

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM
CIÊNCIA DA COMPUTAÇÃO

**Investigando a Capacidade de Comunicação e o Consumo
de Bateria de PDAs em Redes Bluetooth e Wi-Fi**

ALUNO: Roberto Rigolin Ferreira Lopes
ORIENTADOR: Prof. Dr. Hélio Crestana Guardia

SÃO CARLOS
2007

**Investigando a Capacidade de Comunicação e o Consumo
de Bateria de PDAs em Redes Bluetooth e Wi-Fi**

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM
CIÊNCIA DA COMPUTAÇÃO

**Investigando a Capacidade de Comunicação e o Consumo
de Bateria de PDAs em Redes Bluetooth e Wi-Fi**

Roberto Rigolin Ferreira Lopes

**Dissertação apresentada ao Programa
de Pós-Graduação em Ciência da
Computação da Universidade Federal
de São Carlos, como parte dos requisitos
para obtenção do Título de Mestre em
Ciência da Computação.
Orientador: Dr. Hélio Crestana Guardia**

SÃO CARLOS
2007

**Ficha catalográfica elaborada pelo DePT da
Biblioteca Comunitária da UFSCar**

L864ic

Lopes, Roberto Rigolin Ferreira.

Investigando a capacidade de comunicação e o consumo de bateria de PDAs em redes Bluetooth e Wi-Fi / Roberto Rigolin Ferreira Lopes. -- São Carlos : UFSCar, 2009.
80 f.

Dissertação (Mestrado) -- Universidade Federal de São Carlos, 2007.

1. Rede de computadores. 2. Computação móvel. 3. Sistemas de comunicação sem fio. I. Título.

CDD: 004.68 (20^a)

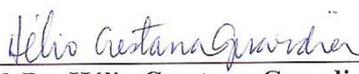
Universidade Federal de São Carlos
Centro de Ciências Exatas e de Tecnologia
Programa de Pós-Graduação em Ciência da Computação

***“Investigando a Capacidade de Comunicação e
Consumo de Bateria de PDAs em Redes Bluetooth e
Wi-Fi”***

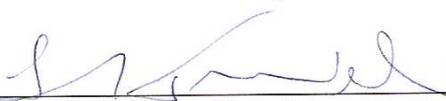
ROBERTO RIGOLIN FERREIRA LOPES

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação.

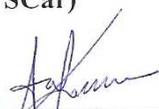
Membros da Banca:



Prof. Dr. Hélio Crestana Guardia
(Orientador – DC/UFSCar)



Prof. Dr. Luis Carlos Trevelin
(DC/UFSCar)



Prof. Dr. Antonio Alfredo Ferreira Loureiro
(DCC/UFMG)

São Carlos
Maio/2007

Dedico este trabalho à minha mãe, Marcela Madalena Rigolin, à minha tia, Lúcia Elvira Rigolin de Almeida e, principalmente, ao meu tio Hudson Rigolin que nos deixou com saudades cedo demais.

Agradecimentos

A minha mãe (Marcela), minha irmã (Milene) e a minha tia (Lúcia). Obrigado pelo incentivo e me desculpem os longos intervalos de tempo sem vê-las;

Ao Prof. Hélio Crestana Guardia pelos impagáveis ensinamentos que variaram desde o âmbito técnico até lições de vida. Muito obrigado professor pela sua amizade e confiança;

Ao Prof. Gilson Alberto Rosa Lima pelo incentivo, amizade e conselhos.

Ao Prof. Peter Zeilhofer pelo apoio, sem o qual as coisas teriam sido bem difíceis e/ou inviáveis.

A Caroline Beatriz Perlin pela amizade, companheirismo e inspirações;

Ao Pedro Northon Nobile pelas diversas madrugadas de discussões intermináveis. Meu amigo, você tornou as coisas bem mais fáceis.

Aos colegas: Leandro A. Villas, Robson de Grande, Danilo Moschetto, Luiz Falaguasta, Prof. Leonardo Barreto, Daniele Santini, William Borges, Luciano de Paula, Guilherme Torres e Luanna Lobato. Nos vemos pelo mundo.

Aos professores: Luis Carlos Trevelin, Wanderley Lopes de Souza, Sérgio Donizetti Zorzo e Célio Moron, que de alguma forma influenciaram minha formação.

“Questions of science, science and progress, do not speak as loud as my heart.”
(The Scientist – Chris Martin *et al.*)

Resumo

Neste trabalho realizamos uma investigação do comportamento de PDAs (*Personal Digital Assistants*) em redes sem fio, com foco na capacidade de comunicação e no consumo de bateria. Partimos da hipótese que as características do *hardware* de um PDA influenciam a capacidade de comunicação e o consumo de bateria do mesmo e, portanto, são relevantes na seleção de acesso. A hipótese é pertinente, uma vez que a seleção automática de acesso é um dos desafios das futuras redes de acesso sem fio. Estas redes almejam a interoperabilidade entre as diversas tecnologias de acesso, para permitir que seus usuários permaneçam sempre conectados ao melhor ambiente de rede disponível (*ABC - Always Best Connected*). Com objetivo de verificar esta hipótese foi desenvolvida uma aplicação chamada *PDA Stress*, que, por meio de experimentos, realiza a avaliação quantitativa da capacidade de comunicação e do consumo de bateria de PDAs. Foram experimentados dois PDAs, com capacidades de processamento distintas, em três redes: Bluetooth, Wi-Fi modo infra-estruturado e Wi-Fi modo *ad hoc*. Os resultados obtidos neste trabalho ratificam a hipótese supracitada. Como a implementação do conceito ABC ainda é tema de pesquisa, acreditamos que, hoje, a escolha do dispositivo móvel e da tecnologia de comunicação deve ser feita na fase de projeto da aplicação. Para auxiliar tal escolha, foi desenvolvida uma aplicação chamada *PDA Chooser*, que é um SSD (Sistema de Suporte à Decisão) que suporta a escolha de um PDA e da tecnologia de comunicação para um determinado cenário. A base de conhecimento do SSD foi construída a partir dos resultados obtidos nos estudos de caso.

Abstract

This work presents an investigation about the behavior of PDAs (Personal Digital Assistants) in wireless networks, focusing on the communication capability and battery consumption. Our hypothesis consists that hardware features of a PDA influence the communication capability and battery consumption; therefore, they are relevant in the access selection. This hypothesis is pertinent because the automatic access selection is still a challenge in the next generation of wireless networks. These networks aim the interoperability among the various technologies of access to allow users to be connected always using the best possible network environment (ABC - Always Best Connected). With the objective of verifying the hypothesis, we have developed an application called PDA Stress which, through experiments, makes the quantitative evaluation of the communication capability and battery consumption of PDAs. In our experiments, we have used two PDAs with different processing capabilities in three different networks: Bluetooth, infra-structured Wi-Fi and ad hoc Wi-Fi. The results of this work confirm the presented hypothesis. As the ABC concept is still a subject of research, we believe that the choice of the mobile device and the communication technology should be done during the project phase of an application. To help this choice, we have developed an application called PDA Chooser, which is a Decision Support System (DSS) that supports selecting an appropriate PDA and communication technology to an specific scenario. The knowledge basis of the DSS was created based on the results we collected in the case studies.

Sumário

Lista de Figuras.....	III
Lista de Tabelas	V
Lista de Acrônimos	VI
1 Introdução.....	1
2 Conceitos Básicos	5
2.1 Computação Móvel	6
2.2 Próxima Geração de Redes Sem Fio	9
2.3 Tecnologias de Acesso Sem Fio.....	12
2.3.1 IEEE 802.15.1 - Bluetooth	13
2.3.2 IEEE 802.11 – Wi-Fi.....	16
2.3.3 Cenários Típicos	19
2.4 Considerações Finais	21
3 Investigando o Comportamento de PDAs.....	22
3.1 Investigando o Consumo de Bateria.....	23
3.1.1 Utilizando a API do Sistema Operacional.....	24
3.1.2 Monitorando a Potência Dissipada.....	29
3.1.3 Comparação Entre os Dois Métodos	33
3.2 Investigando a Capacidade de Comunicação	35
3.3 Ambiente Experimental.....	37
3.4 Ambientes de Simulação	40
3.5 Considerações Finais	42
4 Resultados.....	44
4.1 Rede Bluetooth	45
4.2 Rede Wi-Fi Infra-estruturada.....	49

4.3	Rede Wi-Fi <i>Ad hoc</i>	53
4.4	Comparação do Desempenho dos Dois PDAs	57
4.5	Resultados da Simulação	61
4.6	Considerações Finais	66
5	Sistema de Suporte à Decisão	67
5.1	Modelagem do SSD.....	68
5.2	PDA Chooser	71
5.3	Considerações Finais	73
6	Conclusões.....	74
6.1	Trabalhos Futuros	76
	Referências Bibliográficas	78
	Anexos	A-1
	Implementação dos Experimentos.....	A-1
	PDA Stress Mobile	A-1
	PDA Stress Server	A-4

Lista de Figuras

Figura 1. Exemplo de mobilidade: (a) Sala de reunião e (b) Escritório.	7
Figura 2. Exemplo de mobilidade: (c) Em trânsito e (d) Na empresa do cliente.	8
Figura 3. Arquitetura funcional de um ambiente ABC.	10
Figura 4. Duas piconets formando uma scatternet.	14
Figura 5. Pilha de protocolos Bluetooth.	15
Figura 6. Topologias básicas do padrão 802.11.	18
Figura 7. Conexão de duas BSS formando uma ESS.	18
Figura 8. Cenários típicos.	20
Figura 9 . Percentual de bateria em função do tempo.	26
Figura 10. Percentual de bateria em função do volume de dados.	28
Figura 11. Comportamento do PDA nas redes Bluetooth e Wi-Fi.	29
Figura 12. Hardware que monitora o consumo de bateria.	31
Figura 13. Comportamento do PDA nas redes Bluetooth e Wi-Fi.	32
Figura 14. Comparação do comportamento do PDA nas duas abordagens.	34
Figura 15. Eficiência do PDA com os dois métodos nas duas redes.	35
Figura 16. Ambiente de rede onde a aplicação <i>PDA Stress</i> é empregada.	36
Figura 17. Ambiente experimental: (a) Redes ad hoc e (b) Rede infra-estruturada.	38
Figura 18. Foto dos dois PDA s: Zire 72s (à esquerda) e LifeDrive (à direita).	39
Figura 19. Pontos de medição no ambiente experimental e no ambiente de simulação.	41
Figura 20. Esboço do ambiente no qual foram realizados os experimentos.	44
Figura 21. Taxas de transmissão do Zire 72s na rede Bluetooth.	46
Figura 22. Consumo de bateria do Zire na rede Bluetooth.	47
Figura 23. Taxas de transmissão do LifeDrive na rede Bluetooth.	48

Figura 24. Consumo de bateria do LifeDrive na rede Bluetooth.....	49
Figura 25. Taxa de transmissão do Zire 72s na rede Wi-Fi infra-estruturada.	50
Figura 26. Consumo de bateria do Zire na rede Wi-Fi infra-estruturada.	51
Figura 27. Taxa de transmissão do LifeDrive na rede Wi-Fi infra-estruturada.	52
Figura 28. Consumo de bateria do LifeDrive na rede Wi-Fi infra-estruturada.	53
Figura 29. Taxa de transmissão do Zire 72s na rede Wi-Fi ad hoc.	54
Figura 30. Consumo de bateria do Zire na rede Wi-Fi <i>ad hoc</i>	55
Figura 31. Taxa de transmissão do LifeDrive na rede Wi-Fi ad hoc.....	56
Figura 32. Consumo de bateria do LifeDrive na rede Wi-Fi <i>ad hoc</i>	57
Figura 33. Comparação entre os dois PDAs nas três redes.	59
Figura 34. Resultados da simulação com Bluetooth.....	62
Figura 35. Resultados experimentais com Bluetooth.	63
Figura 36. Resultados da simulação na rede Wi-Fi.	64
Figura 37. Resultados experimentais nas redes Wi-Fi.	64
Figura 38. Caracterização dos PDAs e tecnologias de acesso.....	69
Figura 39. Fatos e regras da base de conhecimento.	70
Figura 40. Formulário de consulta.....	72
Figura 41. Formulário com os resultados da consulta.	72
Figura 42. Trecho de código que implementa as dez etapas dos experimentos.	A-2
Figura 43. Trecho do código fonte da função <i>enviar()</i>	A-3
Figura 44. Interface do módulo Mobile.....	A-4
Figura 45. Trecho de código da classe <i>PDAStressServer</i>	A-5
Figura 46. Trecho de código da classe <i>ThreadClient</i> - Envio.	A-6
Figura 47. Trecho de código da classe <i>ThreadClient</i> - Recepção.	A-6
Figura 48. Interface gráfica do módulo <i>Server</i>	A-7

Lista de Tabelas

Tabela 1. Características da tecnologia Bluetooth.....	14
Tabela 2. Características do padrão IEEE 802.11b.	17
Tabela 3. Resultados da primeira abordagem.....	25
Tabela 4. Equações das retas que corrigem os resultados dos experimentos.....	28
Tabela 5. Resultados da segunda abordagem.	32
Tabela 6. Etapas de cada experimento.....	36
Tabela 7. Características dos PDAs estudados.....	39
Tabela 8. Formato dos logs da aplicação PDA Stress Server.....	A-7

Lista de Acrônimos

AAA	Authorization, Authentication and Accounting
ABC	Always Best Connected
ACL	Asynchronous Connectionless
AP	Access Point
API	Application Programming Interface
BS	Base Station
BSS	Basic Service Set
CIO	Chief Information Officer
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CTS/RTS	Clear To Send/Request To Send
DCF	Distributed Coordination Function
DQPSK	Differential Quadrature Phase Shift Keying
DS	Distribution System
ESS	Extended Service Set
FEC	Forward Error Correction
FTP	File Transfer Protocol
GSDR	Grupo de Sistemas Distribuídos e Redes
GUI	Graphical User Interface
HIP	Host Identity Protocol
HTTP	HyperText Transfer Protocol
IBSS	Independent Basic Service Set
IM	Instant Messenger

IP	Internet Protocol
IPSec	IP Security
ISM	Industrial Scientific Medicine
ISM	Industrial, Scientific and Medical
Java SE	Java Standard Edition
L2CAP	Logical Link Control and Adaptation Protocol
LAN	Local Área Network
LCD	Liquid Crystal Display
LMP	Link Manager Protocol
MAC	Medium Access Control
MSS	Maximum Segment Size
NAT	Network Address Transnlation
NS2	Network Simulator 2
OSI	Open Systems Interconnection
PAN	Personal Área Network
PCF	Point Coordination Function
PDA	Personal Digital Assistant
PDU	Protocol Data Unit
PHY	Physical Layer
PPP	Point-to-Point Protocol
QoS	Quality of Service
RF	Radio Frequency
RFCOMM	Radio Frequency Communication
SCO	Synchronous Connection Oriented Link
SDK	Software Development Kit

SDP	Service Discover Protocol
SIG	Special Interest Group
SIP	Session Iniation Protocol
SMTP	Simple Mail Transfer Protocol
SSD	Sistema de Suporte a Decisão
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
WLAN	Wireless Local Área Network
WPA	Wi-Fi Protected Access
WPAN	Wireless Personal Área Network
WWAN	Wireless Wide Area Networks

1 Introdução

Atualmente, dispositivos móveis apresentam diferentes características de *display* e de entrada de dados, possuem diferentes capacidades de processamento, de armazenamento, de bateria e utilizam vários tipos de sistemas de comunicação. Redes de rádio celular, redes sem fio, conexões discadas ou de banda larga, são tipicamente utilizadas para acesso à Internet, sendo que, geralmente, os dispositivos implementam mais de uma destas tecnologias.

Dentre as tecnologias de comunicação sem fio, merecem destaque os padrões IEEE 802.11 [Crow *et al.*, 1997] e IEEE 802.15.1. O primeiro é popularmente conhecido como Wi-Fi [Gast, 2002] e foi criado para oferecer serviços de rede similares aos disponíveis nas LANs (*Local Area Networks*) com fio, por meio de uma tecnologia de comunicação sem fio com elevadas taxas de transmissão (11, 54 e 200 Mbps). O segundo é mais conhecido como Bluetooth [Bluetooth, 2004] e foi criado visando à substituição dos cabos que interligam os computadores aos periféricos, *e.g.* teclado, mouse, terminal de vídeo, impressora, por uma tecnologia de comunicação sem fio com baixas taxas de transmissão (1 a 3 Mbps), mas também com baixo consumo de energia.

A próxima geração de redes sem fio, provavelmente, será um complexo sistema heterogêneo, composto por diversas tecnologias de acesso. No estado da arte, um dos principais desafios é prover a interoperabilidade entre estas as tecnologias de acesso, para permitir que os usuários permaneçam sempre conectados à melhor rede disponível (ABC - *Always Best Connected*). O conceito ABC tem o objetivo de manter o dispositivo móvel conectado à rede que ofereça as melhores condições em termos de custo, consumo de energia, QoS (*Quality of Service*) e segurança [Gustafsson *et al.*, 2003].

Devido a fatores técnicos e comerciais, a implementação do conceito ABC resulta em uma tarefa bastante complexa. Dentre tais fatores, destacam-se a dificuldade em desenvolver um algoritmo de seleção de acesso que considere tanto as características das tecnologias de comunicação sem fio quanto dos dispositivos móveis que as utilizam. Segundo Fordor *et al.* (2004), os principais desafios para implementação do conceito ABC são: a identificação dos parâmetros relevantes para seleção de acesso; o desenvolvimento um algoritmo de seleção de acesso em função destes parâmetros; e o projeto de uma arquitetura que suporte mobilidade, segurança e implemente uma entidade de seleção automática de acesso.

O presente trabalho foi realizado junto ao projeto LabPalm [LabPalm, 2006], cujo principal objetivo é o desenvolvimento de componentes de conectividade para o sistema operacional Palm OS [Foster, 2002]. Tais componentes visam à redução da complexidade no desenvolvimento de aplicações que necessitem de conectividade por meio das pilhas de protocolo Bluetooth, IrDA e/ou TCP/IP. Um dos objetivos secundários do projeto é o desenvolvimento de um mecanismo para realizar a seleção automática da tecnologia de acesso em função da quantidade de energia disponível no dispositivo móvel e da taxa de transmissão mínima para atender a aplicação. Uma vez selecionada a tecnologia, um componente de conectividade especializado é empregado para realizar a comunicação.

