tempo, é que ocorre variação nas taxas de transferência, devido a aspectos inerentes às próprias características do tipo de enlace físico utilizado. Isto influencia na vazão total da rede, que é seriamente comprometida pela estação que estiver transmitindo em uma taxa bem menor que as demais, gerando um problema conhecido como anomalia de MAC (Marques 2013). Nestas circunstâncias, a estação com baixa taxa de transmissão detém por mais tempo o acesso ao meio, até concluir a sua transmissão (rajada UDP), o que degrada a qualidade da rede em questão.

Assim, todas as características apresentadas nesta seção visam impor um ambiente de “dificuldade” para a arquitetura proposta, de forma a testá-la e verificar o nível de eficiência ofertada pela mesma.



### 4.3. Cenário de testes com mobilidade

Os experimentos envolvendo mobilidade desejam verificar qual o tempo de realização total do *handover*. Assim, foi considerado o tempo que o móvel levaria para fazer uma requisição ao *webservice*, receber as informações da melhor rede disponível, verificar se a rede está ao alcance e acima de um limiar mínimo de RSSI (35%), de modo a não perder o enlace físico, e por conseguinte fazer a mudança de rede.

Para este caso específico de testes, foi necessário que o móvel tivesse duas *interfaces* habilitadas, isto é, quando houvesse a perda de conexão da rede *wireless* (*hard handover*) (Monteiro 2012) seria necessário fazer a requisição, usando por exemplo a interface 3G para receber a dica de rede do site parceiro e migrar para nova rede *wireless* disponível.

O cenário com essa alteração é apresentado na Fig. 4.2.

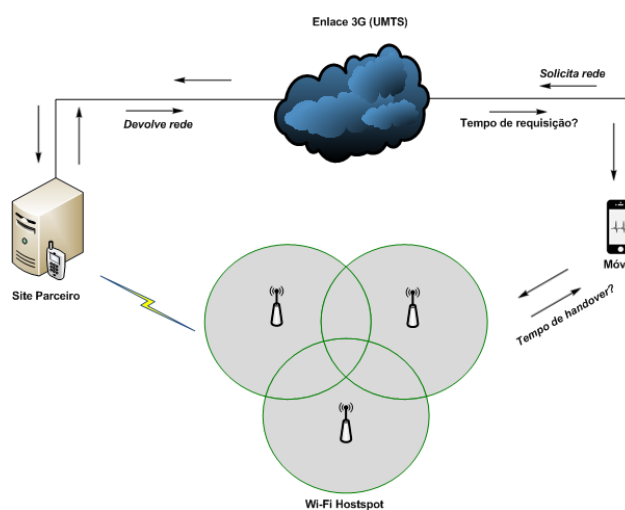


Figura 4.2.: Cenário dos Experimentos com Mobilidade.

Observa-se na Fig. 4.2 que o tempo total para realização do *handover* é dado pela soma do tempo de requisição com o tempo de troca da rede realizada no dispositivo. Para os casos onde o dispositivo móvel não possua uma segunda interface de rede, a aplicação móvel trata isso internamente através de mecanismos de consulta à base local.

Nesse conjunto de testes utilizou-se um *smartphone* da fabricante LG, modelo L65, com o android versão 4.4.2 e kernel 3.4.0+. Para este conjunto de testes, o móvel encontrava-se com a interface *wireless* desconectada, e as requisições ao site parceiro eram realizadas utilizando uma conexão 3G de uma operadora de celular local. O conjunto de testes não considerou para o deslocamento do móvel a velocidade do portador do

dispositivo, nem mesmo a distância real para o AP, considerando-se apenas que o móvel encontrava-se dentro da área de cobertura das redes avaliadas (as redes eram “enxergadas” pelo *smartphone*).

## 4.4. Conclusão

Neste capítulo foram apresentados os cenários de testes usados para realizar a validação da arquitetura, de maneira a verificar a capacidade de resiliência da mesma, e a sua eficiência em situações que envolvem *stress* das redes avaliadas.

Os experimentos foram realizados em ambiente de bancada, buscando utilizar características heterogêneas de *hardware*, visando desta forma prover um ambiente o mais próximo possível de redes em produção, tornando a aplicabilidade da arquitetura real.

# Capítulo 5.

## Resultados

### 5.1. Introdução

Neste capítulo são detalhados os resultados obtidos para os cenários de experimentos descritos no capítulo 4. Inicialmente são detalhados os dados obtidos nos experimentos, os testes estatísticos aplicados e, por fim, a análise e discussões sobre os resultados obtidos para cada um dos cenários considerados.

### 5.2. Resultados dos testes sem mobilidade

Os resultados dos testes realizados para este cenário revelam o percentual de vezes que cada uma das redes foi selecionada por cada uma das entidades da arquitetura, SP e móvel, ao longo das iterações. Além disso, avaliou-se os valores dos critérios utilizados no processo de tomada de decisão, visando aferir com algum nível de certeza qual rede seria selecionada como a melhor, sob as mesmas condições de tráfego.

É importante salientar que cada teste é constituído de 99 iterações, e seguem as configurações definidas nas Tabelas 4.2 e 4.3.

A Tabela 5.1 apresenta o percentual de escolha de cada uma das redes avaliadas, em cada um dos testes realizados do ponto de vista do SP. Note que, para o conjunto dos 9 testes realizados são calculados a média, desvio padrão (STD), variância (VAR) e intervalo de confiança para média (IC), ao nível de confiança de 95%.

**Tabela 5.1.:** Seleção de Rede – Site Parceiro.

Teste	Rede 1 (%)	Rede 2 (%)	Rede 3 (%)
1	75.76	22.22	2.02
2	80.61	18.37	1.02
3	64.65	33.33	2.02
4	62.63	30.3	7.07
5	61.96	21.74	16.3
6	72.63	11.58	15.79
7	48.98	24.49	26.53
8	32.65	25.51	41.84
9	3.03	28.28	68.69
—	—	—	—
<b>Média</b>	55.88	23.98	20.14
<b>STD</b>	24.56	6.54	22.62
<b>VAR</b>	603.39	42.73	511.95
<b>IC</b>	36.99 a 74.76	18.96 a 29.01	2.75 a 37.53

A Tabela 5.2 apresenta os mesmos dados de avaliação, porém do ponto de vista do dispositivo móvel.

Após a obtenção dos dados resultantes, sob o ponto de vista de cada uma das entidades da arquitetura proposta, partiu-se para uma análise descritiva, visando verificar se há diferenças significativas entre o que é observado pelo SP e pelo móvel, para um mesmo cenário de avaliação e sob diversas condições de tráfego.

Inicialmente, definiu-se as seguintes hipóteses:

$$\begin{aligned}
 H_0 &: \text{Não há diferenças entre o SP e o móvel (hipótese nula)} \\
 H_1 &: \text{Existe diferença entre o SP e o móvel (hipótese alternativa)}
 \end{aligned}
 \tag{5.1}$$

Tabela 5.2.: Seleção de Rede – Móvel.

Teste	Rede 1 (%)	Rede 2 (%)	Rede 3 (%)
1	89.9	7.07	3.03
2	79.38	16.49	4.13
3	69.39	26.53	4.08
4	64.29	21.42	14.29
5	80.43	10.87	8.7
6	73.74	9.09	17.17
7	40.21	25.77	34.02
8	26.26	35.35	38.39
9	2.09	52.08	45.83
—	—	—	—
<b>Média</b>	58.41	22.74	18.85
<b>STD</b>	29.25	14.39	16.40
<b>VAR</b>	855.40	207.12	268.95
<b>IC</b>	35.93 a 80.89	11.74 a 33.80	6.24 a 31.45

De posse das hipóteses, foram realizados a leitura dos dados em arquivo texto através do Rstudio<sup>1</sup>, que é uma interface com diversos recursos visuais para o software R<sup>2</sup>. Em seguida, foram realizadas as inferências estatísticas (intervalo de confiança para o conjunto de amostras, valor-p, amplitude médio e os testes de normalidade e independência das amostras) considerando um nível de significância de 95%.

O experimento consistiu-se de um comparativo simples envolvendo o percentual de escolha de cada uma das redes, por parte do SP e do móvel. Para tal, aplicou-se o método de amostras *teste t* sobre o vetor diferença entre os grupos amostrais considerados (Montgomery & Runger 2011).

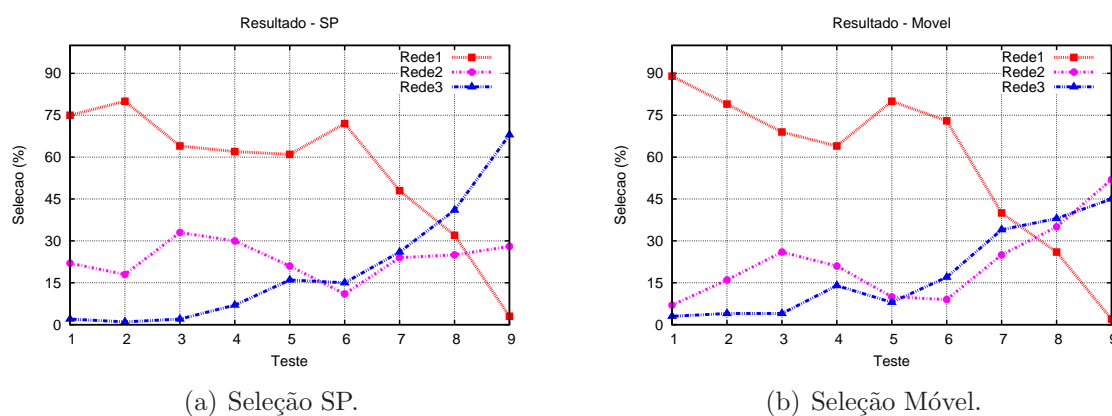
A partir dos resultados encontrados e considerando as restrições do cenário proposto, observou-se através dos testes de pareamento (*t-test*), que o *p-value* é maior que o nível de significância desejado ( $\alpha = 0.05$  ou 95%), para cada umas redes avaliadas. Assim sendo, não há evidências amostrais suficientes para rejeitar a hipótese nula. Dessa forma, conclui-se que não há diferença no processo de seleção realizado pelo site parceiro e pelo móvel, para o cenário de teste avaliado.

<sup>1</sup><http://www.rstudio.com/>

<sup>2</sup><http://www.r-project.org/>

Outro ponto refere-se aos resultados obtidos pelo teste de normalização de Shapiro-Wilk, onde os valores encontrados descrevem uma distribuição normal das amostras, isto é, têm-se que o  $p$ -value é bem maior que o nível de significância desejado, que é também de 95%. Pelo teste de Durbin-Watson tem-se que não há relação de dependência entre as variáveis amostrais, exceto para as amostras que fazem a comparação de seleção para a Rede 2, onde houve violação da premissa de independência (Montgomery & Runger 2011). Esse teste foi repetido várias vezes, e esse comportamento se manteve. As alterações bruscas de alguns parâmetros, como o *delay* para esta rede, contribuíram para que houvesse a violação. O comportamento de todos os critérios utilizados na tomada de decisão são destacados posteriormente neste capítulo.

Essa característica evidencia a eficiência da arquitetura proposta, o que justifica levar de forma permanente o processo de tomada de decisão acerca da escolha da melhor rede para um servidor (seleção centrada na rede), nesta proposta nomeado como site parceiro. Obviamente se faz necessário testes em vários outros cenários, dada a grande heterogeneidade dos ambientes que utilizam redes sem fio. Contudo, a premissa de que não há diferença significativa entre o que é observado pelo SP e o móvel foi validada.



**Figura 5.1.:** Seleção realizada pelo SP e Móvel.

A Fig. 5.1 consolida os resultados apresentados nas Tabelas 5.1 e 5.2. Note que a partir do teste 7, a Rede 1 tem sua proporção de escolha inferior as demais redes. Esse comportamento de maneira geral representa o “gargalo” do AP para tráfego contínuo acima de 5Mbit/s. Contudo, para entender a tomada decisão por parte dos dispositivos, através do método Promethee II, é preciso verificar os valores dos critérios para cada umas das redes, em cada um dos testes realizados.