Neste contexto, o presente trabalho partiu da hipótese que as características de *hardware* do dispositivo móvel influenciam a capacidade de comunicação e o consumo de bateria do mesmo e, portanto, são relevantes na seleção de acesso. Dada a variedade de dispositivos móveis disponíveis, o mecanismo de seleção será mais eficiente se utilizar parâmetros específicos que caracterizam o real comportamento de cada dispositivo. Isso é necessário, pois a capacidade de processamento do dispositivo determina o volume de dados que o mesmo consegue manipular durante uma comunicação e o consumo de bateria determina sua autonomia. Dessa forma, um PDA pode não conseguir utilizar a vazão nominal

de uma determinada rede sem fio e pode não oferecer a autonomia desejada por determinada aplicação.

Com objetivo de verificar a hipótese levantada foi desenvolvida uma aplicação chamada *PDA Stress*, que, por meio de experimentos, visa à avaliação quantitativa da capacidade de comunicação e do consumo de bateria de PDAs. Foram experimentados dois PDAs com capacidades de processamento distintas em três redes: Bluetooth, Wi-Fi modo infra-estruturado e Wi-Fi modo *ad hoc*. Os resultados obtidos neste trabalho ratificam a hipótese supracitada. Como o conceito ABC está sendo desenvolvido para a próxima geração de redes sem fio, acredita-se que, hoje, a escolha do dispositivo móvel e da tecnologia de comunicação deve ser feita na fase de projeto da aplicação. Para auxiliar tal escolha, foi desenvolvida uma aplicação chamada *PDA Chooser*, que é um SSD (Sistema de Suporte à Decisão) que suporta a escolha de um PDA e tecnologia de comunicação para um determinado cenário. A base de conhecimento do SSD foi construída a partir dos resultados obtidos nos estudos de caso.

A caracterização do comportamento de dispositivos móveis em redes sem fio foi anteriormente investigada por Junior *et al.* (2005), que avalia a sobrecarga gerada na recepção de dados em uma rede Wi-Fi utilizando o protocolo IPSec (*IP Security*). Já Feeney *et al.* (2001) realiza uma investigação do consumo de energia de interfaces de rede Wi-Fi, operando em modo *ad hoc*, com objetivo de auxiliar o desenvolvimento de protocolos *energy-aware*. O presente trabalho utiliza métodos semelhantes para fins distintos e propõe um SSD para disponibilizar os resultados de forma objetiva e de fácil acesso, manutenção e atualização.

A seqüência deste trabalho está estruturada da seguinte forma: o capítulo 2 apresenta os conceitos relacionados à computação móvel, à próxima geração de redes de acesso sem fio e às tecnologias de acesso sem fio; o capítulo 3 apresenta os métodos utilizados para investigar a capacidade de comunicação e consumo de bateria de PDAs, descreve os estudos

de caso e, por fim, detalha o ambiente de simulação utilizado; o capítulo 4 apresenta os resultados obtidos nos estudos de caso e nas simulações; o capítulo 5 modela e implementa um SSD que utiliza os resultados dos experimentos para auxiliar a escolha do PDA e tecnologia de acesso que atendem a um determinado cenário; e, finalmente, no capítulo 6 são apresentadas as conclusões, incluindo direcionamentos para trabalhos futuros.

2 Conceitos Básicos

A ampla disponibilidade de tecnologias de comunicação sem fio e de dispositivos móveis permitem a implementação do paradigma computacional conhecido como Computação Móvel. O universo deste paradigma é composto por diversos dispositivos portáteis, *e.g.* telefones celulares, *smartphones*, *tablets*, *laptops* e PDAs, que implementam uma ou várias tecnologias de acesso sem fio e, dessa forma, permitem e/ou possibilitam o acesso a uma rica diversidade de aplicações enquanto o usuário se movimenta. Durante a movimentação, o usuário pode conectar-se a diferentes ambientes de redes, os quais podem se diferenciar em termos de tecnologia de comunicação, largura de banda disponível, qualidade do sinal e de mecanismos de segurança.

Neste contexto, este capítulo apresenta os principais conceitos relacionados a este trabalho e à computação móvel, à próxima geração de redes de acesso sem fio e às tecnologias de acesso sem fio. A seqüência do capítulo está estruturada da seguinte forma: a seção 2.1 apresenta os conceitos básicos relacionados à computação móvel; já a seção 2.2 apresenta as características da próxima geração de redes de acesso sem fio; a seção 2.2 concentra-se nas principais características das tecnologias de acesso Bluetooth e Wi-Fi, que foram utilizadas nos experimentos deste trabalho; e, finalmente, a seção 2.4 apresenta algumas considerações sobre o capítulo.

2.1 Computação Móvel

Imielinski *et al.* (1994) previram que em um futuro próximo aquela data, milhões de usuários estariam carregando computadores portáteis equipados com tecnologias de acesso sem fio e, dessa forma, implementariam o novo paradigma computacional denominado Computação Móvel ou Nômade. Tal paradigma permite que os usuários móveis acessem informações e recursos de rede enquanto se deslocam. Segundo Kleinrock (1995), a Computação Móvel é um sistema que atende a necessidade de computação e comunicação de usuários nômades, enquanto eles se movem de um lugar para outro de forma transparente, integrada e conveniente.

As previsões de Imielinski *et al.* (1994) são realidade nos dias atuais, devido à portabilidade e à conectividade dos dispositivos móveis, *e.g.* PDAs e *smartphones*, e a disponibilidade de tecnologias digitais de acesso sem fio. Tais elementos compõem um ambiente que possibilita o acesso permanente a uma rede fixa ou móvel em qualquer lugar e a qualquer hora, *i.e.* possibilita a mobilidade. Segundo Mateus *et al.* (2004), este paradigma está mudando a forma como trabalhamos, nos comunicamos, nos divertimos, estudamos e fazemos outras atividades quando estamos em movimento ou não desejamos ficar “presos” a uma infra-estrutura fixa de comunicação de dados. Tal mudança é produto da soma das características almejadas pela Computação Móvel que, segundo Kleirock (1995), são a independência de localização, de movimento, de plataforma e a disponibilidade de acesso a serviços de rede.

Atualmente, as características supracitadas estão parcialmente implementadas e existem vários pontos críticos que são temas de pesquisa, como por exemplo, o chaveamento entre tecnologias de acesso distintas, também chamado de *handover* vertical [Indulska *et al.*, 2003]. A mudança de rede ainda exige uma ação do usuário, *i.e.* o usuário tem que escolher a

nova rede e eventualmente definir alguns parâmetros de segurança e autenticação. Para ilustrar um cenário típico disponível atualmente imaginemos um CIO (*Chief Information Officer*), de uma empresa, que tem as seguintes atividades programadas: i) reunião com a diretoria da empresa - durante a manhã; ii) visita a um cliente para apresentar novas soluções - durante a tarde.

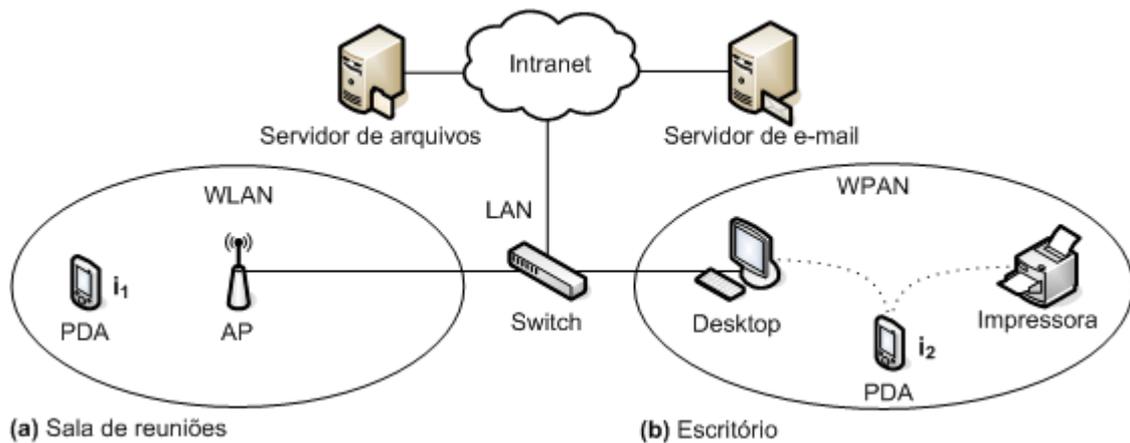


Figura 1. Exemplo de mobilidade: (a) Sala de reunião e (b) Escritório.

Durante a reunião, o CIO, conecta seu PDA a WLAN da sala de reunião, faz anotações e acessa sua caixa de e-mail corporativa; a Figura 1 (a) esboça este ambiente de rede. Terminada a reunião, ele se desloca para seu escritório e se conecta a uma WPAN, imprime alguns documentos para os diretores e transfere as notas da reunião para seu *Desktop*, utilizando um ambiente de rede parecido com o apresentado na Figura 1 (b). Durante o almoço, ele se conecta a um serviço de dados da rede de telefonia, revisa a apresentação e solicita informações para um subordinado por meio de um e-mail, utilizando o ambiente de rede esboçado na Figura 2 (c). Terminado o almoço, parado em um congestionamento, ele recebe o e-mail com as informações solicitadas e modifica a apresentação. No cliente, o CIO se conecta à WLAN disponível e, durante a apresentação, recupera dados na *Intranet* de sua empresa para sanar eventuais dúvidas do cliente, por meio do ambiente apresentado pela Figura 2 (d).

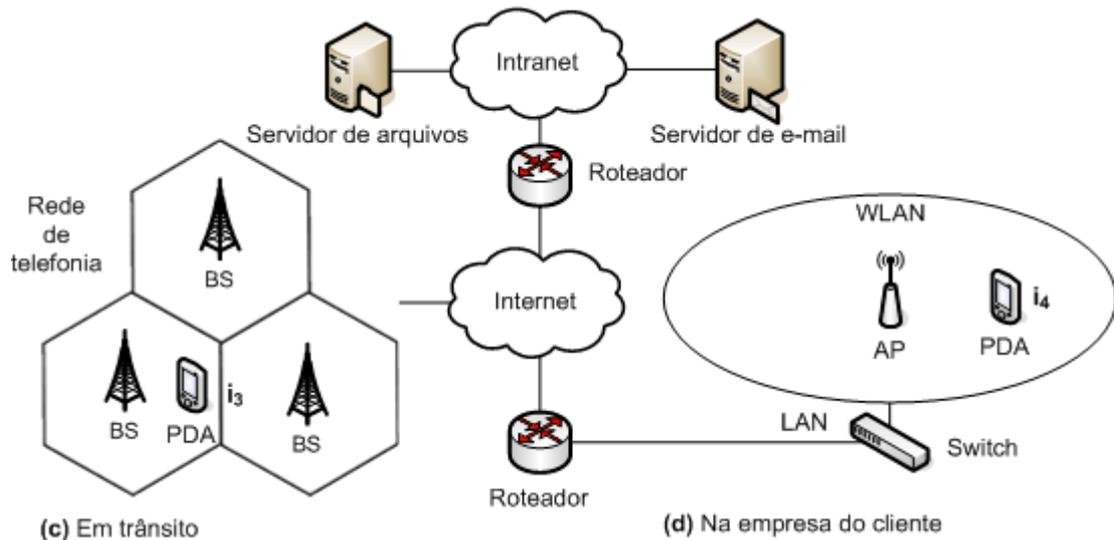


Figura 2. Exemplo de mobilidade: (c) Em trânsito e (d) Na empresa do cliente.

Este exemplo explicita os principais elementos envolvidos em um ambiente móvel, que segundo Mateus *et al.* (2004) são três: o tipo e a capacidade de processamento do dispositivo móvel, a mobilidade do usuário e da unidade móvel, e a comunicação com outro elemento computacional por meio de um canal de comunicação sem fio. Existem diversos tipos de dispositivos móveis (*e.g.* telefones celulares, *smartphones* e PDAs), cuja utilidade está estreitamente relacionada à capacidade de processamento e de armazenamento disponível para as aplicações. A mobilidade está relacionada à portabilidade e autonomia do equipamento, à velocidade que o usuário se desloca no ambiente e à área de cobertura do ambiente de rede. Por fim, a vazão de dados suportada pelo canal de comunicação sem fio determina as características das informações que podem ser transmitidos na rede com eficiência. Em suma, o produto destes três elementos limita as funcionalidades que uma aplicação pode oferecer ao usuário móvel.

No estado da arte, a Computação Móvel é uma área de pesquisa consolidada, que abrange diversos esforços para melhorar a experiência dos usuários nômades. Tais esforços estão focados em temas que variam do desenvolvimento de tecnologias de comunicação mais eficientes, até questões sociais sobre a utilização do paradigma. Para Mateus *et al.* (2004) a

Computação Móvel envolve todas as áreas da Ciência da Computação e criou problemas que não existiam até então. Isso ocorre devido à herança deixada pelos paradigmas anteriores, onde foram desenvolvidas técnicas que naturalmente são empregadas no novo paradigma, mas que, eventualmente, podem se mostrar pouco eficientes ou ineficientes neste novo contexto. Dessa forma, todas as áreas estão adaptando suas técnicas e/ou criando novas técnicas para atender o novo paradigma, gerando uma imensa demanda científica.

2.2 Próxima Geração de Redes Sem Fio

A próxima geração de redes sem fio, conhecida como B3G (*Beyond Third Generation*) ou 4G (*Fourth Generation*), busca a interoperabilidade entre diversas tecnologias de comunicação, para permitir que seus usuários permaneçam sempre conectados à melhor rede disponível (ABC - *Always Best Connected*). O conceito ABC tem o objetivo de manter o dispositivo móvel conectado à rede que ofereça as melhores condições em termos de custo, consumo de energia, QoS (*Quality of Service*) e segurança [Gustafsson *et al.*, 2003].

Para Fordor *et al.* (2004), as redes ABC devem permitir que os usuários se conectem a serviços por meio da tecnologia de comunicação sem fio mais apropriada, em termos de alguns parâmetros pré-definidos. Tais parâmetros podem ser definidos pelo usuário e/ou pelo contexto (ambiente de rede). Ao usuário, cabe a definição das características de QoS da aplicação, preferências pessoais e/ou características do dispositivo móvel em questão. O contexto deve permitir a recuperação de informações como: os pontos de acesso disponíveis, qualidade dos sinais e percentual de utilização da rede e demais parâmetros de QoS.

Outra contribuição de Fordor *et al.* (2004) é a identificação dos dois requisitos chaves para implementação do conceito ABC, que são: o desenvolvimento de algoritmos para a seleção automática do acesso, e a implementação de uma arquitetura que suporte o conceito.

Os algoritmos devem garantir que os usuários sempre estejam com a melhor conectividade disponível em função de parâmetros definidos pelo usuário e pelo contexto. A arquitetura deve permitir a descoberta de pontos de acesso, administrar a mobilidade e implementar uma entidade de seleção de acesso que receba os parâmetros e execute um algoritmo para seleção da melhor rede.

No estado da arte, os algoritmos para seleção de acesso propostos focam-se no atendimento dos requisitos de QoS do usuário (aplicação) em questão [Furuskär *et al.*, 2002, 2005] e [Fordor *et al.*, 2004]. Além disso, uma arquitetura foi proposta por Gustafsson *et al.* (2003), conforme apresenta a Figura 3. Nesta figura são apresentadas as entidades que compõem a arquitetura do lado do dispositivo móvel e da rede ABC.

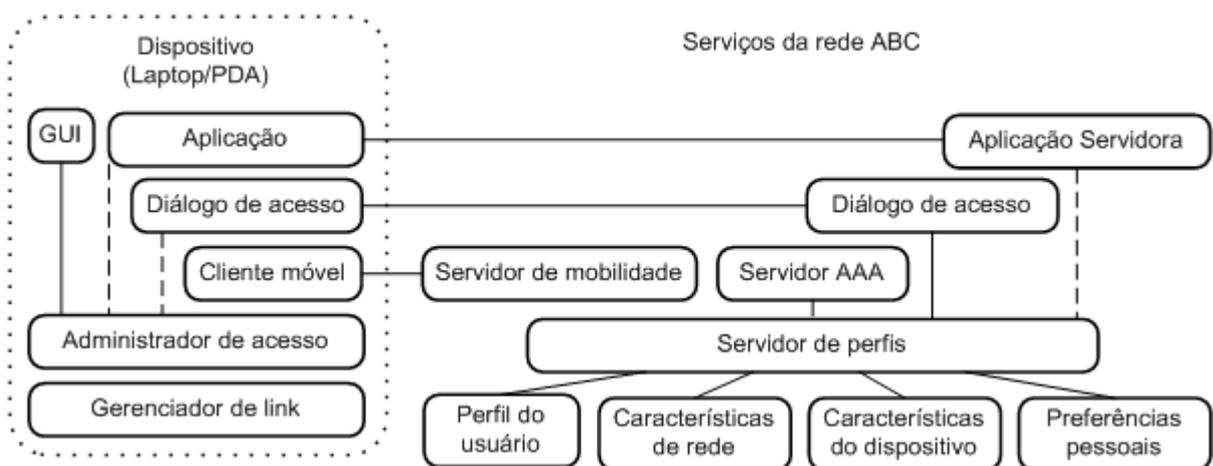


Figura 3. Arquitetura funcional de um ambiente ABC.

Do lado do dispositivo móvel, tem-se um gerenciador de *link* responsável pela descoberta de pontos de acesso; uma vez identificados, o administrador de acesso faz a seleção e administração da conexão. Para permitir a mobilidade faz-se necessária a implementação de um cliente de algum mecanismo de mobilidade, *e.g.* *Mobile IP*, *HIP (Host Identity Protocol)* ou *SIP (Session Initiation Protocol)* [Dias *et al.*, 2006]. Também é prevista uma interface gráfica (*GUI – Graphical User Interface*) para configuração do gerenciador de acesso e um diálogo para definição dos parâmetros de conexão ou seleção manual da conexão.

Do lado da rede ABC, tem-se um servidor de perfis que é responsável pela gerência dos perfis de usuários, das preferências pessoais, das características dos dispositivos móveis e das redes. Um servidor AAA (*Authorization, Authentication and Accounting*) é o responsável pela autorização de acesso, pela autenticação e pelo gerenciamento de privilégios e recursos da conta. O servidor de mobilidade implementa o lado servidor de algum mecanismo que ofereça mobilidade. E, finalmente, o diálogo de acesso permite que o usuário defina os parâmetros da conexão.

Com esta arquitetura espera-se que o usuário móvel faça um único *logon* no sistema e obtenha a melhor conectividade enquanto se movimenta, de forma transparente e conveniente, *i.e.* sem a necessidade de intervenção do usuário durante os eventuais *handovers*. Se o CIO, do exemplo apresentado na seção anterior (seção 2.1), estivesse utilizando uma rede ABC, não seria necessária uma nova conexão a cada mudança de ambiente. Na realidade, o conceito vai além, permitindo que usuário não perca a conexão enquanto se movimenta. Dessa forma, será possível iniciar o *download* de um arquivo em uma rede e mantê-lo durante os eventuais *handovers* verticais que serão necessários enquanto o usuário está se movimentando.

Em suma, a próxima geração de redes de acesso sem fio provavelmente serão complexos sistemas heterogêneos, compostos por várias tecnologias de acesso sem fio e habitados por dispositivos móveis que implementarão mais de uma tecnologia de comunicação e executarão diversas aplicações baseadas na Internet. É consenso da comunidade científica que o protocolo IP continuará sendo amplamente utilizado para homogeneizar o sistema na camada de rede. Dessa forma, estas redes serão baseadas nos princípios, serviços e protocolos da Internet, formando as chamadas *All-IP Networks*.

2.3 Tecnologias de Acesso Sem Fio

As tecnologias de acesso sem fio são elementos centrais na provisão de mobilidade para usuários móveis, pois permitem a criação de redes que oferecem serviços sem a utilização de cabos; o meio físico é o ar e os dados são sinalizados por meio de ondas eletromagnéticas. Geralmente, estas redes não substituem as redes com fio e sim as complementam, ampliando a área de cobertura da rede sem a necessidade de obras civis e oferecendo mobilidade aos seus usuários. Por outro lado, as questões de segurança são críticas nessas redes uma vez que dados sensíveis podem ser enviados em *broadcast* para fora dos limites físicos da instituição em questão.

Quanto à área de cobertura, estas redes são tipicamente classificadas em redes metropolitanas (WWANs - *Wireless Wide Area Networks*), redes locais (WLANs - *Wireless Local Area Networks*) e redes pessoais (WPANs - *Wireless Personal Area Networks*). A área de cobertura é determinada pela potência do sinal que carrega a portadora; já a vazão de dados esta relacionada à largura de banda, aos mecanismos de multiplexação e modulação, à concorrência de acesso ao meio e à distância do ponto de acesso.

A vazão de dados oferecida por uma rede sem fio tende a decrescer em função da distância entre o elemento computacional e o ponto de acesso. Isso acontece porque durante a propagação do sinal pelo meio ocorre um fenômeno denominado espalhamento do sinal, que tende a reduzir o nível de potência do sinal com o quadrado da distância. Contudo, em ambientes fechados, existem obstáculos que podem oferecer resistência e, com isso, acelerar a atenuação do sinal, devido à ocorrência de fenômenos como reflexão e difração [Stein, 2003].

Considerando PDAs, a classe de dispositivos móveis alvo do presente trabalho, Bluetooth e Wi-Fi são tecnologias de acesso típicas. Desta forma, esta seção apresenta as

principais características dos padrões IEEE 802.15.1 (subseção 2.3.1) e IEEE 802.11 (subseção 2.3.2) que foram utilizados em experimentos neste trabalho. Por fim, a subseção 2.3.3 apresenta os cenários típicos para as duas tecnologias abordadas.

2.3.1 IEEE 802.15.1 - Bluetooth

O padrão IEEE 802.11.5.1, mais conhecido como Bluetooth [Bluetooth, 2004], foi criado visando à substituição dos cabos que interligam os computadores aos periféricos, *e.g.* teclado, mouse, terminal de vídeo e impressora, por uma tecnologia de comunicação sem fio de baixo custo, com pequeno alcance e com baixas taxas de transmissão, mas também com baixo consumo de energia. Contudo, atualmente Bluetooth se consolidou como uma tecnologia de comunicação sem fio que permite a criação de redes *ad hoc* com suporte à comunicação de dados síncronos (SCO – *Synchronous Connection-Oriented*) e assíncronos (ACL – *Asynchronous Connectionless*).

As principais características da tecnologia são apresentadas na Tabela 1 (adaptada de [Muller, 2000]). Bluetooth utiliza um mecanismo de dispersão de espectro com saltos de frequência (FHSS - *Frequency Hopping Spread Spectrum*) na faixa de frequência de 2.4 GHz da banda ISM (*Industrial, Scientific and Medical*). O espectro é dividido em 79 ou 23 canais e o rádio salta sistematicamente entre estes canais utilizando uma seqüência pseudo-randômica, com objetivo de evitar eventuais interferências nos canais. Além disso, utiliza modulação GFSK (*Gaussian Frequency Shift Keying*) para prover um *link* com velocidade de 1 Mbps [Bhagwat, 2001]. Foram previstas três classes de interfaces de rede interoperáveis que se diferenciam pela área de cobertura -100, 20 e 10 metros – e, conseqüentemente, pela potência utilizada pelo rádio – 100, 2,5 e 1 mW (mili Watt) - classes 1, 2 e 3, respectivamente.

Tabela 1. Características da tecnologia Bluetooth.

Atributos	Características
Tipo de conexão	Dispersão de espectro (FHSS)
Faixa de frequência	Banda ISM de 2.4 GHz
Potência de transmissão	100mW, 2,5 mW e 1 mW
Taxa de transmissão	1 Mbps
Área de cobertura	10, 20, 100 metros
Número de estações	Oito por <i>piconet</i> . As piconets podem se conectar formando <i>scatternets</i> .
Segurança	Para autenticação, chave de 128 bits; para criptografar, chave configurável de 8 até 128 bits.

Quanto à topologia, o conjunto de até oito dispositivos compartilhando o mesmo canal de comunicação forma uma rede chamada *piconet*, conforme apresenta a Figura 4. O dispositivo central chamado de mestre é o responsável por encaminhar os dados entre os demais integrantes da rede, chamados de escravos. Para criar uma rede com mais de oito nós, cria-se uma nova *piconet* e um dos dispositivos será responsável por criar uma ponte entre as duas redes, que juntas formam uma *scatternet*.

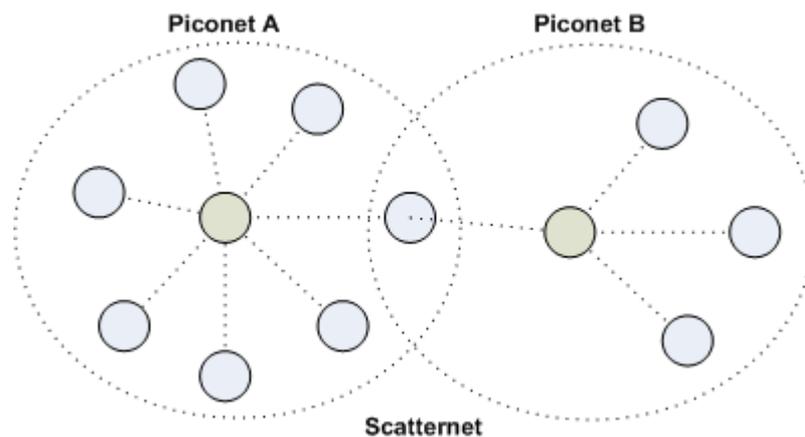


Figura 4. Duas piconets formando uma scatternet.

A pilha de protocolos Bluetooth é apresentada na Figura 5. Nesta figura, tem-se na base o Rádio Bluetooth, seguido da camada *Base Band* e da LMP (*Link Manager Protocol*). Tipicamente, estas três camadas são implementadas no *hardware* e os demais protocolos são implementados por meio de *software*. O padrão especifica as características do Rádio Bluetooth, tais como, frequência de operação, canais de comunicação, níveis de potência permitidos para transmissão e nível de sensibilidade do receptor. A camada *Base Band*

implementa a camada física (PHY) e o mecanismo de controle de acesso ao meio (MAC – *Medium Access Control*), incluindo tarefas como descoberta de dispositivos, formatação do *link* e comunicação síncrona e assíncrona. Já o protocolo LMP (*Link Manager Protocol*) define um conjunto de mensagens utilizadas para configuração e gerenciamento da *Base Band*. Em seguida, tem-se uma interface chamada HCI (*Host Controller Interface*) que disponibiliza uma interface padrão para controle do *hardware* Bluetooth.

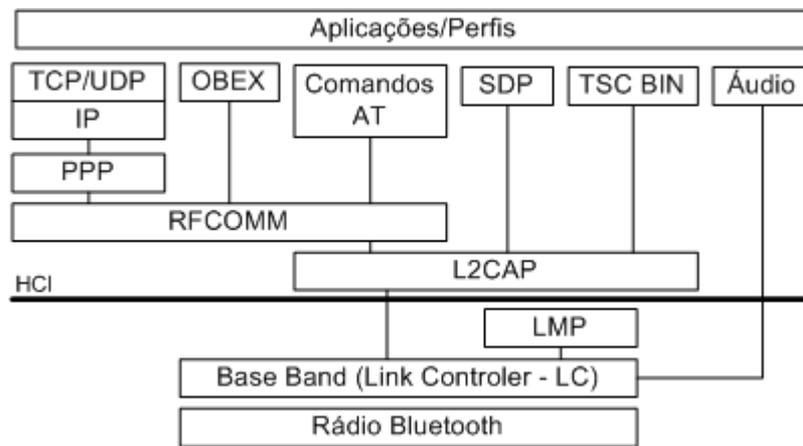


Figura 5. Pilha de protocolos Bluetooth.

A camada L2CAP (*Logical Link Control & Adaptation Protocol*) provê comunicação orientada a pacotes, em um canal de comunicação assíncrono (ACL), para as camadas superiores. Uma vez realizada a conexão L2CAP, o nó cliente tem que descobrir os serviços que o nó servidor disponibiliza por meio de um protocolo chamado SDP (*Service Discovery Protocol*).

O protocolo RFCOMM (*Radio Frequency Communication Protocol*) faz uso do protocolo L2CAP e define métodos para emular a conexão serial especificada pelo padrão RS-232. Este padrão permite a transmissão serial de dados binários e é, geralmente, implementado nas portas de comunicação do tipo COM. Com isso, periféricos e protocolos que necessitam de comunicação serial podem utilizar a tecnologia Bluetooth. Um exemplo é o protocolo PPP (*Point-to-Point Protocol*) que utiliza uma linha serial para oferecer uma interface orientada a pacotes para as camadas superiores incluindo a pilha TCP/IP.

O SIG (*Special Interest Group*) [Bluetooth SIG, 2002] responsável pelo desenvolvimento da tecnologia Bluetooth, definiu treze perfis de serviços que, geralmente, são utilizados por aplicações. Entre os perfis de serviços estão à transferência de arquivos, à emulação da porta serial e o acesso a LAN. Estes perfis utilizam os protocolos mais adequados da pilha para prover determinado serviço e, dessa forma, facilita o desenvolvimento de aplicações. Para ilustrar, a aplicação desenvolvida, para implementar os experimentos propostos neste trabalho, utilizou o serviço de acesso a LAN para criar uma conexão TCP em um *link* Bluetooth. Este serviço só permite conexões ponto-a-ponto, ou seja, não permite a comunicação em *piconets*, uma vez que as conexões TCP/IP são encapsuladas em um túnel PPP que, por sua vez, utiliza uma conexão RFCOMM.

2.3.2 IEEE 802.11 – Wi-Fi

As WLANs tipicamente utilizam tecnologias de acesso sem fio concebidas por meio da implementação de um dos padrões da família IEEE 802.11, cujo objetivo é servir de guarda-chuva para a definição de vários padrões que especificam desde a camada física e protocolos de acesso ao meio até mecanismos para prover QoS e segurança em enlaces sem fio (*e.g.* infra-vermelho e frequências de rádio). Estes padrões são utilizados pela indústria para compor uma tecnologia de acesso sem fio popularmente conhecida como Wi-Fi. Esta tecnologia provê altas taxas de transmissão como, por exemplo, 11, 54 e 200 Mbps com os padrões IEEE 802.11 b, g e n, respectivamente. A interoperabilidade entre produtos de fabricantes distintos é garantida por uma organização, sem fins lucrativos, chamada *Wi-Fi Alliance* [Wi-Fi, 2006].

Os experimentos realizados neste trabalho utilizaram o padrão IEEE 802.11b. Este padrão utiliza a faixa de frequência de 2.4 GHz da banda ISM, multiplexação OFDM

(*Orthogonal Frequency Division Multiplexing*) e modulação DQPSK (*Differential Quadrature Phase Shift Keying*) [Proakis, 2001] para alcançar taxas de até 11 Mbps. Disponibiliza um conjunto de mecanismos de segurança denominado WPA (*Wi-Fi Protected Access*), que, resumidamente, pode prover integridade, confidencialidade e autenticidade por meio da utilização de técnicas de criptografia. Além disso, o padrão prevê que durante a transmissão de dados, o rádio pode dissipar até 100 mW (mili Watts), conforme apresenta a Tabela 2 (adaptada de [Muller, 2000]).

Tabela 2. Características do padrão IEEE 802.11b.

Atributos	Características
Tipo de multiplexação	Divisão ortogonal de frequências (OFDM)
Faixa de frequência	Banda ISM de 2.4 GHz
Potência de transmissão	100 mW
Taxa de transmissão	1, 2 e 11 Mbps
Área de cobertura	Depende da antena utilizada
Número de estações	Várias estações por AP e múltiplos APs por rede
Segurança	WPA

O padrão define dois modos de operação: *ad hoc* e infra-estruturado. Nas redes *ad hoc*, os nós são independentes e não existe uma entidade centralizadora. Geralmente, estas redes são compostas por um número pequeno de elementos computacionais, são criadas para uma de finalidade bem definida e por um curto espaço de tempo. Já nas redes infra-estruturadas, toda a comunicação entre os nós é mediada por uma entidade central denominada AP (*Access Point*). Ao contrário das redes *ad hoc*, as redes infra-estruturadas podem conter vários elementos computacionais e, tipicamente, são permanentes.

Três topologias básicas foram especificadas no padrão: IBSS (*Independent Basic Service Set*); BSS (*Basic Service Set*); e ESS (*Extended Service Set*). Uma rede configurada em modo *ad hoc* forma uma IBSS, conforme apresenta a Figura 6 (a). Nesta figura, tem-se uma rede sem fio composta por três dispositivos móveis que se comunicam de forma independente. Uma rede configurada em modo infra-estruturado forma uma BSS, conforme

esboça a Figura 6 (b). Esta figura apresenta uma rede formada por dois dispositivos e um AP, de forma que toda comunicação entre os nós da rede é realizada por meio do AP. Um grupo de duas ou mais BSSs formam uma ESS, conforme apresenta a Figura 7.

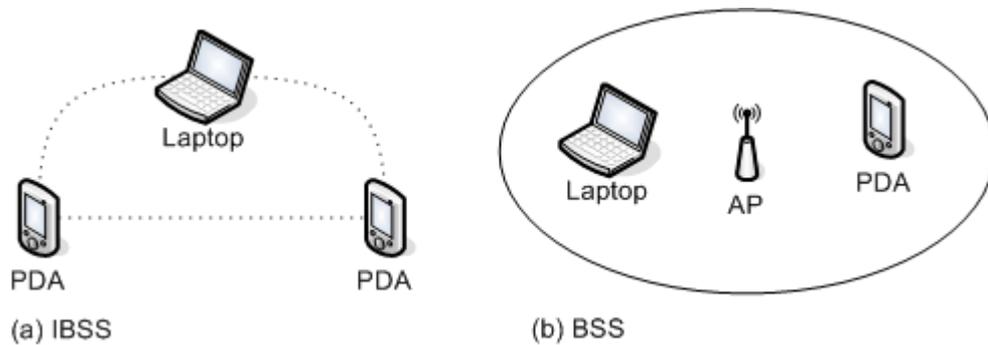


Figura 6. Topologias básicas do padrão 802.11.

Geralmente, uma BSS oferece serviços em uma área relativamente pequena (*e.g.* um auditório, um escritório ou uma casa), por outro lado, as ESSs permitem a cobertura de uma área maior, por meio da comunicação entre duas ou mais células BSSs. Neste ambiente, os APs são configurados para realizar a função de ponte (*Bridge*), e estão interligados por meio de um *backbone* que é denominado sistema de distribuição (DS – *Distribution System*).

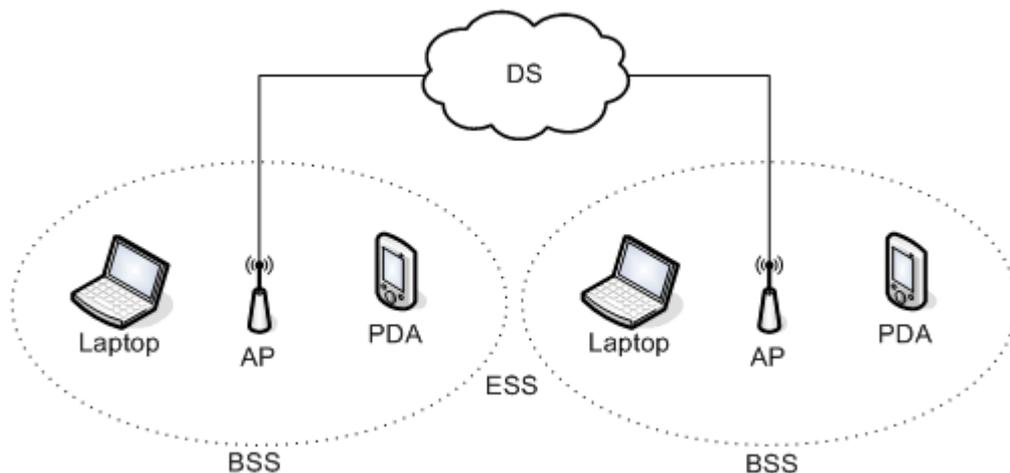


Figura 7. Conexão de duas BSS formando uma ESS.

Atualmente, os padrões da família 802.11 estão, geralmente, disponíveis nos equipamentos portáteis (*e.g.* *laptops*, *smartphones* e PDAs) e são utilizados em WLANs de universidades, empresas e aeroportos, possibilitando fácil acesso à Internet para os usuários móveis. Esta tecnologia, também é bastante utilizada na borda de redes *mesh* [Akyildiz *et al.*,

2005]. Redes *mesh* são redes sem fio compostas por diferentes tecnologias de acesso sem fio, cujas principais características são a auto-configuração, auto-reparo e auto-organização. No Brasil, estas redes estão sendo implementadas em lugares onde não é possível lançar cabos (*e.g.* cidades históricas e favelas) e, tipicamente, possuem o *backbone* implementado com o padrão IEEE 802.16 e, na borda, tem-se diversos *hotspots* IEEE 802.11 para o acesso efetivo da população.