Dessa forma, faz-se a análise do critério *delay* nas duas entidades. As Tabelas 5.3 e 5.4 apresentam os valores desses parâmetros. Convém destacar que considerou-se até 06 casas decimais em alguns testes, em virtude dos valores obtidos serem muito pequenos e se diferenciarem apenas a partir da quarta casa decimal. Essa característica influenciou também na escolha das curvas de preferência utilizadas pelo método Promethee II, conforme comentado anteriormente.

**Tabela 5.3.:** Valores do critério Delay: SP.

Delay (ms)	Teste	Média	Max	Min	STD	Var	IC
Rede 1	1	7.671245	54.30842	0.9213588	10.68636	114.1983	5.539888 a 9.802602
	2	3.858498	83.12962	0.7487568	9.261113	85.76821	2.011402 a 5.705595
	3	5.336928	29.85116	0.9130253	7.855227	61.70458	3.770231 a 6.903625
	4	7.598368	95.75148	1.05839	13.53419	183.1743	4.899021 a 10.297715
	5	11.33666	58.34226	1.044122	15.41301	237.5608	8.213697 a 14.459628
	6	3.534516	18.65103	0.6453832	5.556811	30.87815	2.408602 a 4.660431
	7	2.425589	23.02786	0.7034175	4.595754	21.12096	1.508982 a 3.342196
	8	4.469583	24.36451	0.7993212	6.553631	42.95008	3.155663 a 5.783503
	9	6.427544	27.14174	0.9859138	6.75955	45.69152	5.079375 a 7.775712
Delay (ms)	Teste	Média	Max	Min	STD	Var	IC
Rede 2	1	8.761964	67.10085	1.267502	10.92928	119.4491	6.582158 a 10.941771
	2	7.516344	177.9472	1.092348	22.49248	505.9119	3.030297 a 12.002391
	3	2.234341	27.65694	0.8296682	4.211386	17.73577	1.394395 a 3.074287
	4	3.51675	81.32516	0.9507788	9.122631	83.22239	1.697273 a 5.336226
	5	64.05162	638.4551	1.002579	129.8629	16864.38	37.73895 a 90.36429
	6	142.5679	481.8009	0.58691	150.1657	22549.75	112.1415 a 172.9943
	7	302.9441	1301.708	0.6318075	400.913	160731.3	222.9834 a 382.9048
	8	18.39114	546.5412	0.7920542	81.3654	6620.328	2.078404 a 34.703866
	9	2.897578	52.31273	0.6525287	6.777717	45.93744	1.545786 a 4.249369
Delay (ms)	Teste	Média	Max	Min	STD	Var	IC
Rede 3	1	99.09841	437.9471	1.664772	85.4814	7307.07	82.04945 a 116.14738
	2	13.61123	185.9595	1.830393	35.67606	1272.781	6.495765 a 20.726693
	3	5.346091	128.1034	1.492345	13.83495	191.406	2.586758 a 8.105424
	4	4.773494	23.0359	1.943483	3.36978	11.35542	4.101403 a 5.445585
	5	26.61942	327.9175	0.1682913	44.05226	1940.601	17.69361 a 35.54524
	6	17.7255	79.20543	1.090914	18.38556	338.0287	14.00024 a 21.45076
	7	16.02011	174.2913	0.738305	32.4708	1054.353	9.543926 a 22.496300
	8	1.379302	16.48232	0.8069538	2.042079	4.170088	0.9698911 a 1.7887131
	9	2.569538	12.78718	0.7162787	2.767806	7.660752	2.017509 a 3.121568

Observe que o tempo médio do *delay* (em milissegundos) para os testes 5, 6 e 7 são bastante altos, devido ao aumento gradativo da inserção de tráfego pelo *iperf*. Esse comportamento é percebido por ambas as entidades.

Para o critério *jitter*, apresentado nas Tabelas 5.5 e 5.6, percebe-se que a variação de atraso para os testes 5, 6 e 7 também é mais acentuada, visto que está associada ao aumento do *delay*. Conseqüentemente, há um aumento da perda de pacotes em valores absolutos (número de pacotes) nestes mesmos testes para a Rede 1 e a Rede 2.

Conforme indicado nas Tabelas 5.7 e 5.8, percebe-se ainda que embora com uma taxa de transmissão bastante alta, como é o caso dos testes 8 e 9, com taxas de transmissão de 8 Mbit/s e 11 Mbit/s, respectivamente, o número de pacotes perdidos não é tão

**Tabela 5.4.:** Valores do critério Delay: Móvel.

Delay (ms)	Teste	Média	Max	Min	STD	Var	IC
Rede 1	1	2.980892	17.19517	0.5507275	4.874543	23.76117	2.008682 a 3.953102
	2	3.681415	24.40619	0.5557363	6.116836	37.41568	2.461434 a 4.901397
	3	4.492161	22.28646	0.55418	6.525992	42.58857	3.190575 a 5.793748
	4	5.505865	23.80026	0.5799825	7.036993	49.51927	4.095037 a 6.916693
	5	4.412756	31.37122	0.6374287	6.822139	46.54159	3.052105 a 5.773408
	6	2.804547	21.83919	0.59303	5.115479	26.16812	1.784282 a 3.824811
	7	8.074277	21.1921	0.6670525	7.683406	59.03473	6.541849 a 9.606705
	8	4.490357	31.41584	0.7239497	6.944066	48.22006	3.105387 a 5.875327
	9	6.453437	29.39004	1.007002	8.156203	66.52365	4.826711 a 8.080163
Delay (ms)	Teste	Média	Max	Min	STD	Var	IC
Rede 2	1	12.07758	167.9529	0.910335	19.75712	390.3439	8.137094 a 16.018072
	2	7.16901	407.1522	0.6579066	40.76187	1661.53	-0.9608021 a 15.2988222
	3	1.342973	11.28082	0.596419	1.460698	2.133638	1.051642 a 1.634304
	4	14.55338	434.6996	0.7349338	69.33437	4807.255	0.6527164 a 28.4540396
	5	204.6089	2509.308	0.8312888	598.7894	358548.7	85.18248 a 324.03536
	6	565.3892	1644.729	0.60676	680.4374	462995	429.6784 a 701.1001
	7	351.0588	1633.051	0.4936592	637.1127	405912.7	223.9889 a 478.1287
	8	1.265912	17.82598	0.6111483	2.272242	5.163084	0.8127213 a 1.7191027
	9	17.43209	816.6958	0.5268637	109.3679	11961.33	-4.380941 a 39.245130
Delay (ms)	Teste	Média	Max	Min	STD	Var	IC
Rede 3	1	77.4464	347.2208	0.9510112	77.44457	5997.661	62.00036 a 92.89245
	2	9.57732	147.6512	0.894345	26.04545	678.3653	4.382647 a 14.771992
	3	2.582079	60.23969	0.8379475	6.435207	41.41189	1.298600 a 3.865559
	4	1.932729	6.553183	0.8692732	1.048454	1.099256	1.722527 a 2.142930
	5	81.36044	627.9417	0.9798275	147.2194	21673.56	51.99804 a 110.72284
	6	2.993916	9.956233	0.7491625	2.767792	7.660673	2.441890 a 3.545942
	7	1.413839	3.343606	0.745952	0.8972581	0.8050722	1.234884 a 1.592794
	8	0.9255748	2.560568	0.761265	0.2769485	0.07670046	0.8703384 a 0.9808112
	9	0.9804416	3.859425	0.7304099	0.6451329	0.4161965	0.8517721 a 1.1091111



expressivo. Isso acontece devido aos próprios mecanismos de gerência da janela de transmissão, ofertado pelo algoritmo DCF e pelo método CSMA/CA (*frames beacon*) do padrão 802.11. Além disso, devido ao tamanho dos pacotes ICMP serem pequenos (64 *bytes*), o tempo da janela de transmissão dedicada ao móvel e ao SP eram suficientes para a transmissão dos pacotes, recebendo as confirmações por parte dos APs (Kurose & Ross 2013). Esse comportamento é evidenciado na Rede 3, onde não há injeção de tráfego adicional. Entretanto, por ser uma rede em produção (não controlada nos testes), a disputa pelo “direito” de transmitir é maior entre os dispositivos, acarretando em alguns testes o aumento da perda de pacotes.

**Tabela 5.5.:** Valores do critério Jitter: SP.

Jitter (ms)	Teste	Média	Max	Min	STD	Var	IC
Rede 1	1	2.442477	9.826231	0.6438462	1.53132	2.344941	2.137060 a 2.747893
	2	0.971609	8.40652	0.453493	0.9574784	0.9167648	0.7806433 a 1.1625747
	3	1.310994	4.47633	0.6168072	0.9804324	0.9612477	1.115450 a 1.506537
	4	1.670829	8.7682	0.7547005	1.50281	2.258438	1.371098 a 1.970559
	5	2.074713	5.909236	0.7633495	1.693519	2.868006	1.731574 a 2.417852
	6	0.7529059	3.593197	0.268875	0.7745616	0.5999456	0.5959652 a 0.9098467
	7	0.6346628	3.060654	0.3179016	0.5554578	0.3085333	0.5238787 a 0.7454469
	8	0.9378194	3.98729	0.2901175	0.8508458	0.7239385	0.7672356 a 1.1084032
	9	1.223658	3.794116	0.3777262	0.8227476	0.6769136	1.059563 a 1.387752
Jitter (ms)	Teste	Média	Max	Min	STD	Var	IC
Rede 2	1	2.966144	7.288209	1.015154	1.401178	1.9633	2.686684 a 3.245605
	2	1.731036	7.420868	0.8090063	1.164886	1.35696	1.498704 a 1.963369
	3	1.065724	4.535598	0.641404	0.5149415	0.2651647	0.963021 a 1.168427
	4	1.319592	9.375724	0.6488862	0.9204336	0.8471981	1.136015 a 1.503169
	5	4.734171	24.66255	0.7965987	5.835545	34.05359	3.551779 a 5.916562
	6	6.086325	15.2825	0.32851	4.866613	23.68392	5.100258 a 7.072392
	7	9.524771	36.35722	0.325485	12.64467	159.8876	7.002837 a 12.046706
	8	1.606732	25.76283	0.413085	3.828445	14.65699	0.8391775 a 2.3742867
	9	1.13184	5.483498	0.4131763	0.8692603	0.7556135	0.9584691 a 1.3052109
Jitter (ms)	Teste	Média	Max	Min	STD	Var	IC
Rede 3	1	6.49196	19.0808	1.312535	3.305829	10.9285	5.832624 a 7.151296
	2	3.356705	17.6544	1.459083	3.013282	9.079866	2.755717 a 3.957693
	3	1.979172	5.758566	1.083236	0.8530358	0.7276701	1.809037 a 2.149307
	4	2.53796	5.106705	1.498284	0.6921511	0.4790731	2.399913 a 2.676007
	5	7.788057	51.38466	0.04217875	13.06837	170.7823	5.14016 a 10.43595
	6	3.451421	9.648678	0.671825	2.361138	5.574971	2.973010 a 3.929832
	7	2.252492	16.99816	0.3144663	3.333576	11.11273	1.587622 a 2.917362
	8	0.6593675	2.934016	0.3960475	0.3188766	0.1016823	0.5954368 a 0.7232982
	9	0.6876682	2.251651	0.2952845	0.3681598	0.1355417	0.6142400 a 0.7610964

O critério RSSI, embora com algumas medidas bruscas de pico, em virtude de questões inerentes ao próprio meio de transmissão, sujeito portanto a interferências, apresenta variação média entre 71% a 100% para o dispositivo móvel e entre 85% e 100% para o site parceiro, considerando a escala de medição de 0 a 70 da fabricante Atheros. Os valores absolutos são apresentados nas Tabelas 5.9 e 5.10.

Por se tratar de uma comparação não enviesada, ou seja, cada critério tem a mesma importância ou peso, o decisor considera automaticamente a rede que apresenta os menores valores de *delay*, *jitter* e perda de pacotes, e os maiores valores de RSSI. Ao término de cada teste (99 iterações), tem-se o percentual de vezes que cada rede foi selecionada.