2.3.3 Cenários Típicos

As tecnologias de acesso sem fio Bluetooth e Wi-Fi permitem a criação de WPANs e WLANs, respectivamente. Geralmente a área de cobertura destas redes se restringe ao espaço físico de uma sala ou prédio, oferecendo aos usuários mobilidade de até algumas centenas de metros quadrados. Estas restrições são relevantes durante o desenvolvimento de aplicações móveis, pois, eventualmente, o usuário pode sair da área de cobertura e desejar continuar utilizando a aplicação.

Neste contexto, aplicações móveis podem ser projetadas em função da infra-estrutura de rede disponível. Neste caso, dois cenários são típicos. No primeiro cenário, Figura 8 (a), a aplicação é sempre utilizada em um ambiente com cobertura de uma rede sem fio. Dessa forma, a lógica de negócios e os dados podem estar em um servidor de retaguarda e o PDA os acessa por meio de um *browser* ou de chamadas remotas, de forma que a aplicação depende da rede para funcionar. No segundo cenário, Figura 8 (b), em um momento t_0 a aplicação pode ser utilizada em um ambiente sem cobertura de rede e eventualmente em um momento t_1 se deslocar para um ambiente onde exista cobertura de rede ou vice-versa. Neste caso, a aplicação móvel deve implementar as regras de negócios e a persistência de dados no dispositivo e eventualmente pode sincronizar os dados com o servidor de retaguarda.

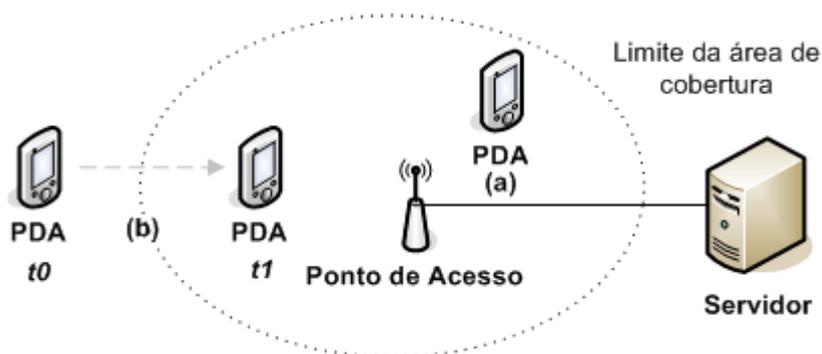


Figura 8. Cenários típicos.

Os experimentos descritos neste trabalho foram realizados no cenário (a) com o objetivo de quantificar os limites do PDA. Portanto, o mesmo sempre está conectado a uma rede sem fio realizando o envio ou a recepção de dados. O objetivo desses experimentos é realizar uma avaliação quantitativa da taxa de transmissão e do consumo de bateria nas redes Bluetooth e Wi-Fi, o próximo capítulo (Capítulo 3) descreve os métodos utilizados nesta avaliação.

Além disso, este trabalho também discute, no capítulo 5, a modelagem e implementação de um SSD, denominado *PDA Chooser*, para auxiliar a escolha do PDA e tecnologia de comunicação que atenda a um dado cenário. Neste caso, um cenário é caracterizado por duas variáveis: taxa de transmissão e autonomia da bateria. A taxa de transmissão refere-se à vazão mínima de dados necessária para atender uma aplicação móvel em questão; já a autonomia da bateria refere-se ao tempo de uso da aplicação sem a necessidade de recarregar a bateria do PDA.

Para caracterizar um cenário espera-se que, na fase de projeto de uma aplicação móvel, o especialista responsável consiga identificar o volume médio de dados que será trafegado na rede sem fio e, com isso, estimar a taxa de transmissão mínima para atender a aplicação. Já a autonomia da bateria exige uma observação do ambiente onde a aplicação será utilizada e os turnos de trabalho dos usuários com objetivo de estimar o intervalo de tempo em que o PDA deverá funcionar sem a necessidade de recarga de bateria.

2.4 Considerações Finais

O presente capítulo apresentou brevemente os conceitos básicos relacionados à Computação Móvel, destacando o impacto deste paradigma no dia-a-dia dos usuários e pesquisadores. Em seguida, listou os principais desafios da implementação da próxima geração de redes sem fio, especificamente os relacionados à implementação do conceito ABC. Para finalizar, foram apresentadas as principais características dos padrões IEEE 802.15.1 e 802.11b, que especificam as tecnologias de acesso sem fio Bluetooth e Wi-Fi, na devida ordem. As duas tecnologias foram utilizadas em estudos de caso neste trabalho. Tais estudos de caso são descritos no próximo capítulo.

3 Investigando o Comportamento de PDAs

O presente trabalho nasceu no contexto do projeto LabPalm com o objetivo inicial de abordar o problema da escolha automática da tecnologia de acesso. A idéia central era implementar um mecanismo de seleção de acesso, no nível da aplicação, que realizasse a escolha do acesso em função da quantidade de energia disponível no dispositivo e da taxa de transmissão mínima necessária para atender a aplicação em questão. Tal mecanismo visa à maximização da autonomia do equipamento, garantindo a vazão de dados esperada pela aplicação.

Neste sentido, iniciou-se um esforço para investigar o comportamento de PDAs durante a transmissão de dados com tecnologias de acesso sem fio, especificamente Bluetooth e Wi-Fi. O foco foi a observação das taxas de transmissão e do consumo de bateria, visando à obtenção da eficiência do equipamento durante a comunicação, dada em unidades de dados por unidade de energia.

Conhecida a eficiência do dispositivo, com cada tecnologia de acesso suportada, e a quantidade de energia disponível, é possível estimar a autonomia do equipamento a partir de um ponto no tempo. Dessa forma, um mecanismo de seleção pode utilizar estes dados para realizar a escolha do acesso mais adequado no momento. Contudo, as informações sobre o estado atual das redes seriam desconsideradas (*e.g.* qualidade do sinal, taxa de uso e demais parâmetros de QoS). Além disso, seria necessário conhecer a eficiência de cada dispositivo, uma vez que eles possuem características diversas e conseqüentemente apresentam comportamentos distintos.

Neste contexto, o presente trabalho partiu da hipótese que as características de *hardware* do dispositivo móvel influenciam a capacidade de comunicação e o consumo de

bateria do mesmo e, portanto, são relevantes na seleção de acesso. Dada a variedade de dispositivos móveis disponíveis, o mecanismo de seleção teria que utilizar parâmetros específicos para caracterizar o real comportamento de cada modelo de dispositivo. Isso é necessário, pois a capacidade de processamento do equipamento determina o volume de dados que o mesmo consegue manipular durante a comunicação e o consumo de bateria determina sua autonomia. Portanto, um PDA pode não conseguir utilizar a vazão nominal de uma determinada rede sem fio e pode não oferecer a autonomia desejada por determinada aplicação.

Com objetivo de verificar esta hipótese foram desenvolvidos experimentos que visam à avaliação quantitativa da capacidade de comunicação e do consumo de bateria de PDAs durante a transmissão de dados em redes sem fio. As próximas seções descrevem os métodos utilizados, com a seguinte organização: a seção 3.1 descreve dois métodos para monitorar o consumo de bateria durante a comunicação; a seção 3.2 apresenta os experimentos que visam à avaliação da capacidade de comunicação dos PDAs; a seção 3.3 mostra o ambiente experimental configurado; a seção 3.4 apresenta o ambiente de simulação adotado; e, finalmente, a seção 3.5 apresenta as considerações finais do capítulo.

3.1 Investigando o Consumo de Bateria

Os PDAs são equipamentos autônomos equipados com baterias, cuja autonomia está estreitamente relacionada à capacidade da bateria e ao consumo de energia dos componentes de *hardware*, como por exemplo, processador, *display* e interfaces de rede. Dessa forma, a eficiência energética do dispositivo depende das suas características de *hardware*. Eficiência energética é a razão entre a quantidade de trabalho e a quantidade de energia consumida para realizar tal trabalho. No contexto deste trabalho, eficiência energética refere-se ao volume de

dados que um PDA consegue enviar e/ou receber por unidade de energia consumida nesta tarefa.

Neste sentido, foram experimentados dois métodos para monitorar o consumo de bateria durante a transmissão de dados em uma rede sem fio. O primeiro método, apresentado na subseção 3.1.1, utiliza uma chamada de sistema para recuperar o percentual de bateria disponível no dispositivo e o segundo método, apresentado na subseção 3.1.2, utiliza um *hardware* para monitorar a corrente elétrica no condutor que interliga a bateria ao PDA para assim obter a potência utilizada.

3.1.1 Utilizando a API do Sistema Operacional

A idéia central deste método é utilizar a API (*Application Programming Interface*) do sistema operacional para recuperar o estado da bateria e assim quantificar o consumo de bateria. Neste trabalho, foi utilizado um PDA fabricado pela Palm Inc., cujo sistema operacional é o Palm OS [Foster, 2002]. Este sistema operacional disponibiliza em seu SDK (*Software Development Kit*) a API 68K [Wilson *et al.*, 2003]. Esta API implementa uma função chamada *SysBatteryInfo()* que retorna algumas informações sobre a bateria e, entre elas, o percentual de bateria disponível no PDA.

Com objetivo de verificar a viabilidade deste método, um experimento preliminar foi implementado. Neste experimento, o PDA recupera sistematicamente o percentual de bateria disponível, por meio da função *SysBatteryInfo()*, monta uma mensagem e a envia para um servidor. A cada trinta segundos o servidor registra o percentual de bateria disponível, o tempo transcorrido e o subtotal de dados recebidos. A mensagem é construída com o valor retornado pela função supracitada mais a adição de um *padding*. Este *padding* é adicionado para criar uma mensagem do tamanho do MSS (*Maximum Segment Size*) negociado no

handshake TCP, neste caso 536 Bytes. Na verdade, a mensagem é um *buffer* estático e, a cada iteração, é adicionado o percentual de bateria disponível, dessa forma, é eliminada a sobrecarga gerada pela alocação e preenchimento do *buffer*.

Este experimento foi realizado com o PDA modelo Zire 72s, protocolo TCP, em duas redes: Bluetooth e Wi-Fi *ad hoc*. Para ambas as tecnologias o experimento foi realizado até que a bateria do PDA se esgotasse e repetido cinco vezes, as médias e os desvios padrões são apresentadas na Tabela 3.

Tabela 3. Resultados da primeira abordagem.

Variáveis	Bluetooth		Wi-Fi	
	Resultado	Desv. Pad.	Resultado	Desv. Pad.
Dados (MB)	145,68	9,71	420,10	45,05
Autonomia (h)	3,87	0,09	1,99	0,13
Taxa (KB/s)	10,71	0,45	60,27	8,71
Eficiência (KB/J)	45,04	-	129,89	-

As autonomias médias do PDA foram próximas de 4 e 2 h, na rede Bluetooth e Wi-Fi, respectivamente, ou seja, a autonomia do PDA na rede Bluetooth é aproximadamente duas vezes superior à autonomia na rede Wi-Fi. Por outro lado, o volume médio de dados transmitidos, durante este tempo, foi em torno de 420 MB, com Wi-Fi, e próximo de 145 MB, com Bluetooth, em outras palavras, com Wi-Fi, o dispositivo transmitiu um volume de dados quase três vezes superior ao volume de dados observado com Bluetooth. Isso ocorreu, porque o PDA, na rede Wi-Fi, apresentou taxas de transmissão quase cinco vezes superiores às recuperadas na rede Bluetooth. Tal comportamento resultou em eficiências próximas de 45 e 130 KB/J, na rede Bluetooth e Wi-Fi, na devida ordem, *i.e.* o PDA, na rede Wi-Fi, se mostrou quase três vezes mais eficiente que na rede Bluetooth.

Este método é longo e os resultados permitem apenas uma análise superficial do consumo de energia do dispositivo. Outra desvantagem, é a falta de precisão da função *SysBatteryInfo()*, que pode ser observada na Figura 9. Nesta figura, tem-se o percentual de bateria disponível em função do tempo na rede Bluetooth, em (a), e na rede Wi-Fi, em (b).

Em (a) e (b), no instante 0 tem-se 100 % da bateria disponível; conforme o tempo passa, o percentual de bateria diminui. Contudo, em (a), depois de 187 minutos a função retorna 0 % de bateria disponível, mas o PDA continua transmitindo dados por mais 45 minutos. O mesmo ocorre em (b), depois de 64 minutos a função retorna 0 % de bateria disponível, mas o PDA continua transmitindo dados por mais 52 minutos.

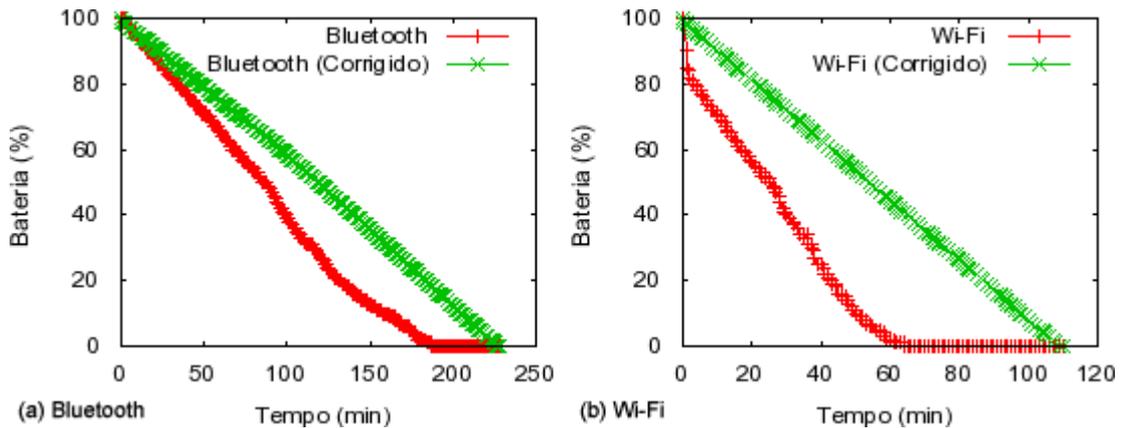


Figura 9 . Percentual de bateria em função do tempo.

Para corrigir este erro foi utilizada uma técnica simples da Geometria Analítica Plana que consiste na obtenção da equação da reta por meio de dois pontos conhecidos. Tal abordagem é adotada, pois era esperado um comportamento linear no consumo de bateria, uma vez que o PDA está realizando a mesma atividade sistematicamente. Os dois pontos são o instante inicial A e o final B . No instante inicial tem-se zero minuto de experimento e 100 % da bateria disponível, $A(0, 100)$. Já no instante final, tem-se X minutos de experimento e 0 % de bateria disponível, $B(X, 0)$, onde X é a autonomia da bateria.

Thomas, em [Thomas, 2002] explica que dados dois pontos $A(x_1, y_1)$ e $B(x_2, y_2)$, o coeficiente angular m da reta que passa pelos dois pontos é dado por:

$$m = \frac{y_2 - y_1}{x_2 - x_1}$$

A equação reduzida da reta pode ser recuperada por:

$$y = m.x + w$$

Onde w é o coeficiente linear da reta que é a ordenada do ponto $(0, w)$. Para ilustrar, tomemos os resultados obtidos na rede Wi-Fi apresentados pela Figura 9 (b). Para este caso os pontos são $A(0, 100)$ e $B(110, 0)$. O ponto A representa o início do experimento onde o tempo é 0 e o PDA está com 100 % de carga de bateria. O ponto B representa o final do experimento onde o tempo é aproximadamente 110 minutos e a carga da bateria é zero. Dessa forma, o coeficiente angular da reta é:

$$m = \frac{0 - 100}{110 - 0} = \frac{-50}{55}$$

E a equação da reta é:

$$y = \frac{-50x}{55} + 100$$

Esta equação supostamente retorna o real percentual de bateria disponível no dispositivo em função do tempo. Tais resultados podem ser observados na Figura 9 (b), a mesma correção foi feita para Bluetooth e é apresentada na mesma figura em (a).

Outra forma de visualizar os resultados é apresentada na Figura 10. Nesta figura, tem-se o percentual de bateria em função do volume de dados transmitidos, na rede Bluetooth, em (a), e na rede Wi-Fi, em (b). Dado o volume de dados total, é possível calcular a o volume de dados transmitido por unidade de energia, *i.e.* a eficiência energética durante a comunicação. O fabricante da bateria disponibiliza a capacidade máxima do produto; no caso do Zire 72s é de 920 mAh (mili ampere hora). Os experimentos quantificaram o volume médio de dados transmitidos com a carga de uma bateria. Dado o volume de dados e a capacidade da bateria é possível calcular a eficiência E com:

$$E_{\text{tecnologia}} = \frac{D_{\text{total}}}{(C_{\text{bateria}} * 3600)}$$

Onde D é o volume de dados total (KB), C é a capacidade da bateria declarada pelo fabricante (Ah) e o número 3.600 se refere ao número de segundos em uma hora. Dessa forma, o PDA tem eficiências de aproximadamente 45 e 135 KB/J, com Bluetooth e Wi-Fi, respectivamente, conforme apresentou a Tabela 3.

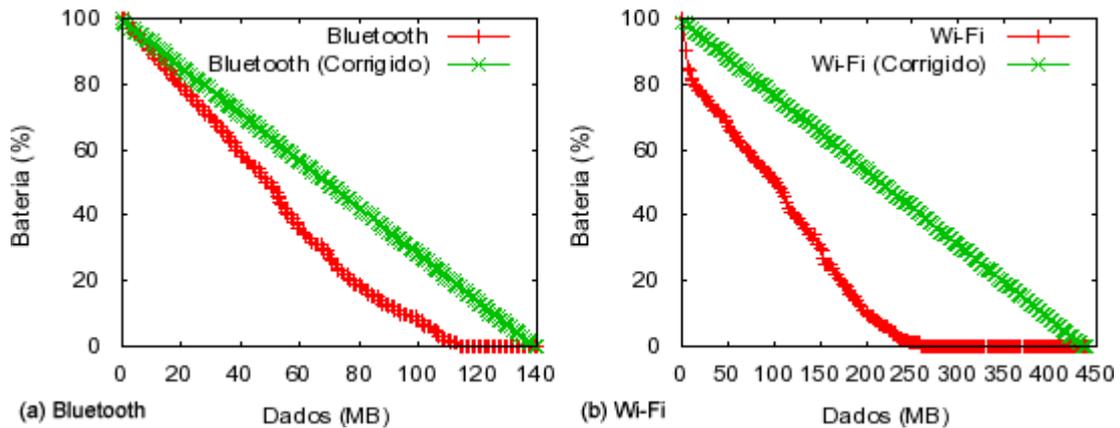


Figura 10. Percentual de bateria em função do volume de dados.