A Fig. 5.2 apresenta de forma consolidada a variação média para o critério *delay*. Para melhor visualização gráfica, os valores acima de 40ms foram fixados em 50ms. Os

**Tabela 5.6.:** Valores do critério Jitter: Móvel.

Jitter (ms)	Teste	Média	Max	Min	STD	Var	IC
Rede 1	1	0.6208773	2.553048	0.10865	0.6684848	0.4468719	0.4875504 a 0.7542043
	2	0.6159984	3.095226	0.1196425	0.7716795	0.5954893	0.4620897 a 0.7699072
	3	0.6836027	2.683954	0.1294888	0.7752186	0.6009639	0.5289881 a 0.8382173
	4	0.8188496	3.506016	0.1452637	0.8391857	0.7042326	0.6506035 a 0.9870957
	5	0.9656002	5.118094	0.2245362	1.204752	1.451428	0.7253166 a 1.2058838
	6	0.5547705	3.675424	0.1450325	0.6882966	0.4737522	0.4174922 a 0.6920489
	7	1.096371	2.651564	0.2029637	0.8702205	0.7572838	0.9228088 a 1.2699336
	8	0.7604564	4.131362	0.1869787	0.9102095	0.8284814	0.5789183 a 0.9419945
	9	1.058877	3.836584	0.2904687	1.05338	1.10961	0.848784 a 1.268970
Jitter (ms)	Teste	Média	Max	Min	STD	Var	IC
Rede 2	1	2.772078	14.27095	0.671535	1.929091	3.72139	2.387327 a 3.156828
	2	1.619116	29.53916	0.4533019	2.889426	8.348785	1.042830 a 2.195402
	3	0.6997719	1.888925	0.3609638	0.240528	0.05785372	0.6517994 a 0.7477444
	4	1.285054	27.29965	0.5147002	2.735688	7.483991	0.736583 a 1.833525
	5	11.69753	88.4305	0.6617662	24.72644	611.397	6.76593 a 16.62913
	6	52.81825	155.3129	0.3418062	60.99102	3719.905	40.65380 a 64.98269
	7	28.34792	140.311	0.1635119	51.02566	2603.618	18.17103 a 38.52481
	8	0.5650142	1.83154	0.291848	0.2792114	0.07795899	0.5093265 a 0.6207019
	9	1.984628	68.20129	0.21715	9.55005	91.20345	0.07990462 a 3.88935232
Jitter (ms)	Teste	Média	Max	Min	STD	Var	IC
Rede 3	1	4.71609	12.25568	0.3928525	3.278293	10.74721	4.062246 a 5.369934
	2	2.10786	8.649898	0.3151725	1.624254	2.638202	1.783908 a 2.431812
	3	0.6365258	2.841632	0.2213775	0.4836299	0.2338979	0.5400675 a 0.7329840
	4	0.8298942	2.413346	0.2831024	0.3712148	0.1378004	0.7554704 a 0.9043181
	5	9.190453	55.05336	0.4198075	13.43577	180.52	6.510736 a 11.870171
	6	0.9889798	1.95036	0.1119875	0.4692536	0.2201989	0.8953888 a 1.0825707
	7	0.3561032	1.112525	0.09879638	0.3129948	0.09796575	0.2936775 a 0.4185289
	8	0.2939589	1.297462	0.13422	0.1782479	0.03177232	0.2584079 a 0.3295098
	9	0.2346904	1.073778	0.08349737	0.179374	0.03217503	0.1989149 a 0.2704659

valores absolutos podem ser consultados nas Tabelas 5.3 e 5.4. Observe que até o teste 7 (tráfego de 5 Mbit/s), a Rede 1 apresenta as menores médias. Esse bom desempenho em relação ao parâmetro *delay*, bem como para os demais, faz com que a mesma seja indicada como a melhor, tanto no SP quanto no móvel, em boa parte dos testes realizados, conforme evidenciado nas Tabelas 5.1 e 5.2.

A variação média do critério *jitter* é ilustrada na Fig. 5.3. Os valores maiores que 12ms foram fixados em 15ms. Entretanto, os valores absolutos estão disponíveis nas Tabelas 5.5 e 5.6. Nota-se que a Rede 1 apresenta as menores médias de atraso ao longo dos testes. Entre os testes 5 e 7 tem-se um aumento considerável do *jitter* para a Rede 2, chegando a ser superior à Rede 3, que encontrava-se em produção.

**Tabela 5.7.:** Valores do critério Perda de Pacotes: SP.

Perda	Teste	Média	Max	Min	STD	Var	IC
Rede 1	1	0.07070707	1	0	0.2576394	0.06637807	0.0193218 a 0.1220923
	2	0.02020202	1	0	0.1414068	0.01999588	-0.008001065 a 0.048405105
	3	0.09090909	1	0	0.2889428	0.08348794	0.03328047 a 0.14853772
	4	0.1313131	3	0	0.4200696	0.1764585	0.04753173 a 0.21509453
	5	0.3645833	3	0	0.7963045	0.6341009	0.2032370 a 0.5259296
	6	0.04166667	1	0	0.2008753	0.04035088	0.0009655525 a 0.0823677808
	7	0.04040404	1	0	0.197907	0.03916718	0.0009321845 a 0.0798758963
	8	0.02040816	1	0	0.1421189	0.02019777	-0.008084865 a 0.048901192
	9	0.06060606	2	0	0.2791453	0.07792208	0.004931519 a 0.116280602
Perda	Teste	Média	Max	Min	STD	Var	IC
Rede 2	1	0.06060606	1	0	0.2398206	0.05751391	0.01277469 a 0.10843743
	2	0.05050505	1	0	0.2200991	0.04844362	0.006607056 a 0.094403045
	3	0.01010101	1	0	0.1005038	0.01010101	-0.009944116 a 0.030146136
	4	0.01010101	1	0	0.1005038	0.01010101	-0.009944116 a 0.030146136
	5	0.09375	3	0	0.4371709	0.1911184	0.005170937 a 0.182329063
	6	0.03125	1	0	0.174906	0.03059211	-0.004189247 a 0.066689247
	7	0.1818182	3	0	0.5778856	0.3339518	0.06656093 a 0.29707543
	8	0.04081633	1	0	0.1988818	0.03955397	0.0009430502 a 0.0806896029
	9	1.919192	5	0	1.706391	2.911771	1.578858 a 2.259526
Perda	Teste	Média	Max	Min	STD	Var	IC
Rede 3	1	0.8888889	3	0	0.9022522	0.814059	0.7089379 a 1.0688399
	2	0.2222222	3	0	0.707908	0.5011338	0.08103245 a 0.36341199
	3	0.02020202	1	0	0.1414068	0.01999588	-0.008001065 a 0.048405105
	4	0.03030303	1	0	0.1722922	0.0296846	-0.004060043 a 0.064666103
	5	0.5104167	5	0	0.833443	0.6946272	0.3415454 a 0.6792879
	6	0.1041667	1	0	0.3070802	0.09429825	0.04194644 a 0.16638690
	7	0.02020202	1	0	0.1414068	0.01999588	-0.008001065 a 0.048405105
	8	0.01020408	1	0	0.1010153	0.01020408	-0.01004820 a 0.03045636
	9	0	0	0	0	0	NaN

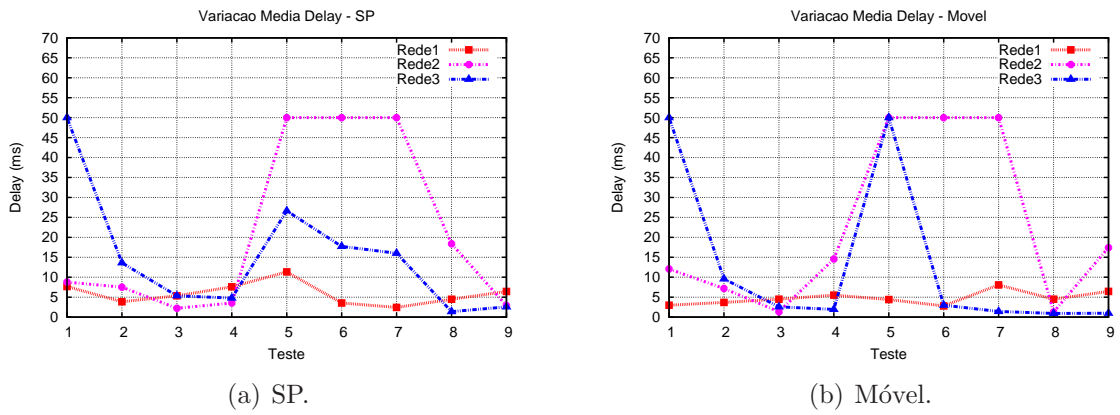


Figura 5.2.: Variação média *Delay*.

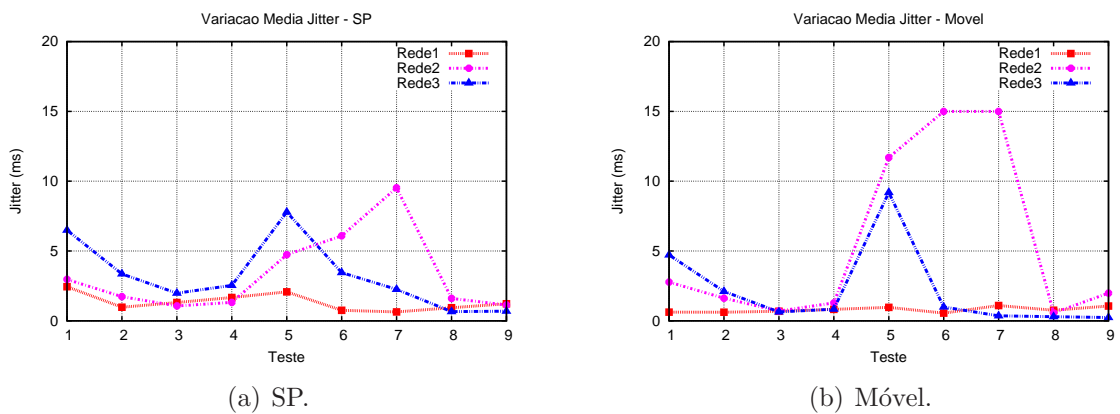


Figura 5.3.: Variação média *Jitter*.

Para o critério perda de pacotes considerou-se o número máximo de pacotes perdidos por teste, diferentemente da média. O intuito é verificar sob as condições de intenso tráfego para as redes de avaliação (Rede 1 e Rede 2) o percentual máximo de perda de pacotes. A Fig. 5.3 ilustra essa variação. Convém salientar que o percentual de perda de pacotes foi calculado baseado no número de pacotes ICMP utilizados a cada iteração durante a execução de um determinado teste. Este valor corresponde a 20 pacotes.

Conforme dito anteriormente, embora haja variações bruscas do nível de sinal percebido pelas entidades, a variação média do critério do RSSI manteve-se uniforme. Na Fig. 5.5 é apresentada essa variação.

**Tabela 5.8.:** Valores do critério Perda de Pacotes: Móvel.

Perda	Teste	Média	Max	Min	STD	Var	IC
Rede 1	1	0.01010101	1	0	0.1005038	0.01010101	-0.009944116 a 0.030146136
	2	0.02020202	1	0	0.1414068	0.01999588	-0.008001065 a 0.048405105
	3	0.07070707	1	0	0.2576394	0.06637807	0.0193218 a 0.1220923
	4	0.1122449	1	0	0.3172905	0.1006733	0.04863218 a 0.17585762
	5	0.05050505	2	0	0.2623962	0.06885178	-0.001828955 a 0.102839056
	6	0.01010101	1	0	0.1005038	0.01010101	-0.009944116 a 0.030146136
	7	0.2222222	1	0	0.4178554	0.1746032	0.1388824 a 0.3055620
	8	0.1111111	2	0	0.3466719	0.1201814	0.04196862 a 0.18025360
	9	0.2020202	2	0	0.4280902	0.1832612	0.1166391 a 0.2874013
Perda	Teste	Média	Max	Min	STD	Var	IC
Rede 2	1	0	0	0	0	0	NaN
	2	0.04040404	1	0	0.197907	0.03916718	0.0009321845 a 0.0798758963
	3	0.01010101	1	0	0.1005038	0.01010101	-0.009944116 a 0.030146136
	4	0	0	0	0	0	NaN
	5	0.2020202	2	0	0.6057513	0.3669347	0.08120523 a 0.32283517
	6	1.424242	4	0	1.629391	2.654917	1.099266 a 1.749219
	7	0.7171717	4	0	1.317391	1.735518	0.4544228 a 0.9799207
	8	0	0	0	0	0	NaN
	9	0.05050505	3	0	0.3606609	0.1300763	-0.02142749 a 0.12243760
Perda	Teste	Média	Max	Min	STD	Var	IC
Rede 3	1	0.3535354	2	0	0.5404671	0.2921047	0.2457411 a 0.4613296
	2	0.09090909	3	0	0.5168766	0.2671614	-0.01218013 a 0.19399831
	3	0	0	0	0	0	NaN
	4	0	0	0	0	0	NaN
	5	2.151515	11	0	3.283737	10.78293	1.496585 a 2.806445
	6	0	0	0	0	0	NaN
	7	0.07070707	1	0	0.2576394	0.06637807	0.0193218 a 0.1220923
	8	0	0	0	0	0	NaN
	9	0	0	0	0	0	NaN

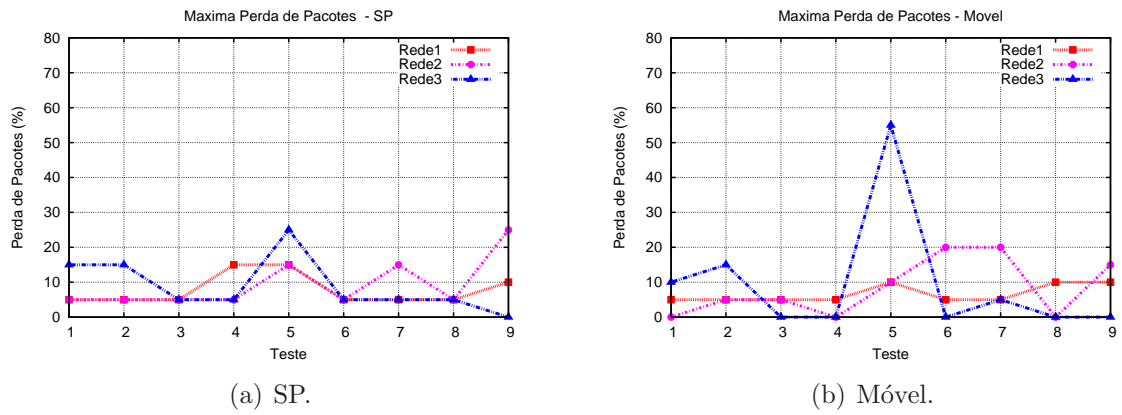


Figura 5.4.: Máxima perda de pacotes.

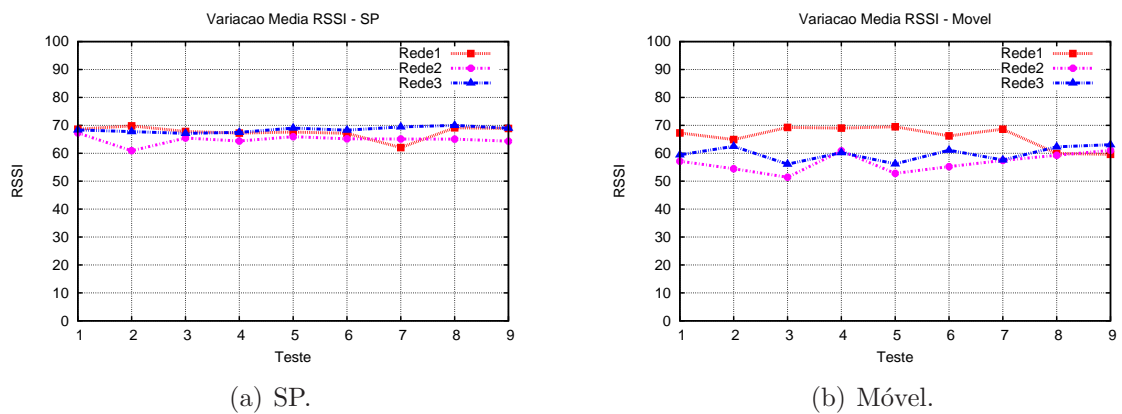


Figura 5.5.: Variação média RSSI.

De maneira a verificar se o desempenho das redes e as indicações de seleção pelo método de tomada decisão, tanto no móvel quanto no SP, se mantinham dentro dos intervalos de confiança obtidos, realizou-se 50 repetições para cada um dos testes, seguindo a mesma metodologia apresentada no Capítulo 4. Cada teste totalizou um tempo aproximado de 35 horas. Em seguida, aplicou-se o teste de comparação *t-test* e toda análise estatística aos dados.

**Tabela 5.9.:** Valores do critério RSSI: SP.

RSSI	Teste	Média	Max	Min	STD	Var	IC
Rede 1	1	68.62626	70	36	4.636998	21.50175	67.70143 a 69.55110
	2	69.79798	70	60	1.414068	1.999588	69.51595 a 70.08001
	3	67.81818	70	22	5.382063	28.9666	66.74475 a 68.89162
	4	67.23232	70	22	8.515087	72.5067	65.53402 a 68.93063
	5	67.59375	70	27	8.162273	66.6227	65.93992 a 69.24758
	6	67.15625	70	23	10.2365	104.7859	65.08214 a 69.23036
	7	62.0404	70	25	15.97506	255.2024	58.85424 a 65.22657
	8	69.13265	70	24	6.060924	36.7348	67.91751 a 70.34779
	9	68.87879	70	24	6.568746	43.14842	67.56867 a 70.18890
RSSI	Teste	Média	Max	Min	STD	Var	IC
Rede 2	1	67.27273	70	59	3.012959	9.077922	66.67180 a 67.87365
	2	60.89899	70	36	5.275076	27.82643	59.84689 a 61.95109
	3	65.47475	70	51	3.450358	11.90497	64.78659 a 66.16291
	4	64.38384	70	57	3.145152	9.891981	63.75655 a 65.01113
	5	65.92708	70	56	2.534996	6.426206	65.41345 a 66.44072
	6	65.16667	70	37	4.216925	17.78246	64.31224 a 66.02110
	7	65.09091	69	54	2.914044	8.491651	64.50971 a 65.67210
	8	65.08163	69	56	2.469081	6.09636	64.58661 a 65.57665
	9	64.32323	69	38	3.184235	10.13935	63.68815 a 64.95832
RSSI	Teste	Média	Max	Min	STD	Var	IC
Rede 3	1	68.36364	70	25	7.763671	60.27458	66.81520 a 69.91207
	2	67.74747	70	25	9.817044	96.37436	65.78950 a 69.70545
	3	67.11111	70	22	11.02584	121.5692	64.91205 a 69.31018
	4	67.49495	70	22	10.31778	106.4566	65.43710 a 69.55279
	5	69.02083	70	26	5.123689	26.25219	67.98268 a 70.05899
	6	68.27083	70	27	8.342004	69.58904	66.58059 a 69.96108
	7	69.43434	70	32	3.949068	15.59514	68.64672 a 70.22197
	8	70	70	70	0	0	NaN
	9	68.9798	70	42	4.152648	17.24449	68.15157 a 69.80803

Para todas as repetições, o percentual de escolha de cada uma das redes consideradas manteve-se nos intervalos de confiança apresentados nas Tabelas 5.1 e 5.2. Em relação aos valores dos critérios para Rede 3 (não controlada), a variação se deu em virtude do aumento e diminuição dos usuários conectados durante a execução do teste, ou seja, ao longo das 35 horas.

**Tabela 5.10.:** Valores do critério RSSI: Móvel.

RSSI	Teste	Média	Max	Min	STD	Var	IC
Rede 1	1	67.28283	70	39	6.077796	36.9396	66.07063 a 68.49502
	2	64.91919	70	32	9.749596	95.05463	62.97467 a 66.86371
	3	69.25253	70	27	4.459211	19.88456	68.36315 a 70.14190
	4	69.0102	70	29	4.88949	23.90711	68.02992 a 69.99048
	5	69.50505	70	66	0.9832655	0.966811	69.30894 a 69.70116
	6	66.23232	70	28	9.798822	96.0169	64.27798 a 68.18666
	7	68.61616	70	40	4.771743	22.76953	67.66445 a 69.56787
	8	59.92929	70	34	13.3388	177.9235	57.26892 a 62.58967
	9	59.61616	70	28	11.80514	139.3614	57.26167 a 61.97066
RSSI	Teste	Média	Max	Min	STD	Var	IC
Rede 2	1	57.18182	67	36	7.305804	53.37477	55.72470 a 58.63894
	2	54.46465	65	34	7.098883	50.39415	53.04880 a 55.88049
	3	51.40404	67	33	6.233417	38.85549	50.16081 a 52.64727
	4	60.84694	69	42	6.961384	48.46087	59.45127 a 62.24261
	5	52.79798	64	32	8.238758	67.87714	51.15479 a 54.44117
	6	55.22222	68	41	7.931538	62.9093	53.64030 a 56.80414
	7	57.43434	66	35	6.463653	41.77881	56.14519 a 58.72350
	8	59.27273	67	39	4.713892	22.22078	58.33256 a 60.21290
	9	61.0404	68	44	5.464018	29.85549	59.95062 a 62.13018
RSSI	Teste	Média	Max	Min	STD	Var	IC
Rede 3	1	59.44444	70	31	6.729803	45.29025	58.10221 a 60.78668
	2	62.49495	70	43	7.174748	51.47702	61.06397 a 63.92593
	3	56.09091	69	43	4.851027	23.53247	55.12339 a 57.05843
	4	60.23469	70	47	6.314443	39.87219	58.96873 a 61.50066
	5	56.19192	68	39	4.420908	19.54442	55.31018 a 57.07365
	6	61.09091	70	52	5.316687	28.26716	60.03051 a 62.15130
	7	57.60606	68	40	5.350602	28.62894	56.53890 a 58.67322
	8	62.29293	70	35	6.820217	46.51536	60.93266 a 63.65320
	9	63.08081	70	38	5.927585	35.13626	61.89857 a 64.26304



### 5.3. Análise para o cenário de testes com mobilidade

Os testes realizados para o cenário com mobilidade envolveu a utilização de uma segunda interface de rede pelo dispositivo móvel. O propósito, conforme descrito no Capítulo 4, consiste em verificar a eficiência da solução proposta pela arquitetura, no que tange ao tempo de *handover* percebido pelo dispositivo móvel, utilizando o sistema operacional android.

Desta forma, realizou-se três testes com quantidades diferentes de requisições HTTP para o SP, onde a aplicação móvel realizou o consumo do *webservice*, recebendo a indicação da rede e, por fim, realizando o *handover*. Esse procedimento é ilustrado na Fig. 4.2. Convém salientar que, durante a realização dos testes, todas as aplicações do dispositivo móvel que fazem consumos de dados na internet foram desativadas.

Os resultados para esse conjunto de testes são detalhados na Tabela 5.11, onde são apresentados o tempo médio, os valores máximo e mínimo, o desvio padrão, a variância, e por fim o intervalo de confiança para a média, ao nível de significância de 95%.

**Tabela 5.11.:** Tempo de *Handover* (em segundos).

Teste	Requisições	Média	Max	Min	STD	VAR	IC
1	50	8.565872	12.68971	6.028629	1.231818	1.517375	8.215793 a 8.915951
2	100	8.852	15.67986	6.023044	1.283332	1.646942	8.597359 a 9.106641
3	200	8.86363	18.68728	6.027285	1.40192	1.965381	8.668149 a 9.059112

Observe que o tempo médio de *handover* é de aproximadamente 9s. Esse valor pode ser considerado bastante representativo, visto que as requisições ocorrem utilizando um enlace de uma operadora de celular e consulta ao IP público do SP, isto é, utiliza infraestrutura de internet na consulta ao *webservice*, e não de uma rede local. A partir desse resultado obtido, o usuário móvel poderia optar em diminuir o tempo de solicitação de rede ao SP, que está automaticamente configurado para 30s.