A mesma técnica de correção foi aplicada em todos os resultados, conforme apresentam as figuras 9 e 10. Tais correções foram obtidas por meio das equações das retas que supostamente descrevem o real comportamento do PDA durante os experimentos, todas as equações são apresentadas na Tabela 4. Estas equações lineares são de fácil implementação e podem ser úteis para estimar a autonomia do dispositivo. Para tanto, basta calcular a inversa das equações que retornam o percentual de bateria em função do tempo, com isso tem-se uma equação que calcula o tempo em função do percentual de bateria.

Tabela 4. Equações das retas que corrigem os resultados dos experimentos.

Variáveis	Bluetooth	Wi-Fi
Bateria em função do tempo	$y = \frac{-25x}{57} + 100$	$y = \frac{-50x}{55} + 100$
Bateria em função dos dados	$y = \frac{-5x}{7} + 100$	$y = \frac{-50x}{219} + 100$

Os resultados corrigidos das duas redes são apresentados na Figura 11. Nesta figura, em (a), tem-se o percentual de bateria em função do tempo e, em (b), tem-se o percentual de

bateria em função dos dados transmitido. Estas retas decrescentes foram obtidas pelas equações da Tabela 4.

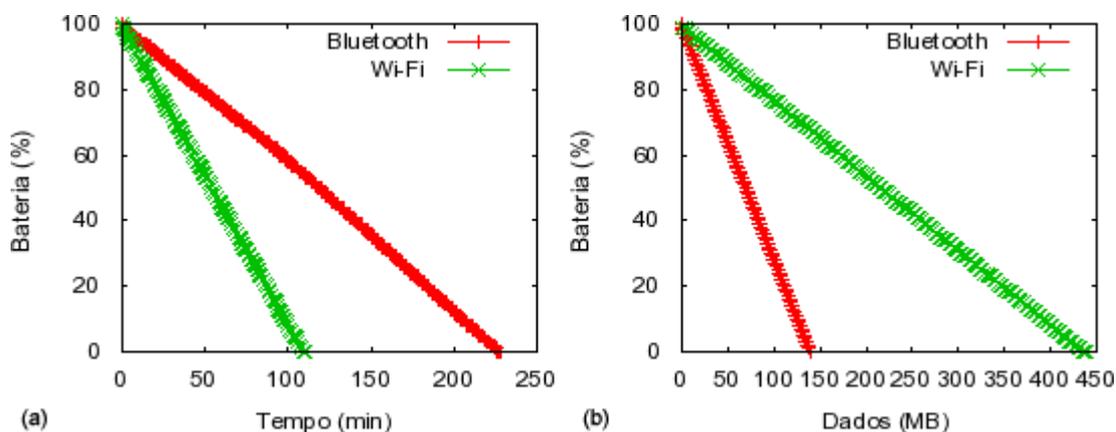


Figura 11. Comportamento do PDA nas redes Bluetooth e Wi-Fi.

Em suma, os experimentos mostram que o PDA apresentou a melhor autonomia na rede Bluetooth e, por outro lado, apresentou a melhor vazão na rede Wi-Fi. Tal comportamento era esperado, uma vez que o rádio Bluetooth consome menos energia que o rádio Wi-Fi; 2,5 mW e 100 mW, respectivamente. Além, disso a rede Wi-Fi tem vazão nominal de 11 Mbps, contra 1 Mbps da rede Bluetooth. Dadas as diferenças, parece que os resultados obtidos com esta abordagem são coesos. Contudo, resta saber qual é a influência da sobrecarga, gerada pelas chamadas sistemáticas a função *SysBatteryInfo()*, na comunicação e no consumo de bateria. A próxima seção apresenta outro método, menos intrusivo, para monitorar o consumo de energia.

3.1.2 Monitorando a Potência Dissipada

A idéia central deste método é monitorar a potência dissipada pelo PDA durante a comunicação, na rede sem fio, sem utilizar a API do sistema operacional. Dessa forma, espera-se reduzir o tempo de execução do experimento e obter resultados mais precisos. Para tanto, foi adotado um método semelhante aos apresentados por Hager e Junior, em [Hager *et*

al., 2005] e [Junior *et al.*, 2005], respectivamente. O trabalho de Hager quantifica o consumo de bateria de PDAs durante a execução de algoritmos de criptografia distintos. Junior investiga a sobrecarga no consumo de energia acrescentada pela utilização de IPSec, para prover confidencialidade em um fluxo UDP (*User Datagram Protocol*). As medições foram feitas durante a recepção de dados com um PDA em uma rede Wi-Fi (IEEE 802.11b). Utilizou-se alguns algoritmos de criptografia, tamanhos de PDU (*Protocol Data Unit*) e velocidades do fluxo.

Hager mede o consumo de bateria acrescentando uma resistência de $0,025 \Omega$ em série entre o PDA e uma fonte de energia. Um multímetro digital mede a voltagem na resistência 10.000 vezes por segundo. Já Junior, mensura o consumo de bateria por meio da voltagem de saída da bateria e dos dados da corrente elétrica, os quais são obtidos por meio de um osciloscópio. Para monitorar a corrente, um resistor do tipo *shunt* de 0.1Ω foi empregado.

No presente trabalho, o consumo de bateria foi recuperado por meio de um *hardware* desenvolvido no contexto do projeto LabPalm, cuja a foto é apresentada na Figura 12. Este *hardware* monitora a utilização da bateria acrescentando uma resistência R , de baixa resistividade, no condutor que liga a bateria ao PDA. Um circuito mede a diferença de potencial v_r entre um ponto antes da resistência e outro ponto depois da resistência. A diferença de potencial é amplificada e em seguida é transformada em sinal digital por um conversor Analógico/Digital (ADC0808). Um microcontrolador de 8 bits ajusta o valor em uma escala de 0 a 255, serializa os dados e os envia para a porta serial de um computador. Com estas variáveis, a potência instantânea P , pode ser recuperada pelo produto da voltagem de entrada (V_{in}) pela corrente:

$$P(t) = V_{in} \frac{v_r(t)}{R}.$$

Já o consumo de energia em um intervalo de tempo $t_0...t_1$ pode ser recuperado por meio da integral da potência no intervalo de tempo [Feeney *et al.*, 2001]:

$$E_{t_0...t_1} = \frac{V_{in}}{R} \int_{t_0}^{t_1} v_r(t) dt.$$



Figura 12. Hardware que monitora o consumo de bateria.

Dada a energia E (Joule) consumida no intervalo de tempo pode-se obter o percentual da capacidade total da bateria que este montante representa. Assim é possível realizar uma comparação entre os dois métodos experimentados. Com objetivo de realizar esta comparação, um experimento semelhante ao descrito na subseção 3.1.1 foi implementado, a única diferença é que não é necessário recuperar o percentual de bateria disponível por meio da função *SysBatteryInfo()*. Dessa forma, elimina-se a sobrecarga gerada pela chamada de sistema durante a criação da mensagem de 536 Bytes (MSS da conexão); nesta abordagem a mensagem é um *buffer* estático.

Este experimento também foi realizado com o PDA Zire 72s, protocolo TCP e nas duas redes: Bluetooth e Wi-Fi *ad hoc*. Ao contrário do outro método, o experimento consistiu na transmissão de 3 MB de dados, com ambas as tecnologias, e com cinco repetições. Portanto, os experimentos ficaram menos intrusivos, significativamente mais curtos e, por outro lado, se tornaram dependentes de informação da capacidade da bateria disponibilizada

pelo fabricante. As médias das potências e taxas de transmissão recuperadas são apresentadas na Tabela 5, as demais variáveis foram estimadas.

Tabela 5. Resultados do segundo método.

Variáveis	Bluetooth		Wi-Fi	
	Resultado	Desv. Pad.	Resultado	Desv. Pad.
Dados (MB)	559,43	-	1.948,24	-
Potência (mW)	35,45	12,02	318,8	43,25
Autonomia (h)	14,31	-	5,55	-
Taxa (KB/s)	11,12	0,13	99,85	0,10
Eficiência (KB/J)	172,96	-	602,35	-

As autonomias médias observadas com o PDA foram próximas de 14,31 e 5,55 h, na rede Bluetooth e Wi-Fi, respectivamente, ou seja, a autonomia com Bluetooth é aproximadamente 2,5 vezes superior a autonomia com Wi-Fi, conforme mostra o gráfico na Figura 13 (a). Por outro lado, o volume médio de dados transmitidos durante este tempo foi em torno de 1,9 GB com Wi-Fi e próximo de 560 MB com Bluetooth, em outras palavras, Wi-Fi transmitiu um volume de dados cerca de 3,4 vezes maior que o volume de dados transmitido com Bluetooth, conforme apresenta o gráfico na Figura 13 (b). Além disso, Wi-Fi apresentou uma taxa de transmissão aproximadamente nove vezes maior que Bluetooth. Estes resultados indicam eficiências de 173 e 602 KB/J, na rede Bluetooth e Wi-Fi, na devida ordem. Portanto, o PDA se mostrou quase três vezes e meia mais eficiente na rede Wi-Fi.

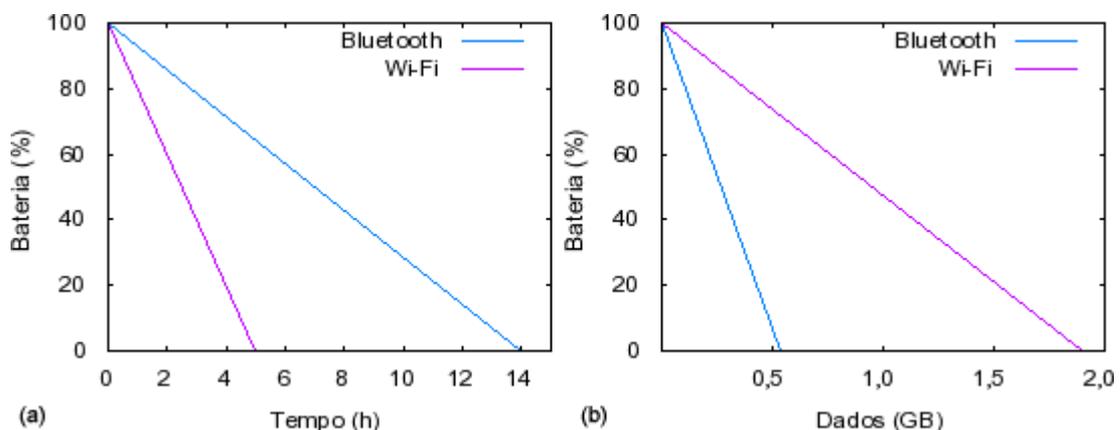


Figura 13. Comportamento do PDA nas redes Bluetooth e Wi-Fi.

A Figura 13 apresenta, em (a), o percentual de bateria em função do tempo e, em (b), o percentual de bateria em função dos dados. Estes dados foram estimados por meio da potência

média dissipada pelo PDA e pela capacidade da bateria disponibilizada pelo fabricante. A próxima subseção apresenta uma comparação entre os dois métodos experimentados.

3.1.3 Comparação Entre os Dois Métodos

Esta comparação se concentra na discussão de dois pontos: tempo de experimentação e confiança dos resultados obtidos. Quanto ao tempo de experimentação, o primeiro método realiza a transmissão de dados até a bateria se esgotar, ou seja, necessita de um tempo para recarga da bateria. Os experimentos levaram, em média, cerca de quatro e duas horas, com Bluetooth e Wi-Fi, respectivamente. Para obter-se o tempo total de experimentação soma-se o tempo médio nas duas redes e multiplica-se por cinco (número de repetições de cada ensaio), totalizando aproximadamente trinta horas de experimentação, sem levar em consideração os intervalos para recarga da bateria. Por outro lado o segundo método utilizou um volume de dados de três MB durante a realização dos experimentos, *i.e.* os experimentos duraram cerca de cinco minutos com Bluetooth e cerca de trinta segundos com Wi-Fi, totalizando aproximadamente trinta minutos de experimentação.

Quanto à confiança dos resultados, os dois métodos apresentaram resultados distintos, sendo que o segundo método obteve resultados superiores nas duas redes, conforme apresenta a Figura 14. Nesta figura, têm-se os valores médios, da autonomia e volume de dados, com o primeiro método *A* (Bluetooth *A* e Wi-Fi *A*) e com o segundo método *B* (Bluetooth *B* e Wi-Fi *B*). A autonomia média observada foi de 3,87 e 14,31 h, com Bluetooth, e 1,99 e 5,55 h, com Wi-Fi, com o primeiro e segundo método, na devida ordem. Estes resultados mostram que o segundo método obteve uma autonomia quase 4 e 3 vezes maior, com Bluetooth e Wi-Fi, respectivamente. Isso influenciou diretamente o volume de dados transmitido que foram próximos de 146 e 559 MB, na rede Bluetooth, e 420 e 1.948 MB, na rede Wi-Fi, com o

primeiro e segundo método, respectivamente. Dessa forma, o segundo método consegue transmitir um volume de dados cerca de 4 e 5 vezes superior ao primeiro método, com Bluetooth e Wi-Fi, respectivamente. Além disso, na rede Wi-Fi, observa-se que a diferença entre os métodos, na autonomia e volume de dados, não são proporcionais, isso ocorreu porque a taxa de transmissão no segundo método também foi superior. Contudo, na rede Bluetooth, as diferenças são praticamente proporcionais, indicando que não ocorreu mudança significativa na taxa de transmissão e sim no consumo de energia; tal fenômeno foi gerado pela sobrecarga no uso de uma conexão TCP em um túnel PPP sobre um *link* Bluetooth. Pode-se concluir que a sobrecarga gerada pelas chamadas sistemáticas da função *SysBateriaInfo()* teve influência significativa nos resultados do primeiro método.

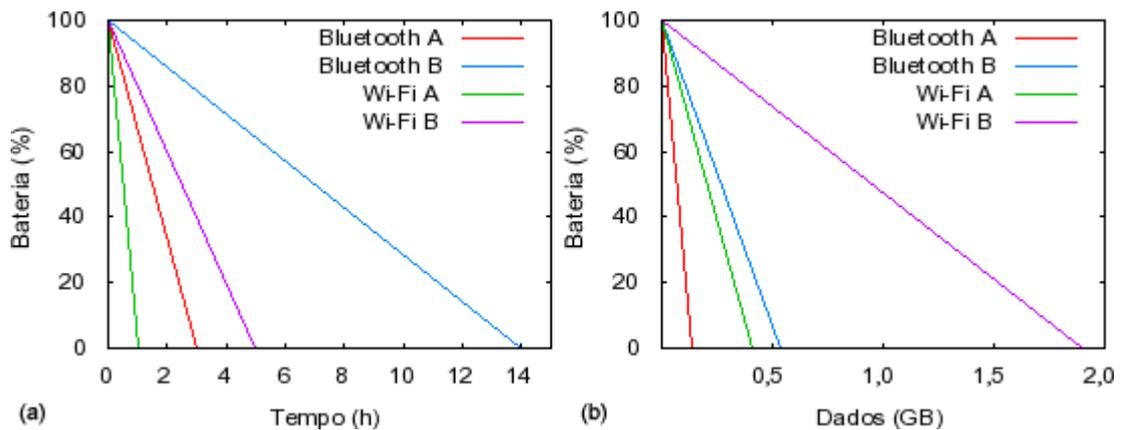


Figura 14. Comparação do comportamento do PDA nas duas abordagens.

A dissemelhança entre os dois métodos, conseqüentemente, é refletida na eficiência do dispositivo nas duas redes, conforme apresenta a Figura 15. Nesta figura, observa-se que as eficiências foram de 45 e 173 KB/J, na rede Bluetooth, e 130 e 602 KB/J, na rede Wi-Fi, com o primeiro e segundo método, respectivamente. Portanto, com o segundo método o PDA foi cerca de 2,8 e 3,4 vezes mais eficiente durante a comunicação.

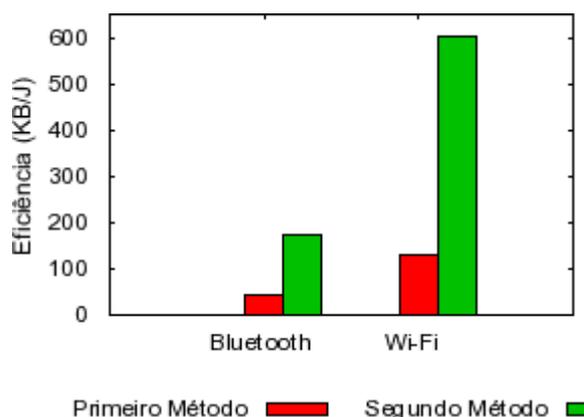


Figura 15. Eficiência do PDA com os dois métodos nas duas redes.

Portanto, adotou-se o segundo método (subseção 3.1.2) para recuperar o consumo de energia dos dispositivos nas redes sem fio. A próxima seção apresenta os experimentos implementados para realizar a avaliação quantitativa da taxa de transmissão.

3.2 Investigando a Capacidade de Comunicação

A avaliação quantitativa da capacidade de comunicação foi realizada por meio de dois experimentos. O primeiro ensaio avalia o envio de 3 MB de dados, enquanto o segundo avalia a recepção da mesma quantidade de dados. O volume de dados escolhido como conveniente foi determinado de forma empírica, após a análise de alguns resultados iniciais. Para obtenção dos primeiros resultados foram realizados experimentos com volumes de dados variando de 1 a 8 MB, onde foi possível observar que trabalhando com volumes superiores a 2 MB, os resultados permanecem praticamente constantes, *i.e.* os resultados oscilaram para volumes de dados menores que 2 MB e foram similares para volumes maiores. Portanto, adotou-se 3 MB visando à minimização do tempo de experimentação.

Com o objetivo de verificar se a sobrecarga de fragmentação dos dados interfere na taxa de transmissão ou no consumo de bateria, foram utilizados *buffers* variando de 64 Bytes a 32 KB. Sendo assim, os experimentos foram divididos em dez etapas, onde os 3 MB foram

enviados utilizando-se um *buffer*, na camada de aplicação, de tamanho variável, cujo tamanho foi dobrado a cada nova etapa. O número de repetições é obtido em função do tamanho do *buffer*, conforme mostra a Tabela 6. Nesta tabela, tem-se na etapa 1 o *buffer* de 64 Bytes que é enviado e/ou recebido 49.152 vezes; conforme o tamanho do *buffer* dobra, o número de repetições diminui pela metade, sendo que na última etapa tem-se um *buffer* de 32 KB e o número de repetições é de 96 vezes.

Tabela 6. Etapas de cada experimento.