O tempo de *handover* por requisição, para cada um dos testes realizados, é apresentado nas Figs. 5.6, 5.7 e 5.8. A presença de *outliers* em todos os testes realizados indicam as diversas condições de tráfego no núcleo da rede da operadora de celular (rede 3G) utilizada durante a execução dos testes. Por esse motivo, as premissas de normalidades testadas usando Shapiro-Wilk foram violadas.

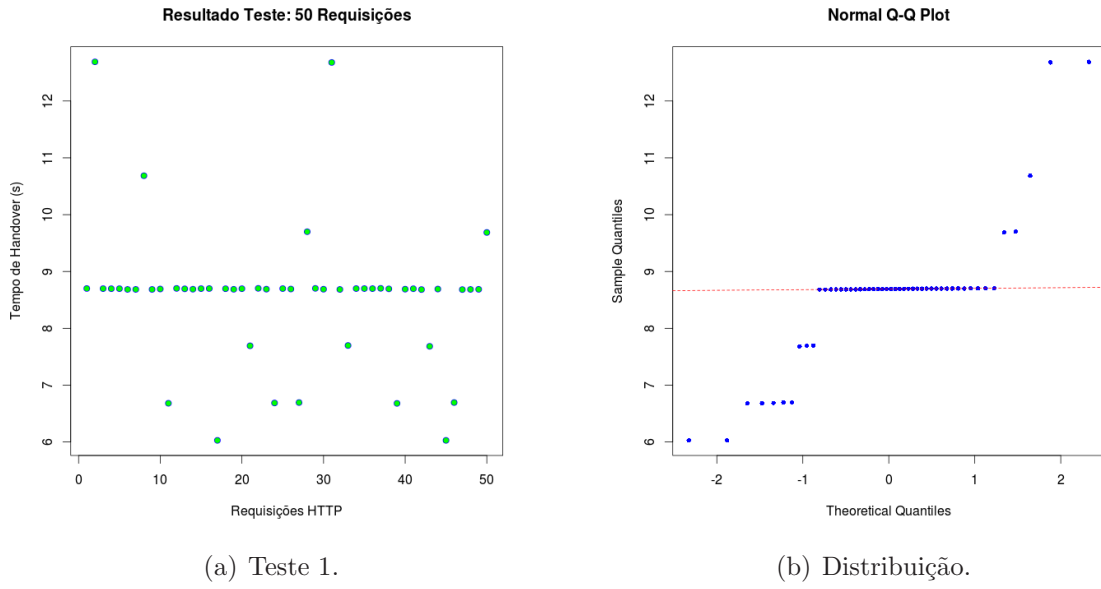


Figura 5.6.: Resultado teste com 50 requisições.

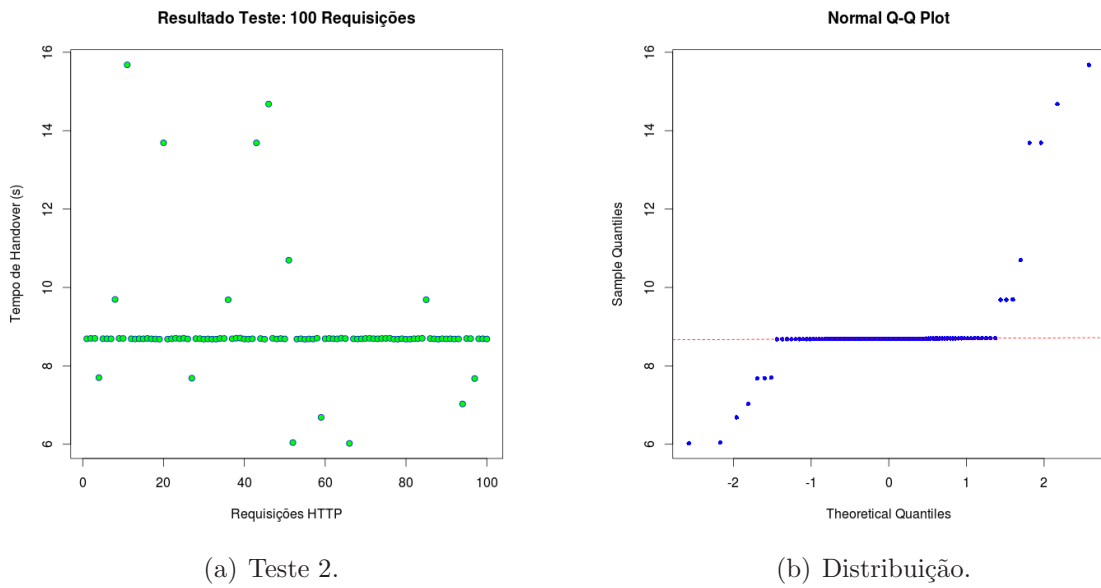


Figura 5.7.: Resultado teste com 100 requisições.

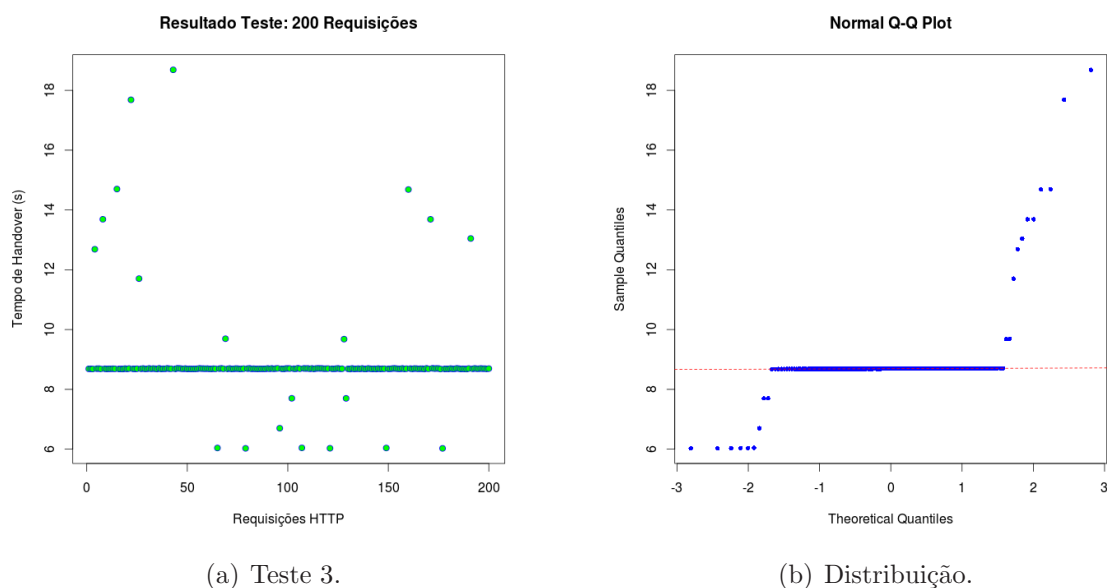


Figura 5.8.: Resultado teste com 200 requisições.

## 5.4. Conclusão

Neste capítulo apresentou-se os resultados e análise para os cenários de teste definidos no capítulo 4. Observa-se a partir dos resultados obtidos e das análises estatísticas aferidas que a solução proposta pela arquitetura pode ser considerada eficiente para os cenários avaliados.

Para o cenário sem mobilidade, observou-se ainda que não existe diferença significativa entre o que é observado pelo dispositivo móvel e pelo servidor. Essa característica reforça o viés de se transferir a carga de processamento, i.e., análise dos parâmetros e a tomada de decisão, no momento da seleção para uma entidade da rede. Desta forma, o dispositivo móvel apenas “consumiria” essa informação, verificando somente se as redes indicadas são alcançadas por ele, evitando realizar medições locais e consumo energético desnecessário.

Para o cenário com mobilidade, os valores numéricos obtidos evidenciam a eficiência da solução no que se refere ao tempo total de *handover*. Nota-se que neste cenário (pior caso), o usuário necessitou utilizar a sua conexão 3G (tarifada) para receber as informações sobre as melhores redes disponíveis em uma determinada zona *wifi*. O núcleo da operadora de telefonia celular funciona como uma “caixa-preta” para arquitetura proposta.

# Capítulo 6.

## Conclusões

Este trabalho concentrou esforços em propor uma solução aplicável para o problema de seleção de redes, envolvendo cenários de redes convergentes. A arquitetura descrita supõe que o processo de seleção deve ser distribuído entre as entidades participantes (SP e móvel), devendo ser realizado sem consumo incremental de energia para o móvel e independente da operadora. O propósito final compreendeu possibilitar ao usuário a melhor experiência possível na utilização dos seus serviços.

Buscou-se desta forma estudar os modelos de seleção disponíveis na literatura, verificando as técnicas que se propõem a resolver o problema. Inicialmente foi necessário um estudo abrangente, de maneira a subsidiar a escolha e a opção pelo método que iria compor o núcleo da arquitetura. Assim, optou-se pelo método Promethee II, pela sua simplicidade e eficiência quando comparado a outros métodos. Na definição dos pesos e prioridades entre os critérios, fez-se uso da escala fundamental do AHP e seus mecanismos de julgamentos.

Em seguida assumiu-se algumas premissas, tais como: a solução deveria ser passível de implantação em servidores com diversas distribuições linux; a implantação deveria ocorrer de forma transparente em redes que já estivessem em produção; ser modularizada e escalável, propondo facilidades para integração de novos módulos; ser independente de fabricantes e modelos específicos de *hardware*; além de utilizar preferencialmente softwares *open source*.

Após a definição das entidades que compõem a arquitetura e do desenvolvimento do modelo de aplicabilidade da solução, partiu-se para a validação da proposta propriamente. Para tal, foram criados dois cenários de testes, visando observar a resiliência (tempo de resposta) da arquitetura sob certas condições. Além disto, buscou-se verificar o

percentual de escolha do móvel e do site parceiro de modo a aferir se compensaria trazer de fato a decisão sobre a seleção de rede para a própria rede.

Os questionamentos levantados no capítulo 1 foram respondidos neste trabalho da seguinte forma:

- Questão *a)* O processo de classificação e seleção das redes utilizou métodos de tomada de decisão multiatributo, protocolos de rede e recursos dos sistemas operacionais das entidades envolvidas (SP e móvel) ;
- Questão *b)* O módulo de processamento da arquitetura assume as preferências (importância dos critérios) definidas pelo operador que implantou a solução SP, ou configuradas pelo usuário quando o processamento é realizado localmente na aplicação móvel;
- Questão *c)* Os critérios foram obtidos em tempo real e independente de tecnologia de acesso. Faz-se necessário apenas que o SP esteja conectado à rede a ser monitorada;
- Questão *d)* O dispositivo móvel realiza o *handover* a partir da indicação feita pelo SP;
- Questão *e)* Embora o tempo de handover obtido na validação da arquitetura seja pequeno, faz-se necessário um protocolo de gerência de mobilidade na camada 3, para que o usuário não tenha sua conexão interrompida em nenhum momento. Essa questão encontra-se fora do escopo deste trabalho.

Convém ressaltar que os requisitos (R1 ... R7) definidos para arquitetura (vide capítulo 3) foram todos atendidos. Desta forma, as contribuições esperadas por esta dissertação foram alcançadas, principalmente no que tange ao modelo de aplicabilidade real da solução. A aplicação móvel pode ser reescrita para outros sistemas operacionais, tais como *iOS* e *Windows Phone*, visto que somente é necessário conhecimento sobre a gerência das *interfaces* de redes desses sistemas. A interoperabilidade é garantida, pois utiliza-se somente requisições HTTP para consumo do *webservice*. O consumo de recursos computacionais da aplicação móvel proposta é bem inferior a aplicações populares como WhatsApp, Facebook, AVG, Skype, dentre outros.

Por fim, verificou-se a partir da análise dos resultados e considerando as restrições dos cenários, que a solução proposta mostrou-se eficiente. O tempo médio de 9s obtido para os cenários envolvendo mobilidade ainda é alto, embora tolerável. Dessa forma, faz-se necessário a realização de novos testes, considerando outros cenários, além da

necessidade de propor melhorias nos algoritmos desenvolvidos, bem como a avaliação de outras técnicas e métodos para o núcleo da arquitetura proposta, de maneira a torná-la uma solução/produto final.

A abordagem *WLAN-First* utilizada pelas operadoras é pragmática, no entanto propostas de modelos e arquiteturas que propõem soluções mais eficientes têm sido amplamente difundidas na literatura, o que deve nortear em um futuro próximo melhoramentos por parte das operadoras de telecomunicações.

## 6.1. Trabalhos Futuros

Como trabalhos futuros pretende-se refinar a solução proposta, melhorando aspectos visuais da aplicação móvel. Além disso, pretende-se realizar a publicação da *app* em um repositório para *download* e utilização pelos usuários finais. Inicialmente para um grupo de teste, tal que os usuários reportem os *bugs* da aplicação e sugestões de melhorias de usabilidade. Visa-se ainda portar a aplicação para dispositivos como *Smart TV* e relógios inteligentes que possuam o sistema operacional android.

Para o Site Parceiro, converter o conjunto de *softwares* e *daemons* desenvolvidos em pacotes de serviços, utilizando modelos de configuração baseados em serviços tradicionais de rede, como DHCP e DNS. Desta forma, possibilitando a disponibilização de tais pacotes em repositórios de diversas distribuições linux. Nesse intuito, propor um modelo de padronização ao IETF (*Internet Engineering Task Force*) através de uma *Internet Draft*, visando obter uma RFC (*Request for Comments*).

Note que, como a arquitetura comporta a aplicação de outras estratégias de seleção em seu núcleo, visa-se realizar novos testes utilizando Redes Neurais ou Algoritmos Genéticos, ou ainda a implementação de uma função utilidade baseada na predição dos valores de QoS das redes avaliadas. Neste caso, a função seria embasada a partir de modelos estatísticos ou modelos não-lineares para sistemas dinâmicos e variantes no tempo.

Objetiva-se ainda melhorar o esquema de colaboração entre os SPs, para troca de informações sobre a qualidade das redes. O intuito é que um conjunto de SPs possa, mediante consultas entre si, fornecer uma cobertura maior de seleção para uma determinada região.

Outro ponto refere-se a melhorias na forma de requisição da melhor rede por parte do dispositivo móvel. Assim, além de solicitar a melhor ou o *ranking* com as melhores redes disponíveis, o móvel passaria a informar também o tipo de perfil de consumo do usuário, i.e., serviços como VoIP, VoD e dados. Esses perfis estariam pré-configurados no SP, sendo que as redes seriam classificadas segundo esses padrões.

Desta forma o SP faria um atendimento personalizado por perfil do usuário. Na versão atual, conforme discutido no capítulo 3, o módulo de processamento assume as preferências definidas pelo operador que implantou a solução SP, ou configuradas pelo usuário quando o processamento é realizado localmente na aplicação móvel.

Por último, utilizar a arquitetura proposta sobre um protocolo de mobilidade. Isso já foi feito, e a aplicação funcionou corretamente para aplicações que fazem uso do protocolo TCP. No entanto, é necessário a realização de um conjunto de testes para avaliar o comportamento da mesma para cenários envolvendo aplicações com requisitos de tempo real (UDP), como *streaming* de vídeo e áudio. Caso os trabalhos apontados sejam satisfatórios, testes envolvendo a parte cliente da arquitetura em VANTs e veículos autônomos passam a ser factíveis.

# Referências

- Alkhwilani, M. & Ayyesh, A. (2008). Access network selection based on fuzzy logic and genetic algorithms, *Advances in Artificial Intelligence* **8**(1): 1–12.
- Bari, F. & Leung, V. C. M. (2007). Automated network selection in a heterogeneous wireless network environment, *Network, IEEE* **21**(1): 34–40.
- Bi, T., Trestian, R. & Muntean, G.-M. (2013). RLoad: Reputation-based load-balancing network selection strategy for heterogeneous wireless environments, *21st IEEE International Conference on Network Protocols (ICNP)* pp. 1–3.
- Bischoff, E. (2008). *Estudo da utilização de algoritmos genéticos para seleção de redes de acesso*, Master's thesis, Universidade de Brasília.
- Braun, R. & Abolhasan, M. (2011). A Comprehensive Survey on RAT Selection Algorithms for Heterogeneous Networks, *International Conference of Communications Networking and Mobile Computing* **5**(1): 104–108.
- Chamodrakas, I. & Martakos, D. (2012). A utility-based fuzzy TOPSIS method for energy efficient network selection in heterogeneous wireless networks, *Applied Soft Computing* **12**(7): 1929–1938.
- Charilas, D. E. & Panagopoulos, A. D. (2010). Multiaccess Radio Network Environments, *IEEE Vehicular Technology Magazine* (December): 40–49.
- Chen, L. (2010). A Distributed Access Point Selection Algorithm Based on No-Regret Learning for Wireless Access Networks, *2010 IEEE 71st Vehicular Technology Conference* pp. 1–5.
- Choi, Y., Ji, H. W., Park, J. Y., Kim, H. C. & Silvester, J. a. (2011). A 3W network strategy for mobile data traffic offloading, *IEEE Communications Magazine* **49**(October): 118–123.
- Ciccarese, G., Blasi, M. D., Marra, P., Mighali, V., Palazzo, C., Patrono, L. & Stefanizzi, M. L. (2009). Vertical handover algorithm for heterogeneous wireless networks, *Fifth International Joint Conference on INC, IMS and IDC, IEEE*, pp. 1948–1954.
- Costa, H. G. (2004). *Introdução ao método de análise hierárquica (análise multicritério no auxílio à decisão)*, XXXVI - Simpósio Brasileiro de Pesquisa Operacional (SBPO).
- Costa, N. (2009). Dynamic Adaptation of Quality of Service for VoIP Communications,



- International Journal on Advances in Networks and Services - ACM* **2**(2): 155–166.
- Dreibholz, T. & Pulinthanath, J. (2014). Applicability of reliable server pooling for SCTP-based endpoint mobility, *Network Working Group - Internet Draft, IETF*.
- Edwin Winston, P., Kuzhaloli, S. et al. (2014). A new adaptive method to optimize SCTP during vertical handover in UMTS/WLAN integrated networks, *Communications and Signal Processing (ICCSP), 2014 International Conference on*, IEEE, pp. 1890–1894.
- Elmasri, R. & Navathe, S. (2005). *Sistemas de banco de dados*, Pearson Addison Wesley.
- Figueira, J., Greco, S. & Ehrgott, M. (2005). *Multiple Criteria Decision Analysis: State of The Art Surveys*, Vol. 78 of *International Series in Operations Research & Management Science*, Springer Science.
- Filho, M. C. S. (2011). *Arquitetura orientada a serviços para o comércio eletrônico no sistema brasileiro de tv digital*, Master's thesis, Universidade de Brasília.
- Freeman, E. (2007). *USE A CABEÇA! PADRÕES DE PROJETO: (DESIGN PATTERS)*, ALTA BOOKS.
- Han, B., Hui, P., Kumar, V. S. A., Marathe, M. V., Member, S., Shao, J. & Srinivasan, A. (2012). Mobile Data Offloading through Opportunistic Communications and Social Participation, *IEEE Transactions on Mobile Computing* **11**(5): 821–834.
- Hsiao, Y.-K. & Lin, Y.-W. (2014). A mobility management scheme for internet of things, *Mobile, Ubiquitous, and Intelligent Computing*, Springer, pp. 569–575.
- Hsieh, R., Zhou, Z. & Seneviratne, a. (2003). S-MIP: a seamless handoff architecture for mobile IP, *IEEE INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Cat. No.03CH37428)* **3**(C): 1774–1784.
- Ibrahim, M., Khawam, K. & Tohme, S. (2009). Network-Centric Joint Radio Resource Policy in Heterogeneous WiMAX-UMTS Networks for Streaming and Elastic traffic, *2009 IEEE Wireless Communications and Networking Conference* pp. 1–6.
- Junior, F. R. L. (2013). *Comparação entre os métodos fuzzy TOPSIS e fuzzy AHP no apoio à tomada de decisão para seleção de fornecedores*, Master's thesis, USP.
- Karim, R., Ding, C. & Chi, C. H. (2011). An enhanced PROMETHEE model for QoS-based web service selection, *Proceedings - 2011 IEEE International Conference on Services Computing, SCC 2011*, pp. 536–543.
- Kassar, M., Kervella, B. & Pujolle, G. (2008). An overview of vertical handover decision strategies in heterogeneous wireless networks, *Computer Communications* **31**(10): 2607–2620.
- Kejuruteraan, K., Ampangan, S., Selatan, S. P. & Pinang, P. (2008). Optimizing

- Network Selection to Support End-User QoS Requirements for Next Generation Networks, *International Journal of Computer Science and Network Security* **8**(6): 113–117.
- Khan, M. A., Toseef, U., Marx, S. & Goerg, C. (2010). Game-Theory Based User Centric Network Selection with Media Independent Handover Services and Flow Management, *2010 8th Annual Communication Networks and Services Research Conference* pp. 248–255.
- Kumaran, U. & Shaji, R. (2014). Vertical handover in vehicular ad-hoc network using multiple parameters, *Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 2014 International Conference on*, IEEE, pp. 1059–1064.
- Kurose, J. & Ross, K. (2013). *Redes de computadores e a internet: uma abordagem top-down*, Pearson.
- Lahby, M., Cherkaoui, L. & Adib, A. (2013). Hybrid network selection strategy by using M-AHP/E-TOPSIS for heterogeneous networks, *2013 8th International Conference on Intelligent Systems: Theories and Applications (SITA)* pp. 1–6.
- Lassoued, I., Bonnin, J.-M., Hamouda, Z. B. & Belghith, a. (2008). A Methodology for Evaluating Vertical Handoff Decision Mechanisms, *Seventh International Conference on Networking (icn 2008)* pp. 377–384.
- Lee, S., Sriram, K., Kim, K., Kim, Y. H. & Golmie, N. (2009). Vertical handoff decision algorithms for providing optimized performance in heterogeneous wireless networks, *Vehicular Technology, IEEE Transactions on* **58**(2): 865–881.
- Marques, C. P. C. (2013). *Identificação de ofensores via análise da sensibilidade de estações na vazão de redes ieee 802.11*, Master's thesis, Universidade Católica de Campinas.
- Masri, M. A. A., Sesay, A. B. & Fapojuwo, A. O. (2014). Session state aware handover procedure for VoIP sessions in heterogeneous wireless networks, *Wireless Communications and Networking Conference (WCNC), 2014 IEEE*, IEEE, pp. 3011–3016.
- Mendes Valenzuela, G., Cunha, R. F. et al. (2011). *Mecanismo de seleção de rede em ambientes heterogêneos baseado em qualidade de experiência (QoE)*, Master's thesis, Universidade Federal de Pernambuco.
- Mohamed, L., Leghris, C. & Abdellah, A. (2012). A survey and comparison study on weighting algorithms for access network selection, *Wireless On-demand Network Systems and Services (WONS), 2012 9th Annual Conference on*, IEEE, pp. 35–38.
- Monteiro, C. C. (2012). *Um Ambiente para Apoio à Integração de Redes Sem Fio Heterogêneas*, PhD thesis, Universidade de Brasília.
- Montgomery, D. C. & Runger, G. C. (2011). *Applied Statistics and Probability for Engineers*, fifth edn, Wiley.