Etapa	Tamanho do buffer (Bytes)	Número de repetições
1	64	49.152
2	128	24.576
3	256	12.288
4	512	6.144
5	1.024	3.072
6	2.048	1.536
7	4.096	768
8	8.192	384
9	16.384	192
10	32.768	96

Estes experimentos foram implementados, com o protocolo TCP, em uma aplicação chamada *PDA Stress*. Esta aplicação possui dois módulos: o módulo *Server* e o módulo *Mobile*. O módulo *Server* é executado em computadores convencionais, e comunica-se, enviando e recebendo dados, com o módulo *Mobile*, executado no PDA; este ambiente de rede é esboçado na Figura 16. O módulo *Server* ainda é responsável por armazenar *logs* da comunicação.

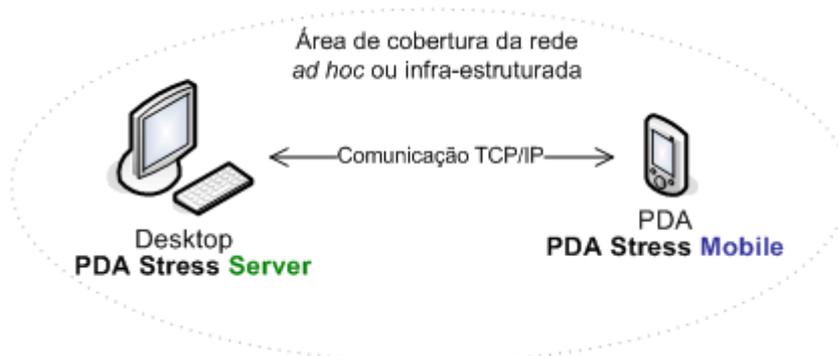


Figura 16. Ambiente de rede onde a aplicação *PDA Stress* é empregada.

O TCP foi adotado porque é um protocolo amplamente empregado na Internet, especialmente em aplicações que necessitam de comunicação fim-a-fim confiável. Dessa forma, vários protocolos da camada de aplicação utilizam o TCP, entre eles estão o HTTP (*HyperText Transfer Protocol*), o SMTP (*Simple Mail Transfer Protocol*) e o FTP (*File Transfer Protocol*). Durante um *handshake* TCP são negociados os parâmetros da conexão, sendo que, no contexto deste trabalho merece destaque o parâmetro MSS (*Maximum Segment Size*) [Postel *et al.*, 1981], que define o tamanho máximo dos segmentos de dados trafegados durante o envio ou a recepção de dados em uma rede.

Os principais trechos de código da implementação dos dois módulos da aplicação *PDA Stress* estão descritos no anexo denominado “Implementação dos Experimentos”. A próxima seção descreve os ambientes de rede onde foram realizados os experimentos descritos nesta seção.

3.3 Ambiente Experimental

Esta seção apresenta o ambiente experimental onde foram realizados os ensaios descritos na seção 3.2. Durante a comunicação, o consumo de energia dos PDAs foi recuperado com o emprego do método descrito na subseção 3.1.2.

Foram desenvolvidos três estudos de caso baseados nos cenários de redes mostrados na Figura 17. Nesta figura, em (a), tem-se uma rede *ad hoc* entre um PDA e um *laptop*, a qual foi configurada com tecnologias de acesso distintas (Bluetooth e Wi-Fi); em (b), tem-se uma rede Wi-Fi infra-estruturada formada por um ponto de acesso, um *gateway*, um *hub*, um *laptop* e um PDA; o ponto de acesso está conectado ao *gateway* por meio de uma interface de rede com fio, e o *gateway* está conectado ao *laptop* por meio de um *hub*.

Dessa forma, o ambiente experimental é composto por três redes distintas: Bluetooth, Wi-Fi *ad hoc* e Wi-Fi infra-estruturada. Também é possível observar que o PDA e o *laptop* estão nas três redes, sendo que o primeiro é o responsável por executar o módulo *Mobile*, da aplicação *PDA Stress*, e o no segundo é executado o módulo *Server*.

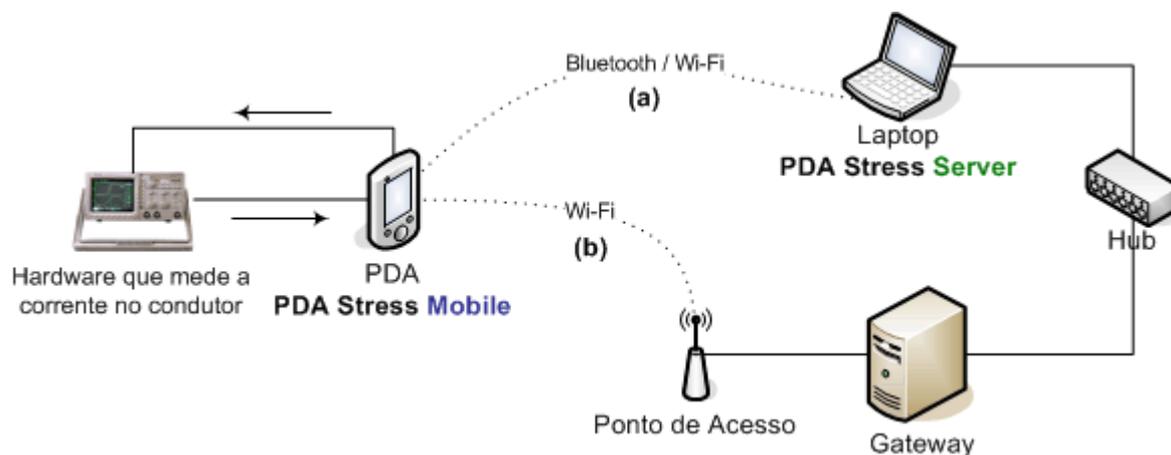


Figura 17. Ambiente experimental: (a) Redes ad hoc e (b) Rede infra-estruturada.

Os equipamentos presentes neste ambiente são: dois PDAs, um Zire 72s e um LifeDrive, ambos fabricados pela Palm Inc.; um ponto de acesso DLink DWL-2000AP+ que implementa o padrão IEEE 802.11g; uma interface USB Bluetooth *ENCORE Eletronics* de classe 1 (até 100 metros de cobertura); um *laptop*, equipado com duas interfaces de rede, uma sem fio Intel® Pro/Wireless 3945ABG (IEEE 802.11g), e outra *Fast Ethernet Realtek* RTL8139/810x; um computador equipado com duas interfaces de rede *Fast Ethernet Realtek* RTL8139/810x; e, por fim, um *hardware* desenvolvido para monitorar o consumo de energia.

As características de *hardware* dos dois PDAs experimentados merecem destaque, uma vez que eles são o objeto de estudo do presente trabalho. A Figura 18 apresenta uma foto dos dois dispositivos; à esquerda, Zire 72s e, à direita, LifeDrive. Nesta foto, observa-se que os equipamentos possuem *display* de tamanhos distintos e que o Zire utiliza uma interface Wi-Fi externa, por outro lado, no LifeDrive a interface é embutida. As duas interfaces de rede implementam o padrão IEEE 802.11b, utilizam o mesmo *chipset* e, segundo o fabricante, permitem alcance de 37 e 122 metros, em ambientes *indoor* e *outdoor*, respectivamente.

Ambos dispositivos possuem interface Bluetooth integrada, conforme descreve a Tabela 7. Estas interfaces implementam a versão 1.1 do padrão e são de classe 2 (até 20 metros de alcance). Quanto à capacidade de armazenamento, ambos utilizam memória principal volátil com 32 e 68 MB, para o Zire e o LifeDrive, na devida ordem, sendo que o último tem um HD (*Hard Disk*) de 4 GB.



Figura 18. Foto dos dois PDA s: Zire 72s (à esquerda) e LifeDrive (à direita).

Quanto à capacidade de processamento, ambos os dispositivos possuem um processador Intel® PXA270. Contudo, o processador do Zire tem *clock* de 312 MHz e o do LifeDrive tem *clock* de 416 MHz, ou seja, o último tem um processador 25 % mais rápido que o primeiro. Quanto à bateria, ambos possuem bateria de 3.7 Volts, mas a bateria do Zire tem capacidade de 920 mAh, enquanto a do LifeDrive tem capacidade de 1.660 mAh, *i.e.* o LifeDrive tem uma bateria com capacidade cerca de 45 % superior à do Zire.

Tabela 7. Características dos PDAs estudados.

Características	Zire 72s	LifeDrive
Tamanho do LCD	320x320	320x480
Bluetooth (1.1)	Integrado	Integrado
Wi-Fi (802.11b)	Interface externa	Integrado
Memória principal	32 MB	64 MB
Memória secundária	-	4 GB (HD)
Clock do processador	320 MHz	416 MHz
Bateria	3.7 V, 920 mAh	3.7 V, 1.660 mAh

No contexto do presente trabalho, merecem destaque a capacidade de processamento e a autonomia da bateria dos PDAs. A velocidade do processador impacta na fragmentação dos dados, no cálculo das seqüências de verificação de erros e nos *loops* de envio e/ou recepção de dados. Enquanto a capacidade da bateria define a autonomia do dispositivo, *i.e.* maior ou menor tempo sem a necessidade de recarga.

3.4 Ambientes de Simulação

Foram criados dois ambientes de simulação, o primeiro é uma rede Bluetooth e o segundo é uma rede Wi-Fi. O principal objetivo é confrontar os resultados obtidos em um ambiente controlado com os dados obtidos em um ambiente real. Por meio desse confronto, espera-se explicitar as restrições impostas pelo *hardware* dos PDAs e identificar eventuais anomalias.

Os experimentos foram realizados em dois pontos distintos das redes, sendo que o primeiro estava localizado a dois metros do ponto de acesso e o segundo a doze metros. Já o ambiente de simulação foi configurado para contemplar um número maior de pontos nas redes, conforme mostra a Figura 19. Nesta figura, tem-se os pontos onde foram realizados os experimentos e os pontos onde foram realizadas as simulações.

Quanto ao comportamento, o nó, que representa o PDA, afastou-se dez metros, do ponto de acesso, a cada trinta segundos, na rede Wi-Fi, e, cinco metros no mesmo espaço de tempo, na rede Bluetooth. Antes de cada deslocamento foi computada a taxa de transmissão alcançada no ponto em questão. Observa-se que o ambiente de simulação tem dois pontos em comum com o ambiente de experimentação, permitindo a discussão dos resultados nestes

pontos. Além disso, este ambiente extrapola o ambiente de experimentação. É importante ressaltar que a realização destas medições em um ambiente real, similar ao proposto para as simulações, resulta em uma atividade bastante trabalhosa e sujeita a erros.

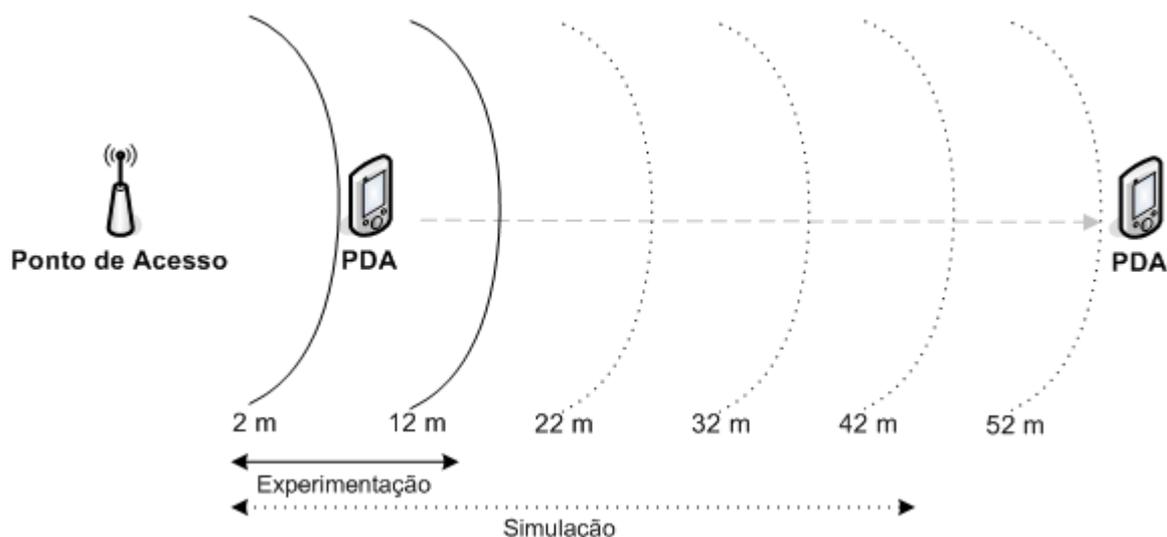


Figura 19. Pontos de medição no ambiente experimental e no ambiente de simulação.

As simulações foram realizadas com o simulador NS2 (*Network Simulator 2*) [NS2, 2006], que permite a simulação da camada física e de acesso ao meio especificadas dos padrões IEEE 802.11 e 802.15. Além disso, simula os protocolos da pilha TCP/IP e permite a configuração de parâmetros específicos da interface de rede como a potência de transmissão e ganho da antena. Para cada tecnologia de acesso foram criados dois ambientes de simulação um *indoor* e outro *outdoor*. A diferença é o modelo de propagação adotado, para o ambiente *outdoor* utilizou-se o modelo *Two-ray ground* e para o ambiente *indoor* utilizou-se o modelo *Shadowing*.

Os dois modelos de propagação são utilizados para estimar a potência do sinal onde é sinalizado um quadro na rede sem fio. O modelo *Two-ray ground* assume que raramente existirá uma linha de visada entre os nós comunicantes na rede, portanto o modelo leva em consideração o caminho direto e a reflexão na superfície do planeta. A potência do sinal é estimada a uma distância d por meio de funções determinísticas. Já o modelo *Shadowing* assume que a potência é uma variável aleatória, devido a fenômenos como espalhamento do

sinal, reflexão e difração [Fall *et al.*, 2006]. Tais fenômenos ocorrem tipicamente em ambientes fechados.

Dentre os parâmetros de configuração da interface de rede, merece destaque o *RXThreshold* que define o limiar da potência de um sinal que pode ser recebido, ou seja, se um quadro for recebido com uma potência inferior a este limite ele é descartado na camada física. Este parâmetro é calculado em função do modelo de propagação e do raio da área de cobertura da interface de rede. O simulador disponibiliza uma ferramenta chamada *threshold* que realiza este cálculo. Segundo o fabricante, a interface Wi-Fi, utilizada pelos PDAs experimentados, abrange áreas com raio de 37 e 122 metros, em ambiente *indoor* e *outdoor*, respectivamente. Enquanto a interface Bluetooth é classe 2, *i.e.* abrange uma área com raio de até 20 metros. Por meio destes dados foram recuperados o *RXThreshold* de cada ambiente.

3.5 Considerações Finais

Este capítulo, em um primeiro momento, descreveu o contexto em que este trabalho foi concebido. A idéia central é desenvolver um mecanismo de seleção automática de acesso, no nível da camada de aplicação, visando à maximização da autonomia dos PDAs. Contudo, observações iniciais indicaram que PDAs com características de *hardware* distintas apresentavam comportamentos diferentes durante a comunicação. Neste sentido, iniciou-se um esforço para quantificar este comportamento, com foco na capacidade de comunicação e no consumo de bateria. Em um segundo momento, foram apresentados os métodos utilizados para realizar a caracterização do comportamento dos PDAs. Especificamente, foram descritos dois métodos para monitoramento do consumo de bateria e dois experimentos para verificar a capacidade de comunicação. Tais experimentos foram implementados em uma aplicação denominada *PDA Stress*. Finalmente foi apresentado o ambiente experimental e os ambientes

de simulação. O próximo capítulo apresenta e discute os resultados obtidos nestes dois ambientes.

4 Resultados

Os experimentos propostos neste trabalho foram realizados no laboratório do projeto LabPalm, alocado no prédio do DC-UFSCar, que dispõe de aproximadamente doze metros quadrados de área, conforme esboça a Figura 20. Nesta figura, observa-se que os ensaios foram realizados em dois pontos do ambiente, sendo que o primeiro está localizado a dois metros do AP (*laptop* no caso das redes *ad hoc*) e o segundo está localizado a doze metros. Tais pontos foram escolhidos por conveniência e porque observações iniciais mostraram que dez metros de diferença já promoviam uma variação no comportamento dos PDAs durante a comunicação, especialmente com a tecnologia Bluetooth. Além disso, o esboço mostra que no ambiente experimental estão dispostos alguns obstáculos, *e.g.* paredes de alvenaria, portas de madeira e janelas de alumínio, que podem interferir na potência do sinal onde são sinalizados os quadros de dados e de controle das redes sem fio.

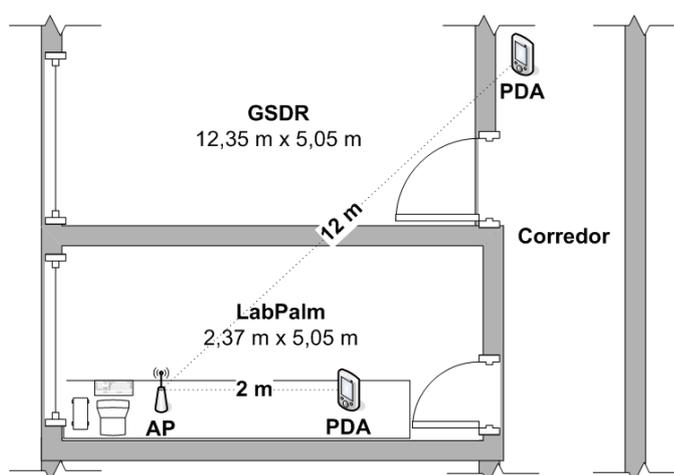


Figura 20. Esboço do ambiente no qual foram realizados os experimentos.

Os resultados apresentados neste capítulo foram obtidos por meio da execução dos experimentos descritos na seção 3.2, o consumo de bateria foi recuperado com emprego do

método descrito na subseção 3.1.2 e os resultados simulados foram obtidos nos ambientes descritos na seção 3.4.

Todos os experimentos foram executados cinco vezes e os resultados apresentados são a média aritmética dos valores recuperados. Além disso, os ensaios utilizaram o protocolo TCP, sendo que os valores de MSS negociados nas conexões, com Bluetooth e Wi-Fi, foram de 536 e 1.380 Bytes, para o Zire e para o LifeDrive, respectivamente. Tais valores são dependentes do sistema operacional de cada dispositivo, sendo possível concluir que, devido à fragmentação, o PDA de menor MSS utiliza mais processamento para transmitir a mesma quantidade de dados quando comparado ao de maior MSS. Por outro lado, em ambientes ruidosos, os quadros maiores têm maior probabilidade de serem corrompidos e, dessa forma, podem gerar sobrecarga devido à necessidade de retransmissão.

Quanto à avaliação do consumo de energia, os dados de consumo referem-se ao somatório do gasto de energia de todos os componentes do PDA (*e.g.* display LCD, processador e interface de rede sem fio), obtendo-se o comportamento real do sistema todo.