- Na, T. & Kim, M. (2014). A novel no-reference PSNR estimation method with regard to deblocking filtering effect in H. 264/AVC bitstreams, *Circuits and Systems for Video Technology, IEEE Transactions on* **24**(2): 320–330.
- Neves, L., Bastos, V. & Almeida, A. T. D. (2002). Utilização do método Promethee II na análise das propostas de preços em um processo de licitação, *XXII Encontro Nacional de Engenharia de Produção* pp. 1–8.
- Nguyen-Vuong, Q. (2008). On utility models for access network selection in wireless heterogeneous networks, *NOMS 2008*. pp. 144–151.
- Niyato, D. & Hossain, E. (2009). Dynamics of network selection in heterogeneous wireless networks: an evolutionary game approach, *Vehicular Technology, IEEE Transactions on* **58**(4): 2008–2017.
- Omheni, N., Zarai, F., Obaidat, M., Hsiao, K.-F. & Kamoun, L. (2014). A novel media independent handover-based approach for vertical handover over heterogeneous wireless networks, *International Journal of Communication Systems* **27**(5): 811–824.
- Parreiras, R. O. (2006). *Algoritmos Evolucionários e Técnicas de Tomada de Decisão em Análise Multicritério*, PhD thesis, Universidade Federal de Minas Gerais.
- Perkins, C. E. (1998). Mobile networking through mobile IP, *IEEE Internet Computing* **2**: 58–69.
- Pradesh, A., Pradesh, A. & Pradesh, A. (2014). A PROMETHEE Approach for Network Selection in Heterogeneous Wireless Environment, *Advances in Computing, Communications and Informatics (ICACCI)* pp. 2560–2564.
- Quental, N. C. & Gonçalves, P. A. d. S. (2013). Uma estratégia de tentativas de handover vertical em grupo, *XXXI Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC 2013)*.
- Radhika, K. & Reddy, A. (2011). Network selection in heterogeneous wireless networks based on fuzzy multiple criteria decision making, *Electronics Computer Technology* pp. 136–139.
- Ramirez, R. & Ramos, R. (2009). A vertical handoff decision algorithm which considers the uncertainty during the decision making process, *2009 IFIP International Conference on Wireless and Optical Communications Networks*, IEEE.
- Ricardo, C. A. (2009). *Otimização do processo de decisão no handover vertical em redes baseadas no subsistema multimídia IP (IMS)*, Master's thesis, PUC-PR.
- Rios, V. M. (2012). *Seleção de redes baseada em técnicas de apoio à decisão*, Master's thesis, Universidade de Brasília.
- Rios, V. M., Gondim, P. R. L. & Monteiro, C. C. (2012). Use of fuzzy logic for networks selection in heterogeneous wireless environment, *14th International Conference on Advanced Communication Technology (ICACT)*, IEEE, pp. 798–803.

- Rodrigues Jr, O. & Dias, K. L. (2006). *Arquitetura para integração de redes sem fio heterogêneas 3GPP/WLAN*, Master's thesis, Universidade Federal do Pará.
- Shalloway, A. & Trott, J. (2004). *Explicando Padrões de Projetos*, BOOKMAN COMPANHIA ED.
- Shen, W. & Zeng, Q. (2008). Cost-function-based network selection strategy in integrated wireless and mobile networks, *Vehicular Technology, IEEE Transactions* **57**(6): 3778–3788.
- Silva, D., Monteiro, C. & Batista, L. (2014). Classificação e seleção de redes via tomada de decisão multicritério, *VI Congreso Internacional de Computación y Telecomunicaciones (COMTEL), Lima*. pp. 258–264.
- Silva, P. H. D. & Junior, N. A. (2014). Ferramenta IPERF: geração e medição de Tráfego TCP e UDP, *Centro Brasileiro de Pesquisas Físicas (CBPF) - Notas Técnicas*. pp. 1–13.
- Silva, R. S. & Cancian, R. L. (2008). Análise de Desempenho da Biblioteca Libpcap: Uma Abordagem Voltada a Gerência de Segurança, *VIII Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais (SBSeg'08)* pp. 357–367.
- Silva, W. B., Almeida, A. & Fonseca, K. V. O. (2014). Um modelo de referência completa para avaliação objetiva da qualidade de vídeo em dispositivos móveis em ambientes do sistema brasileiro de TV digital (SBTVD), *SET-Revista de Radiodifusão* **3**(03).
- Siris, V. a. & Anagnostopoulou, M. (2013). Performance and Energy Efficiency of Mobile Data Offloading with Mobility Prediction and Prefetching, *IEEE 14th International Symposium and Workshops on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. .
- Sommerville, I. (2011). *Engenharia de software*, PEARSON EDUCATION.
- Song, Q. S. Q. & Jamalipour, a. (2005). A network selection mechanism for next generation networks, *IEEE International Conference on Communications, 2005. ICC 2005*. **2**(C): 1418–1422.
- Suresh, N., Palaniappan, R., Mane, P. & Jayant, N. (2009). Testing of a no-reference VQ metric: Monitoring quality and detecting visible artifacts, *Proceedings of the Fourth International Workshop on Video Processing and Quality Metrics for Consumer Electronics*.
- Suresh, N., Yang, O. & Jayant, N. (2007). AVQ: A zero-reference metric for automatic measurement of the quality of visual communications, *Invited Talk, Third International Workshop on Video Processing and Quality Metrics for Consumer Electronics, Scottsdale, AZ, USA*.
- Tanenbaum, A. (2003). *Redes de computadores*, CAMPUS - RJ.
- Thiagarajah, S. P., Ting, A., Chieng, D., Alias, M. Y. & Wei, T. S. (2013). User Data

- Rate Enhancement Using Heterogeneous LTE-802 . 11n Offloading in Urban Area, pp. 11–16.
- Ting, A., Chieng, D., Kwong, K. H., Communication, W., Andonovic, I., Wong, K. D. & Alto, P. (2013). Dynamic Backhaul Sensitive Network Selection Scheme in LTE-WiFi Wireless HetNet, *IEEE 24th International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)* pp. 3061–3065.
- Trestian, R., Ormond, O. & Muntean, G.-M. (2012). Game Theory-Based Network Selection: Solutions and Challenges, *IEEE Communications Surveys & Tutorials* **14**(4): 1212–1231.
- Trestian, R., Ormond, O. & Muntean, G.-M. (2014). Enhanced Power-Friendly Access Network Selection Strategy for Multimedia Delivery Over Heterogeneous Wireless Networks, *IEEE Transactions on Broadcasting* **60**(1): 85–101.
- Valenzuela, G., Ferreira, I., Nóbrega, O. & Cunha, P. (2012). Vertical handover decision based on quality of experience in heterogeneous wireless networks, *Proceedings of the 6th Euro American Conference on Telematics and Information Systems*, ACM, pp. 67–72.
- Wang, L. & Binet, D. (2009). Mobility-based network selection scheme in heterogeneous wireless networks, *Conference, 2009. VTC Spring 2009. IEEE* pp. 1–5.
- Xiang, Z. & Kong, Q. (2012). A mobility management architecture for IPv4/IPv6 networks, *Proceedings of the 2012 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, CyberC 2012* pp. 443–446.
- Zhang, W. (2004). Handover decision using fuzzy MADM in heterogeneous networks, *Wireless Networking Conference (WCNC)* **2**.

# Apêndice A.

## Detalhamento dos Métodos MADM

A seguir é apresentado um exemplo didático da execução do Algoritmo 3.3, no que tange a utilização do método AHP, de forma conjunta com a aplicação do Promethee II para o problema de seleção de redes.

### A.1. Aplicação do Método AHP

Conforme comentado no módulo de processamento, detalhado no capítulo 3, as funções *readMatrix* e *calculateAhpWeights* são responsáveis pelo mapeamento das preferências do usuário (operador do sistema) que implantará a solução. Os questionamentos realizados ao usuário permitem identificar a preferência que cada critério terá em relação ao outro.

Desta forma, considerou-se a escala fundamental de Saaty (Figueira et al. 2005), (Costa 2004). A Tabela A.1 apresenta a relação entre a intensidade da importância (preferência) dos critérios e a definição verbal do mesmo, i.e, a relação entre a escala numérica e a verbal. Os valores 2, 4, 6 e 8, embora não evidenciados, atribuem relações de preferência intermediárias na escala de importância.

Por questões didáticas, assume-se que a matriz com a comparação par a par entre os critérios adotados à luz do foco principal, i.e., seleção de rede, é fornecida pela Tabela A.2.

**Tabela A.1.:** Escala Fundamental de Saaty.

Intensidade da importância	Definição
1	Igual preferência
3	Preferência moderada
5	Preferência forte
7	Preferência muito forte
9	Preferência absoluta

**Tabela A.2.:** Comparação par a par entre os critérios.

Seleção de Rede	Delay	Jitter	Perda	RSSI
Delay	1	2	1/2	3
Jitter	1/2	1	1/2	3
Perda	2	2	1	7
RSSI	1/3	1/3	1/7	1

De posse das preferências dos critérios, é necessário a normalização dos dados. Desta forma são realizados dois procedimentos, sendo eles: i) calcular o somatório de cada elemento das colunas dos critérios; ii) dividir cada elemento pelo somatório da coluna (Costa 2004). Essas operações são apresentadas nas Tabelas A.3 e A.4.

Em seguida, obtém-se as Prioridades Médias Locais (PML) (Costa 2004). As PMLs correspondem as médias das linhas da matriz de julgamento dos critérios. Essa operação é ilustrada na Tabela A.5.

**Tabela A.3.:** Somatório de cada elemento das colunas dos critérios.

Seleção de Rede	Delay	Jitter	Perda	RSSI
Delay	1	2	1/2	3
Jitter	1/2	1	1/2	3
Perda	2	2	1	7
RSSI	1/3	1/3	1/7	1
—	—	—	—	—
Somatório	$1+(1/2)+2+(1/3) = 3.83$	$2+1+2+(1/3) = 5.33$	$(1/2)+(1/2)+1+(1/7) = 2.14$	$3+3+7+1 = 14$

**Tabela A.4.:** Divisão de cada elemento pelo somatório da coluna.

Seleção de Rede	Delay	Jitter	Perda	RSSI
Delay	$1/3.83 = 0.26$	$2/5.33 = 0.38$	$(1/2)/2.14 = 0.23$	$3/14 = 0.21$
Jitter	$(1/2)/3.83 = 0.13$	$1/5.33 = 0.19$	$(1/2)/2.14 = 0.23$	$3/14 = 0.21$
Perda	$2/3.83 = 0.52$	$2/5.33 = 0.38$	$1/2.14 = 0.47$	$7/14 = 0.5$
RSSI	$(1/3)/3.83 = 0.087$	$(1/3)/5.33 = 0.063$	$(1/7)/2.14 = 0.067$	$1/14 = 0.071$

**Tabela A.5.:** Geração das PML.

Seleção de Rede	Delay	Jitter	Perda	RSSI	PML
Delay	0.26	0.38	0.23	0.21	$(0.26+0.38+0.23+0.21)/4 = \mathbf{0.27}$
Jitter	0.13	0.19	0.23	0.21	$(0.13+0.19+0.23+0.21)/4 = \mathbf{0.19}$
Perda	0.52	0.38	0.47	0.5	$(0.52+0.38+0.47+0.5)/4 = \mathbf{0.47}$
RSSI	0.087	0.063	0.067	0.071	$(0.087+0.063+0.067+0.071)/4 = \mathbf{0.07}$

O vetor de prioridades (pesos), obtido após a utilização do método AHP, para os critérios *delay*, *jitter*, perda de pacotes e RSSI é dado respectivamente pelos valores 0.27, 0.19, 0.47 e 0.07. Conforme pode ser visualizado na Tabela A.5, as PMLs refletem a importância que cada critério terá para o problema de seleção de redes. Note que para esta configuração, a importância maior é dada ao critério perda de pacotes, seguido do *delay*, *jitter* e, por fim, o nível de sinal. Este último é tratado como o critério de menor importância.

Esses pesos são passados como parâmetros na função *applyPromethee* do Algoritmo 3.3.

## A.2. Aplicação do Método Promethee II

O método Promethee II considerou quatro redes como alternativas, conforme organização apresentada na Tabela A.6. Neste exemplo, os valores dos critérios para cada uma das alternativas foram definidos de forma arbitrária. Em seguida, definiu-se o tipo de curva a ser utilizada na comparação entre as alternativas, para cada um dos critérios considerados (Figueira et al. 2005).