A seqüência do capítulo está estruturada da seguinte forma: as seções 4.1, 4.2 e 4.3 apresentam os resultados nas redes Bluetooth, Wi-Fi infra-estruturada e Wi-Fi *ad hoc*, respectivamente; a seção 4.4 apresenta uma comparação entre o comportamento dos dois PDAs; a seção 4.5 discute os resultados obtidos no ambiente de simulação; e, finalmente, a seção 4.6 apresenta as considerações finais do capítulo.

4.1 Rede Bluetooth

Durante o envio de dados na rede Bluetooth, o PDA Zire 72s apresentou um comportamento constante a dois e a doze metros do ponto de acesso, com taxas médias de 11,37 e 10,46 KB/s, respectivamente. Portanto, observa-se que ocorreu uma queda na taxa de aproximadamente 8 % no ponto mais distante, conforme apresenta a Figura 21 (a). Já durante

a recepção, o Zire apresentou um comportamento oscilante, especialmente a doze metros, com taxas médias de 21,05 e 13,53 KB/s a dois metros e doze metros, respectivamente, indicando uma queda na taxa de transmissão de aproximadamente 35 %. Além disso, observa-se que a variação na taxa é sensivelmente maior, em ambos os experimentos, no ponto mais distante. Apesar de a distância tornar o comportamento do PDA mais instável observa-se que a recepção foi mais prejudicada, nos dois casos. Isso provavelmente ocorreu porque o *laptop*, utilizado no ambiente experimental, tem capacidade de envio superior à capacidade de recepção do PDA.

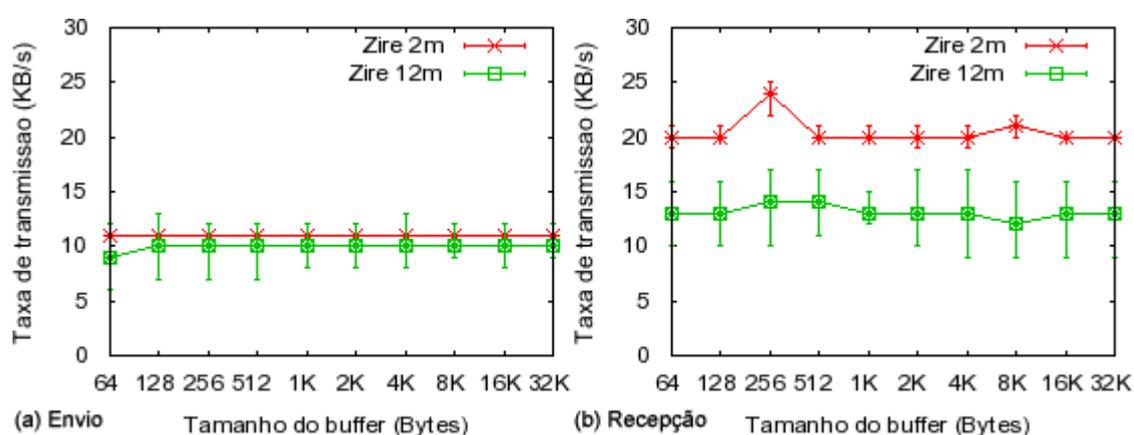


Figura 21. Taxas de transmissão do Zire 72s na rede Bluetooth.

Quanto ao consumo de bateria, o comportamento do Zire é crescente em função do aumento do tamanho do *buffer*. Tal comportamento ocorre tanto no envio quanto na recepção de dados, conforme apresenta a Figura 22. Durante o envio de dados, a potência média mantém-se próxima a 28 mW com *buffer* de 64 Bytes, e chega a atingir 35 mW com *buffer* de 32 KB. Já durante a recepção de dados, a potência dissipada é menor, mas o comportamento também é crescente, com potência média em torno de 19 mW com *buffer* de 64 Bytes, chegando a 23 mW com *buffer* de 32 KB. Por outro lado, as taxas de transmissão, apresentadas na Figura 21, são praticamente constantes durante o envio e recepção de dados, permitindo concluir que o Zire consome mais energia quando está transmitindo *buffers*

grandes, o que provavelmente ocorre devido à sobrecarga gerada pela utilização de TCP/IP *over* PPP em um enlace Bluetooth.

Dentre os resultados apresentados na Figura 22, é importante destacar que as potências médias observadas durante a recepção a doze metros são visivelmente menores que as utilizadas a dois metros. A primeira impressão é que o PDA apresenta um menor consumo de energia no ponto mais distante, contudo, o consumo total obtido nos experimentos mostra o contrário. Isso ocorre, porque as taxas de transmissão foram menores a doze metros e, dessa forma, o PDA levou mais tempo para realizar o experimento. Portanto, apesar de ter dissipado potências menores, apresentou um consumo maior.

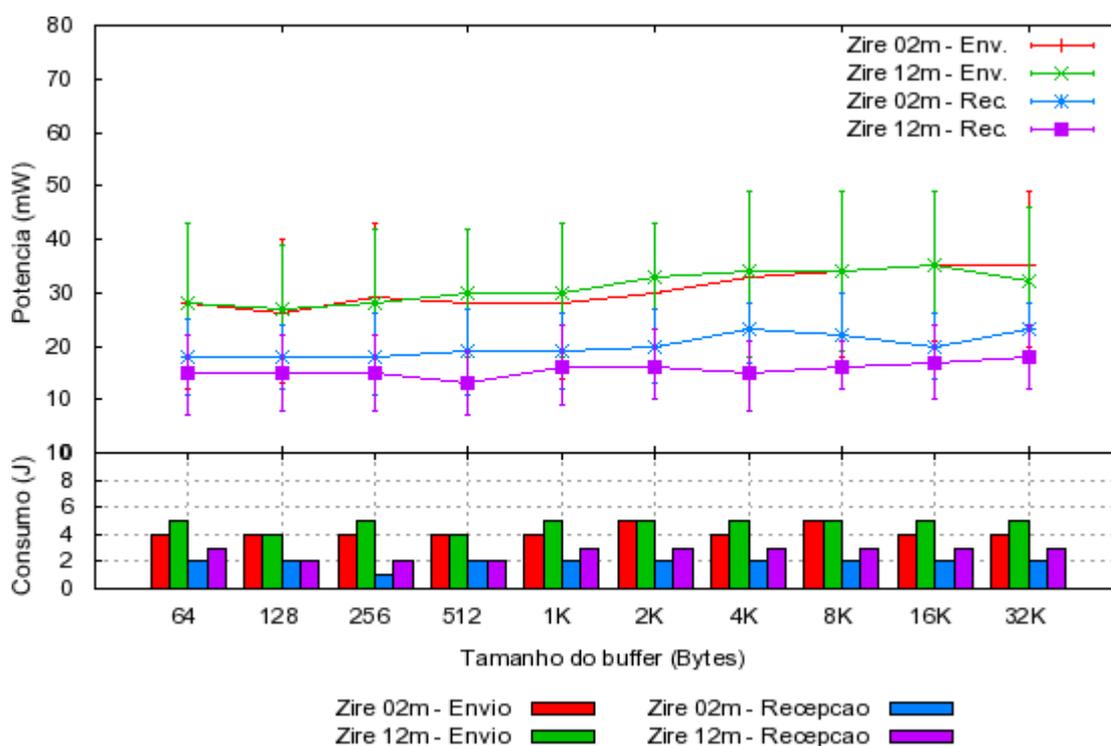


Figura 22. Consumo de bateria do Zire na rede Bluetooth.

Já o PDA LifeDrive apresentou um comportamento sensivelmente inferior no ponto mais distante, nos dois casos, conforme mostra os gráficos na Figura 23. Durante o envio de dados, o PDA obteve taxas médias de 55,89 e 21,27 KB/s a dois e a doze metros, respectivamente. Isso representa uma queda de aproximadamente 62 % na taxa de transmissão, conforme apresenta o gráfico na Figura 23 (a). Neste gráfico, também é possível

observar uma pequena queda na taxa com *buffers* maiores que 2K. Na recepção de dados, o comportamento é oscilante nos dois casos. A dois metros, com *buffer* de 64 Bytes, o PDA apresenta média de 44,78 KB/s e o comportamento é crescente até 52,61 KB/s com *buffer* de 4K; depois, o comportamento é decrescente até chegar em 48,29 KB/s com *buffer* de 32K. Por outro lado, a doze metros, o comportamento é oscilante até o *buffer* de 2K (10,32 KB/s) e, a partir daí, o comportamento passa a ser crescente, chegando a 32,34 KB/s com *buffer* de 32K, conforme apresenta a Figura 23 (b). Pode-se concluir que o PDA tem mais dificuldade para enviar *buffers* grandes, contudo, durante a recepção observa-se o inverso.

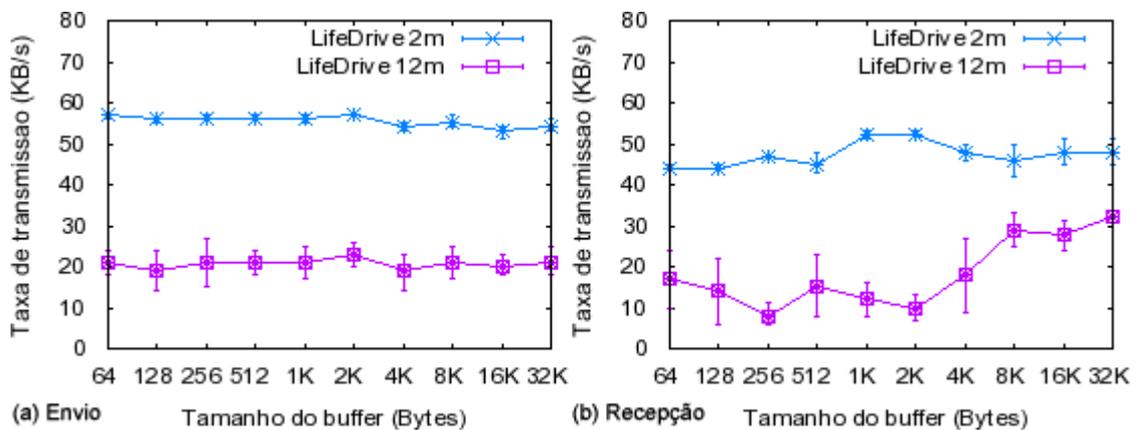


Figura 23. Taxas de transmissão do LifeDrive na rede Bluetooth.

Quanto ao consumo de bateria, o comportamento do LifeDrive, na rede Bluetooth, é apresentado na Figura 24. Nesta figura, observa-se que as maiores potências foram dissipadas no ponto mais próximo do ponto de acesso. As taxas de transmissão associadas a estes resultados, apresentadas na Figura 23, foram sensivelmente superiores no ponto mais próximo do ponto de acesso. Isso tem reflexo direto no consumo total em cada experimento, que, de modo geral, foi maior no ponto mais distante.

É importante observar que o comportamento do PDA é oscilante em todos os casos e não há uma relação trivial entre as taxas apresentadas (Figura 23) e as potências dissipadas (Figura 24). Por outro lado, é visível a relação entre as taxas e o consumo total de energia. Para ilustrar, tomemos a taxa de recepção no ponto mais distante e com *buffer* de 256 Bytes,

que foi o pior caso na recepção (8,65 KB/s), e conseqüentemente, foi o pior caso no consumo total de energia 8,02 J, para transmitir 3 MB.

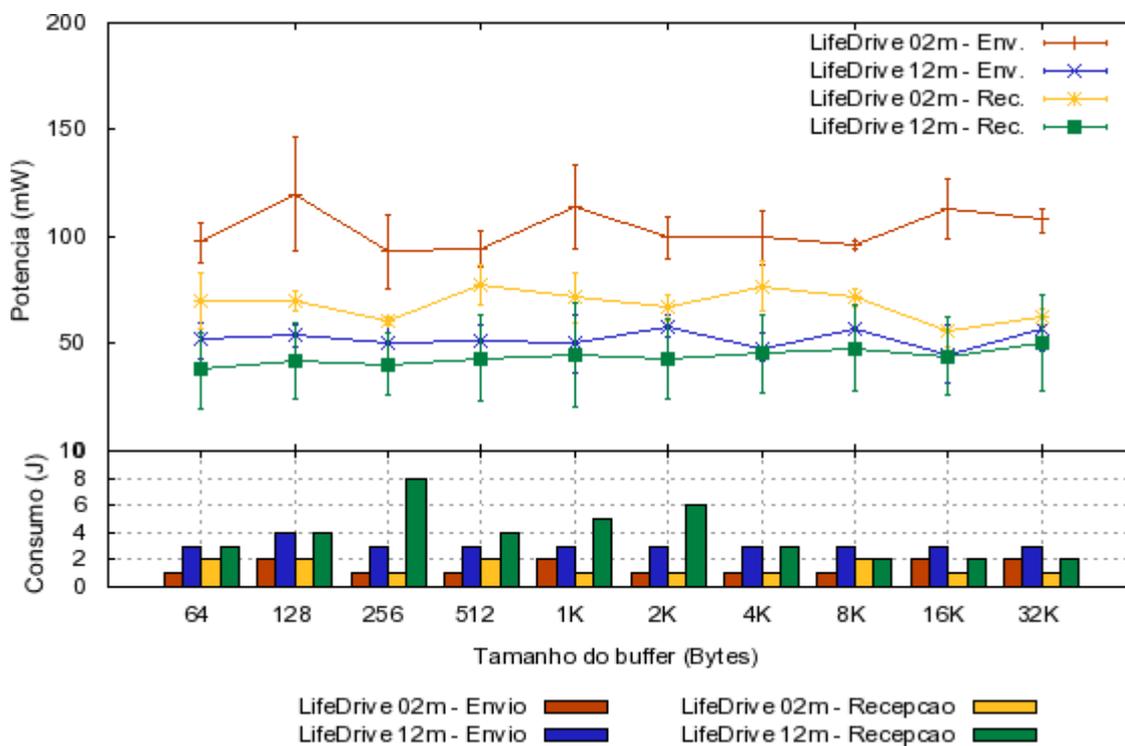


Figura 24. Consumo de bateria do LifeDrive na rede Bluetooth.

Em suma, os dois PDAs apresentaram taxas de transmissão inferiores no ponto mais distante. O PDA Zire apresentou maior diferença na recepção (queda de 35%), enquanto o LifeDrive obteve quedas mais severas nos dois casos, envio e recepção, aproximadamente 62 e 60%, respectivamente. Quanto ao consumo de bateria, é possível concluir que os dois dispositivos apresentaram maior consumo no ponto mais distante e isso aconteceu porque as menores taxas também foram observadas neste ponto. As potências médias dissipadas foram menores no ponto mais distante, provavelmente, porque o volume de dados manipulado foi menor, contudo o consumo foi maior ou igual em todos os casos.

4.2 Rede Wi-Fi Infra-estruturada

O PDA Zire apresentou taxas médias de envio próximas de 34 e 44 KB/s a dois e doze metros, respectivamente, o que representa um crescimento médio de quase 29 % da taxa, no

ponto mais distante, conforme mostra o gráfico na Figura 25 (a). Neste gráfico, observa-se que o PDA apresentou taxas de envio próximas de 23 KB/s, a dois metros, e 31 KB/s, a doze metros, com *buffer* de 64 Bytes, sendo que esta taxa cresceu conforme o tamanho do *buffer* aumentou até 512 Bytes, onde atingiu aproximadamente 37 KB/s a dois metros e 46 KB/s a doze metros, estabilizando para os tamanhos superiores. O mesmo não foi observado durante a recepção, conforme apresenta o gráfico na Figura 25 (b). Neste gráfico, observa-se que para o *buffer* de 64 Bytes a taxa estava próxima de 88 KB/s a dois metros e 91 KB/s a doze metros, diminuindo e oscilando até 69 e 98 KB/s a dois e doze metros, respectivamente, para *buffer* de 32 KB. Estes resultados indicam que o PDA teve um desempenho igual ou superior no ponto mais distante do AP, ou seja, a diferença de dez metros entre os dois pontos não foi significativa.

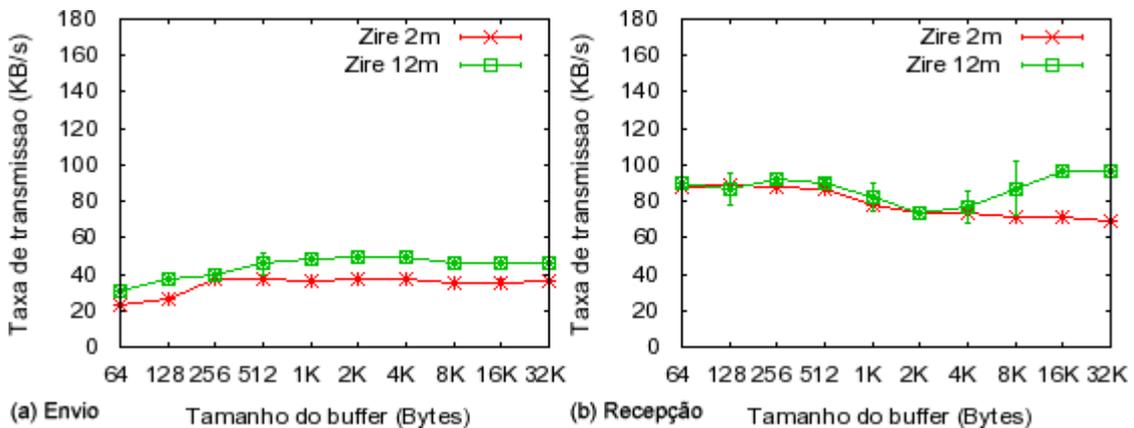


Figura 25. Taxa de transmissão do Zire 72s na rede Wi-Fi infra-estruturada.

Quanto ao consumo de bateria, o Zire dissipou potências maiores, durante o envio de dados, no ponto mais próximo do AP, conforme mostra a Figura 26. Nesta figura, observa-se que, a dois metros, as potências médias oscilaram na faixa de 439 a 468 mW e a doze metros na faixa de 283 a 320 mW. Tal comportamento influenciou o consumo total durante os experimentos de forma que o consumo a dois metros foi superior ao consumo a doze metros em todos os casos. Relembrando, que as taxas de transmissão associadas ao ponto mais próximo foram inferiores ou iguais às taxas apresentadas no ponto mais distante, como

apresentado na Figura 25. Observa-se, também, que o consumo é maior para os *buffers* menores; cerca de 9,5 J, para *buffer* de 64 Bytes, e aproximadamente 6,5 J, para *buffer* de 32 KB, especialmente durante o envio de dados. Provavelmente, isso ocorre devido à necessidade de um número maior de iterações para completar o experimento.