**Tabela A.6.:** Matriz de Critérios e Alternativas - Promethee II.

Alternativas	Delay	Jitter	Perda	RSSI
Rede 1	100.85	89.10	20	54
Rede 2	85.65	34.65	18	62
Rede 3	92.40	66.20	24	31
Rede 4	76.80	48.30	17	50

**Tabela A.7.:** Características dos critérios para o decisor.

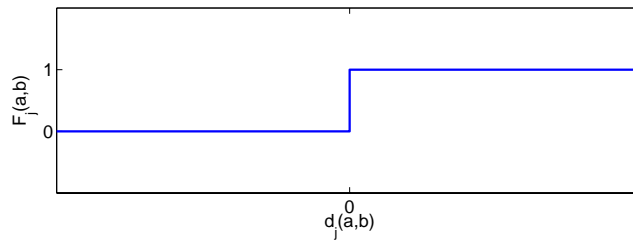
Critérios	Peso	Curva	Parâmetros
Delay	0.27	I	—
Jitter	0.19	I	—
Perda de Pacotes	0.47	I	—
RSSI	0.07	I e V	$p = 60\%$ e $q = 45\%$

A Tabela A.7 apresenta essas informações de forma consolidada. É importante notar que as curvas das funções, também conhecidas como funções de preferência, seguem a equação (A.1) para a curva I (critério usual). Para tal curva, cada alternativa é comparada com a outra, não dependendo de parâmetros adicionais. A alternativa que apresentar o melhor valor para um determinado critério é a vencedora à luz deste critério. Esse processo é denominado imediata preferência estrita.

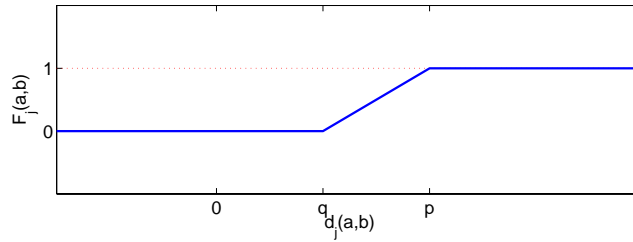
Para a curva V (critério de indiferença), que usa a equação (A.2), são definidos limites de indiferença ( $q$ ) e preferência ( $p$ ). A preferência entre as alternativas é crescente para essa função (Figueira et al. 2005). A Fig. A.1 apresenta as curvas descritas nas equações (A.1) e (A.2).

$$\begin{aligned}
 d_j(a, b) &\leq 0, F_j(a, b) = 0 \\
 d_j(a, b) &> 0, F_j(a, b) = 1
 \end{aligned}
 \tag{A.1}$$

$$\begin{aligned}
 d_j(a, b) &\leq q, F_j(a, b) = 0 \\
 q < d_j(a, b) &\leq p, F_j(a, b) = \frac{d_j - q}{p - q} \\
 d_j(a, b) &> p, F_j(a, b) = 1
 \end{aligned}
 \tag{A.2}$$



(a) Curva I: Critério Usual.



(b) Curva V: Critério de Indiferença.

**Figura A.1.:** Funções de Preferências.

Nas equações (A.1) e (A.2),  $d_j(a, b)$  refere-se à matriz com as alternativas, onde a alternativa  $\mathbf{a}$  é representada na vertical e  $\mathbf{b}$  na horizontal. As comparações entre as alternativas ocorrem em função de um determinado critério, conforme ilustrado na Tabela A.8.  $F_j(a, b)$  representa a função de preferência, a qual assume valores entre 0 e 1, indicando a preferência entre as alternativas. Para a equação (A.2) tem-se ainda os limites de indiferença ( $q$ ) e preferência ( $p$ ).

O grau de sobreclassificação  $\pi(a, b)$ , calculado na comparação entre as alternativas  $\mathbf{a}$  e  $\mathbf{b}$ , é dado pelo conjunto de equações (A.3), em que  $w_j$  representa o peso do critério  $j$ ,  $F_j(a, b)$  a função de preferência,  $g_j$  refere-se ao valor do critério considerando a curva e o peso utilizados,  $\mathbf{A}$  a matriz de comparação das alternativas em relação a um determinado critério, e  $\mathbf{a}$  e  $\mathbf{b}$ , conforme dito anteriormente, são duas alternativas arbitrárias. Esse esquema é detalhado a seguir, conforme as matrizes de comparação para cada um dos critérios (vide Tabelas A.8, A.9, A.10 e A.11).

$$\pi(a, b) = \frac{1}{W} \sum_{j=1}^n w_j \cdot F_j(a, b), \text{ onde } W = \sum_{j=1}^n w_j$$

$$d_j : A \times A \rightarrow R$$

$$d_j(a, b) = g_j(a) - g_j(b), \text{ onde } a, b \in A.$$

$$F_j(a, b) = F_j[d_j(a, b)]$$
(A.3)

A definição das matrizes de comparação por critério segue a formatação dada pela Tabela A.6. É importante salientar que para os critérios *delay*, *jitter* e perda de pacotes, quanto menores forem os valores obtidos melhor é a rede em relação ao parâmetro em questão. A partir dessa premissa, faz-se necessário inverter as equações (A.1), de maneira que as funções de preferência reflitam essa premissa. A alteração é demonstrada na equação (A.4).

$$d_j(a, b) \geq 0, F_j(a, b) = 0$$

$$d_j(a, b) < 0, F_j(a, b) = 1$$
(A.4)

**Tabela A.8.:** Matriz de comparação para o critério delay.

Delay	Rede 1	Rede 2	Rede 3	Rede 4
Rede 1	—	0	0	0
Rede 2	1	—	1	0
Rede 3	1	0	—	0
Rede 4	1	1	1	—

Para o critério RSSI, quanto maiores forem os valores obtidos melhor é a rede em relação a esse critério. O Algoritmo 3.3, através da função *applyPromethee*, permite a definição de limiares de preferência e indiferença, conforme equação (A.2). A escolha da curva tipo V para este critério se deu em virtude das medições sofrerem variações de picos. Essas variações são inerentes ao próprio meio de acesso (o ar), sujeito a interferências, degradação do sinal, dentre outros.

**Tabela A.9.:** Matriz de comparação para o critério jitter.

Jitter	Rede 1	Rede 2	Rede 3	Rede 4
Rede 1	—	0	0	0
Rede 2	1	—	1	1
Rede 3	1	0	—	0
Rede 4	1	0	1	—

**Tabela A.10.:** Matriz de comparação para o critério perda de pacotes.

Perda	Rede 1	Rede 2	Rede 3	Rede 4
Rede 1	—	0	1	0
Rede 2	1	—	1	0
Rede 3	0	0	—	0
Rede 4	1	1	1	—

**Tabela A.11.:** Matriz de comparação para o critério RSSI.

RSSI	Rede 1	Rede 2	Rede 3	Rede 4
Rede 1	—	0	1	1
Rede 2	1	—	1	1
Rede 3	0	0	—	0
Rede 4	0	0	1	—

Após a definição das matrizes de comparação de cada critério, aplica-se os pesos fornecidos pelo AHP, conforme descrição na Tabela A.7. Essas operações podem ser visualizadas nas Tabelas A.12, A.13, A.14 e A.15.

**Tabela A.12.:** Matriz de comparação delay considerando o peso do critério.

Delay	Rede 1	Rede 2	Rede 3	Rede 4
Rede 1	—	$0 \times 0.27 = 0$	$0 \times 0.27 = 0$	$0 \times 0.27 = 0$
Rede 2	$1 \times 0.27 = 0.27$	—	$1 \times 0.27 = 0.27$	$0 \times 0.27 = 0$
Rede 3	$1 \times 0.27 = 0.27$	$0 \times 0.27 = 0$	—	$0 \times 0.27 = 0$
Rede 4	$1 \times 0.27 = 0.27$	$1 \times 0.27 = 0.27$	$1 \times 0.27 = 0.27$	—

**Tabela A.13.:** Matriz de comparação jitter considerando o peso do critério.

Jitter	Rede 1	Rede 2	Rede 3	Rede 4
Rede 1	—	$0 \times 0.19 = 0$	$0 \times 0.19 = 0$	$0 \times 0.19 = 0$
Rede 2	$1 \times 0.19 = 0.19$	—	$1 \times 0.19 = 0.19$	$1 \times 0.19 = 0.19$
Rede 3	$1 \times 0.19 = 0.19$	$0 \times 0.19 = 0$	—	$0 \times 0.19 = 0$
Rede 4	$1 \times 0.19 = 0.19$	$0 \times 0.19 = 0$	$1 \times 0.19 = 0.19$	—

**Tabela A.14.:** Matriz de comparação perda de pacotes considerando o peso do critério.

Perda	Rede 1	Rede 2	Rede 3	Rede 4
Rede 1	—	$0 \times 0.47 = 0$	$1 \times 0.47 = 0.47$	$0 \times 0.47 = 0$
Rede 2	$1 \times 0.47 = 0.47$	—	$1 \times 0.47 = 0.47$	$0 \times 0.47 = 0$
Rede 3	$0 \times 0.47 = 0$	$0 \times 0.47 = 0$	—	$0 \times 0.47 = 0$
Rede 4	$1 \times 0.47 = 0.47$	$1 \times 0.47 = 0.47$	$1 \times 0.47 = 0.47$	—

**Tabela A.15.:** Matriz de comparação RSSI considerando o peso do critério.

RSSI	Rede 1	Rede 2	Rede 3	Rede 4
Rede 1	—	$0 \times 0.07 = 0$	$1 \times 0.07 = 0.07$	$1 \times 0.07 = 0.07$
Rede 2	$1 \times 0.07 = 0.07$	—	$1 \times 0.07 = 0.07$	$1 \times 0.07 = 0.07$
Rede 3	$0 \times 0.07 = 0$	$0 \times 0.07 = 0$	—	$0 \times 0.07 = 0$
Rede 4	$0 \times 0.07 = 0$	$0 \times 0.07 = 0$	$1 \times 0.07 = 0.07$	—

Em seguida, calculam-se os chamados índices de preferência positiva ( $\phi^+$ ) e negativa ( $\phi^-$ ), conforme a equação (A.5). O primeiro corresponde à preferência da alternativa **a** sobre todas as alternativas, enquanto o índice de preferência negativa corresponde ao índice de preferência de todas as alternativas sobre **a**.

$$\begin{aligned} \phi^+(a) &= \sum_{b \in A} \pi(a, b) \\ \phi^-(a) &= \sum_{b \in A} \pi(b, a) \end{aligned} \tag{A.5}$$

As matrizes de preferência positiva e negativa são apresentadas na Tabela A.16.

**Tabela A.16.:** Matrizes de preferência positiva e negativa.

$\pi$	Rede 1	Rede 2	Rede 3	Rede 4
Rede 1	—	0	0.54	0.07
Rede 2	1	—	1	0.26
Rede 3	0.46	0	—	0
Rede 4	0.93	0.74	1	—

Por fim, é calculado a matriz resultante da aplicação do método, isto é, o índice final de preferência, dado pela equação (A.6). O somatório das linhas da Tabela A.16 corresponde ao índice de preferência positivo ( $\phi^+$ ), e o somatório das colunas o índice de preferência negativo ( $\phi^-$ ).

$$\phi(a) = \phi^+(a) - \phi^-(a), a \in A.$$

$$\phi(a) > \phi(b), \text{ent\~{a}o a alternativa } \mathbf{a} \text{ \textit{\'e} prefer\~{i}vel \textit{\'a} alternativa } \mathbf{b}. \tag{A.6}$$

$$\phi(a) = \phi(b), \text{ent\~{a}o a alternativa } \mathbf{a} \text{ \textit{\'e} indiferente \textit{\'a} alternativa } \mathbf{b}.$$

O *ranking* de preferência entre as alternativas ( $\phi$ ), saída do método Promethee II, é apresentado na Tabela A.17.

**Tabela A.17.:** Matriz resultante Promethee II.

Alternativas	$\phi^+$	$\phi^-$	$\phi$
Rede 1	0.61	2.39	-1.78
Rede 2	2.26	0.74	1.52
Rede 3	0.46	2.54	-2.08
Rede 4	2.67	0.33	2.34

A ordem de classificação para este exemplo é portanto: **Rede 4** > **Rede 2** > **Rede 1** > **Rede 3**. Logo, a **Rede 4** é a rede selecionada como a melhor dentre as avaliadas.

Tanto as informações sobre a melhor rede quanto do *ranking* são persistidas no SGBD da arquitetura proposta, finalizando o processo.