Na recepção de dados, o comportamento do Zire é similar nos dois pontos experimentados, conforme apresenta a Figura 26. Nesta figura, observa-se que, a dois metros do AP, as potências médias oscilaram na faixa de 322 a 420 mW e a doze metros na faixa de 317 a 409 mW. Dessa forma, o consumo total durante os experimentos também foi próximo. Ocorreram divergências no consumo com *buffers* de 8, 16 e 32 KB, de forma que, o consumo foi menor no ponto mais distante do AP.

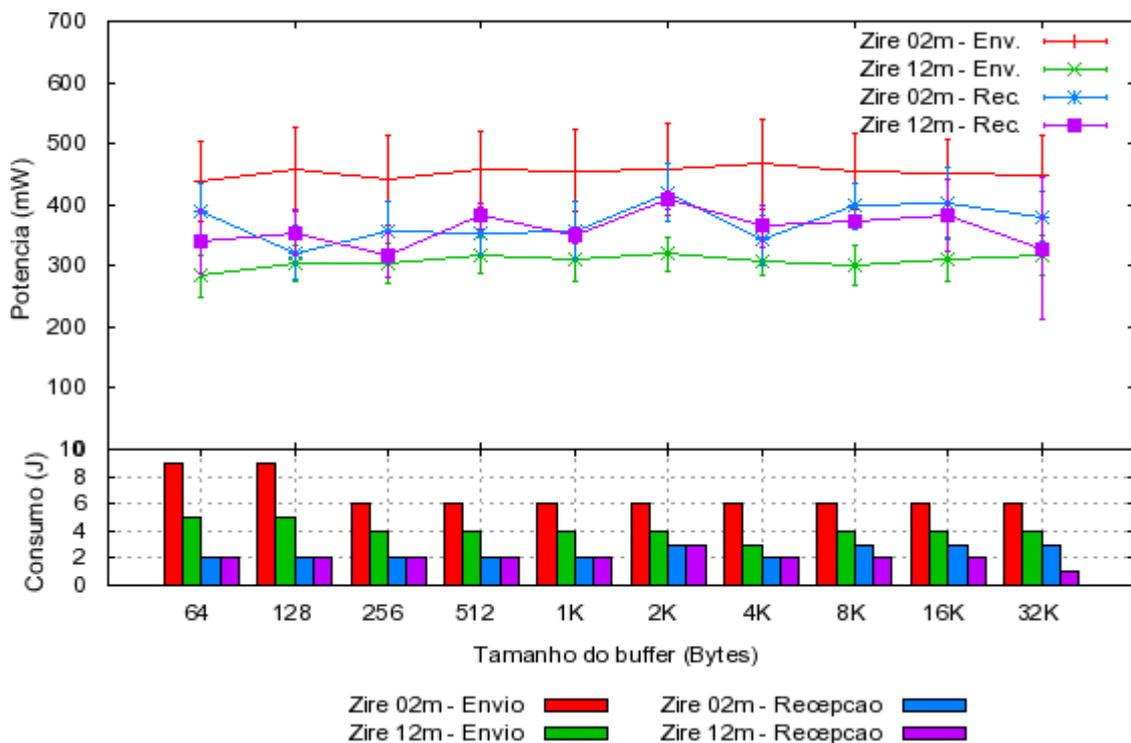


Figura 26. Consumo de bateria do Zire na rede Wi-Fi infra-estruturada.

O PDA LifeDrive, durante o envio de dados, apresentou taxas médias na faixa de 569 a 601 KB/s, a dois metros, e na faixa de 227 a 283 KB/s a doze metros. Isso representa uma queda na taxa de aproximadamente 55 % no ponto mais distante, conforme apresenta o gráfico na Figura 27 (a). Contudo, o mesmo não é observado durante a recepção, conforme

mostra o gráfico na Figura 27 (b). Neste gráfico, tem-se um comportamento praticamente crescente para todos os tamanhos de *buffer*, sendo que as taxas a doze metros oscilaram e são menores que as taxas a dois metros até o *buffer* de 4K; a partir daí, o comportamento é similar. De forma geral, o PDA obteve taxas médias na faixa de 184 a 301 KB/s a dois metros e na faixa de 151 a 301 KB/s a doze metros, indicando uma queda na taxa de transmissão de aproximadamente 15 %, a menor queda observada para este PDA.

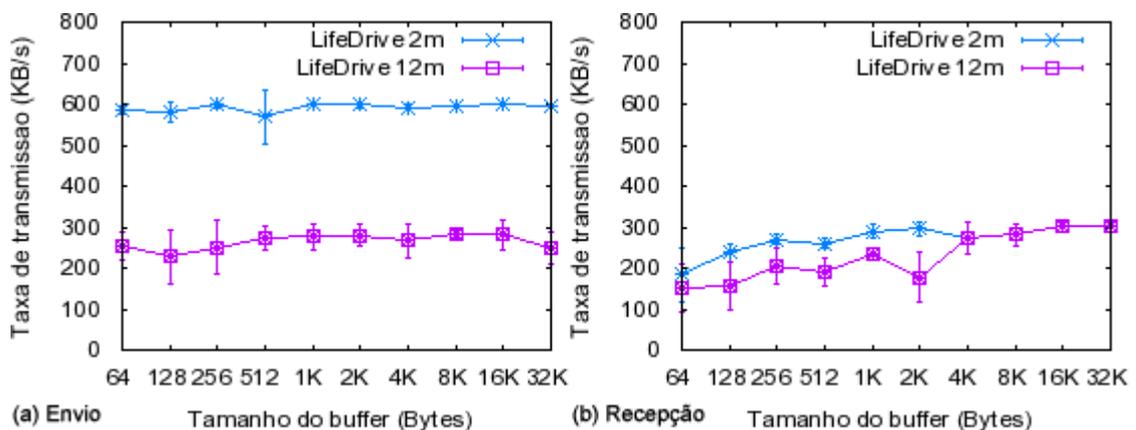


Figura 27. Taxa de transmissão do LifeDrive na rede Wi-Fi infra-estruturada.

Quanto ao consumo de bateria, o LifeDrive dissipou potências maiores no ponto mais distante do AP, durante o envio de dados. As potências médias variaram na faixa de 365 a 404 mW, a dois metros, e na faixa de 414 a 546 mW, a doze metros, conforme mostra a Figura 28. Esta diferença pode ser observada no consumo de energia total em cada experimento, onde todos os experimentos realizados no ponto mais próximo apresentaram um consumo inferior aos valores obtidos no ponto mais distante do AP. É importante lembrar que tal comportamento também foi influenciado pelas taxas de transmissão superiores no ponto mais próximo do AP, conforme mostra a Figura 27.

Já na recepção, o LifeDrive apresentou comportamento oscilante, conforme apresenta a Figura 28. Nesta figura, observa-se que as potências médias oscilam na faixa de 412 a 586 mW, a dois metros, e na faixa de 394 a 600 mW, a doze metros. As faixas de variação da potência apresentaram valores próximos, nos dois casos. Contudo, o consumo médio em cada

etapa do experimento é maior ou igual no ponto mais distante. Isso ocorreu porque as taxas de transmissão relacionadas a estas etapas foram superiores ou equivalentes no ponto mais próximo.

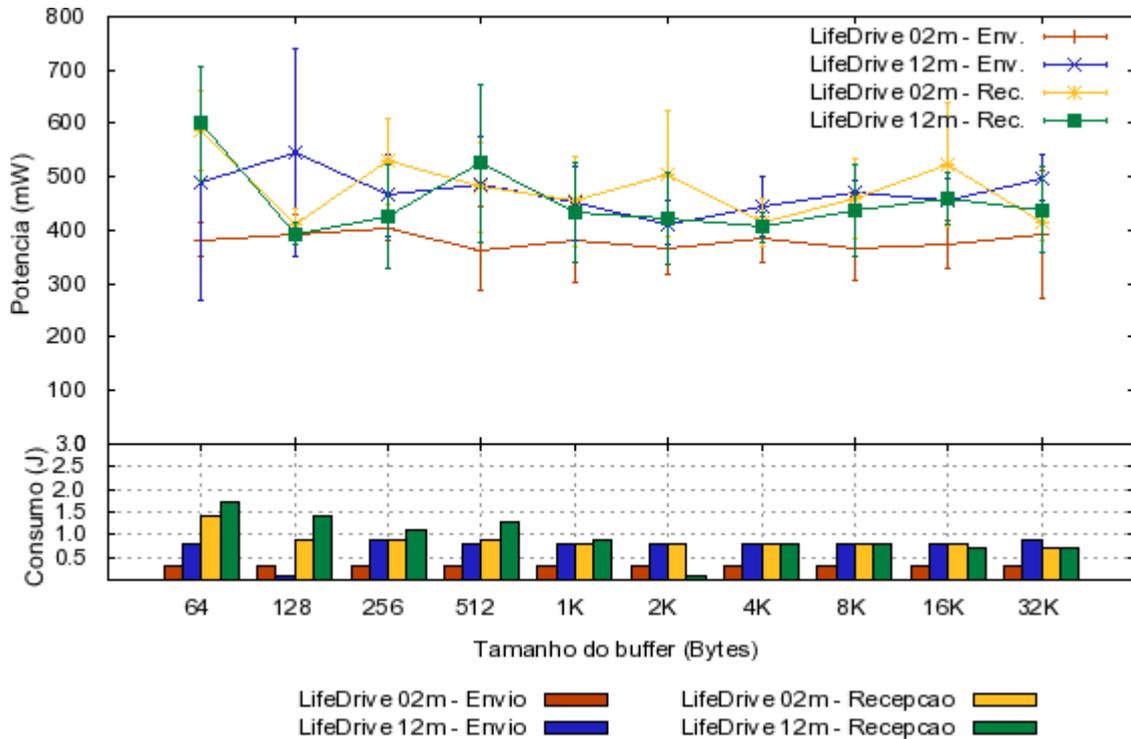


Figura 28. Consumo de bateria do LifeDrive na rede Wi-Fi infra-estruturada.

Em suma, os dois PDAs apresentaram comportamentos distintos na rede Wi-Fi infra-estruturada. O Zire obteve as maiores taxas e menores consumos no ponto mais distante do AP, enquanto o LifeDrive apresentou os melhores comportamentos no ponto mais próximo do AP. Esperava-se um comportamento similar nos dois pontos, uma vez que dez metros é uma diferença relativamente pequena para uma rede infra-estruturada. Portanto, é difícil justificar o comportamento levemente superior do Zire no ponto mais distante do AP e a queda brusca de desempenho do LifeDrive neste mesmo ponto.

4.3 Rede Wi-Fi Ad hoc

O comportamento do PDA Zire, na rede Wi-Fi *ad hoc*, foi bastante semelhante nos dois casos, como mostra os gráficos na Figura 29 (a) e (b), para envio e recepção,

respectivamente. Nesta figura, em (a), observa-se dois comportamentos distintos, crescentes para *buffers* de 64 Bytes até 1 K; a partir daí, as taxas são próximas e oscilantes. As taxas médias variaram na faixa de 75 a 126 KB/s, a dois metros, e na faixa de 76 a 128 KB/s, a doze metros. Estes resultados apontam um aumento de cerca de 1 % nas taxas de transmissão recuperadas no ponto mais distante. Já durante a recepção de dados, Figura 29 (b), observa-se um comportamento oscilante e levemente decrescente em função do crescimento do *buffer*. Neste caso, as taxas médias variaram na faixa de 92 a 110 KB/s a dois metros e na faixa de 96 a 114 KB/s, indicando um aumento insignificante na taxa de aproximadamente 0,5 %. Assim como na rede infra-estruturada, o Zire apresentou taxas inferiores, durante o envio de dados, para os *buffers* menores que 1 KB. Isso provavelmente ocorre devido à necessidade de um número maior de iterações para completar o experimento. Por outro lado, durante a recepção, as melhores taxas foram obtidas com os menores *buffers*, mas a diferença não é tão expressiva quanto no envio.

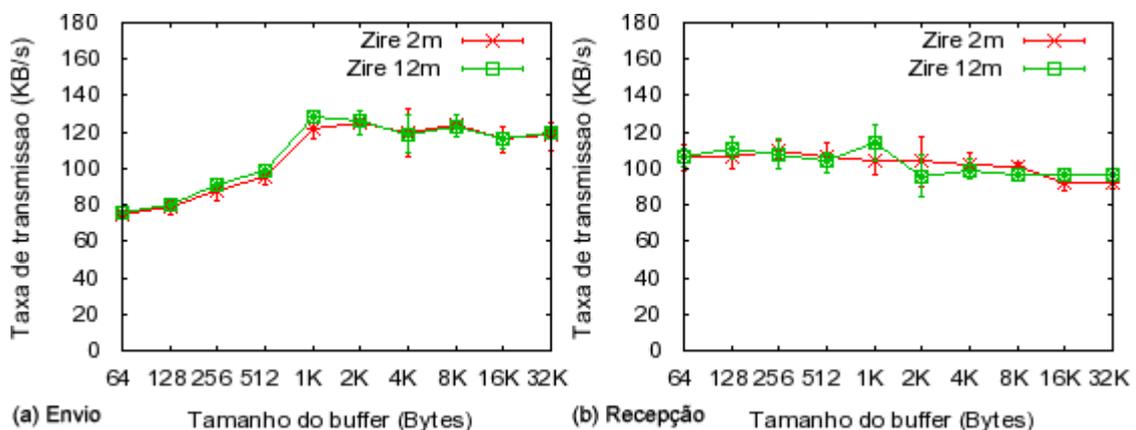


Figura 29. Taxa de transmissão do Zire 72s na rede Wi-Fi *ad hoc*.

Quanto ao consumo de bateria, o Zire dissipou as maiores potências durante o envio de dados no ponto mais próximo, assim como na rede infra-estruturada. Nos outros casos, tem-se um comportamento bastante próximo, conforme apresenta a Figura 30. Nesta figura, observa-se que as potências médias, durante o envio de dados, variaram na faixa de 375 a 418 mW, a dois metros, e na faixa de 296 a 330 mW, a doze metros. Esta diferença influenciou o

consumo total de energia, em cada etapa do experimento, de forma que os maiores consumos ocorreram no ponto mais próximo. Além disso, os maiores consumos ocorreram com os *buffers* menores, sendo que as potências dissipadas são próximas, mas as taxas de transmissão associadas apresentaram significativa variação. Dessa forma, observa-se um consumo médio de 2,8, 2,7 e 2,5 J para *buffers* pequenos, de 64, 128 e 256 Bytes, respectivamente, e consumo médio de 1,9 e 1,8 J para *buffers* grandes, de 16 e 32 KB, respectivamente.

Durante a recepção, o PDA dissipou potências médias na faixa de 245 a 275 mW, no ponto mais próximo, e na faixa de 275 a 338 mW, no ponto mais distante, conforme apresenta a Figura 30. Nesta figura, observa-se que as potências oscilaram com um comportamento levemente crescente conforme o tamanho do *buffer* aumenta. Dessa forma, o consumo total de energia apresenta o mesmo comportamento oscilante, mas com pequenas diferenças.

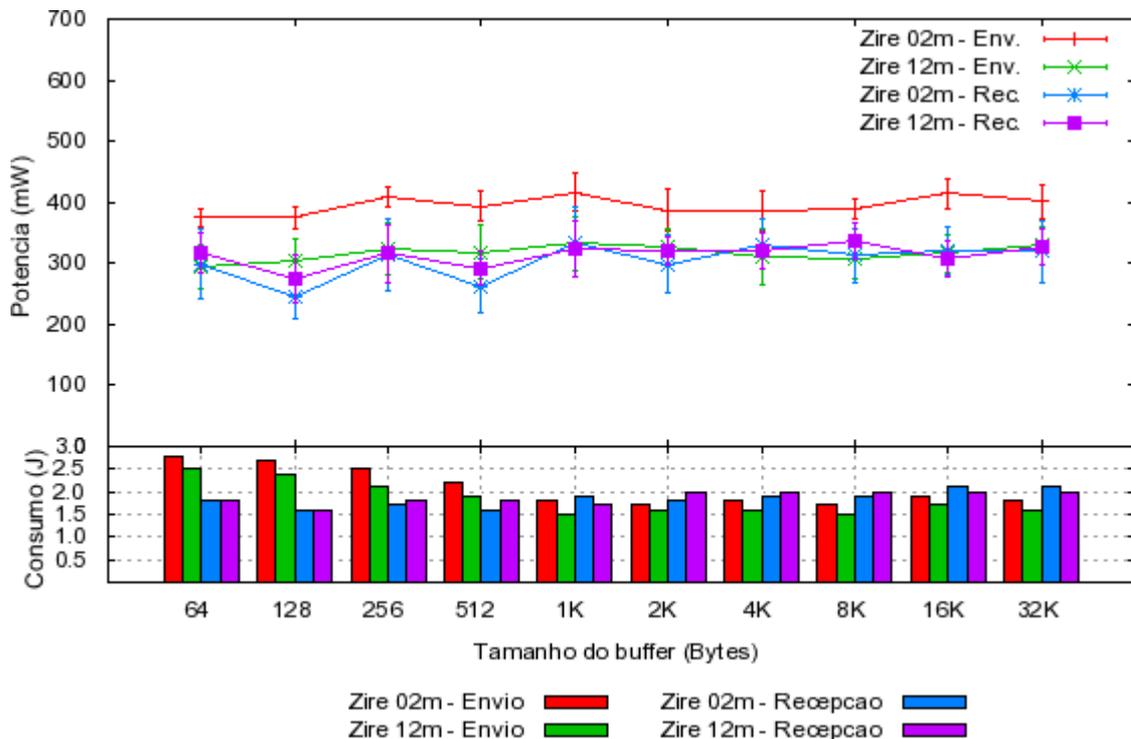


Figura 30. Consumo de bateria do Zire na rede Wi-Fi *ad hoc*.

Quanto às taxas de transmissão, o PDA LifeDrive, apresentou um comportamento oscilante, no envio, e levemente crescente, na recepção, conforme mostram os gráficos na Figura 31 (a) e (b), envio e recepção, respectivamente. Nesta figura, em (a), tem-se as taxas

médias durante o envio de dados que variaram na faixa de 590 a 669 KB/s, no ponto mais próximo, e na faixa de 457 a 584 KB/s, no ponto mais distante. Esta diferença representa uma queda de aproximadamente 16 % na taxa no ponto mais distante. Em (b), tem-se as taxas médias durante a recepção, cujos valores variaram na faixa de 222 a 301 KB/s, a dois metros, e na faixa de 130 a 200 KB/s, a doze metros. Tal diferença aponta uma queda de quase 39 % nas taxas médias recuperadas no ponto mais distante.

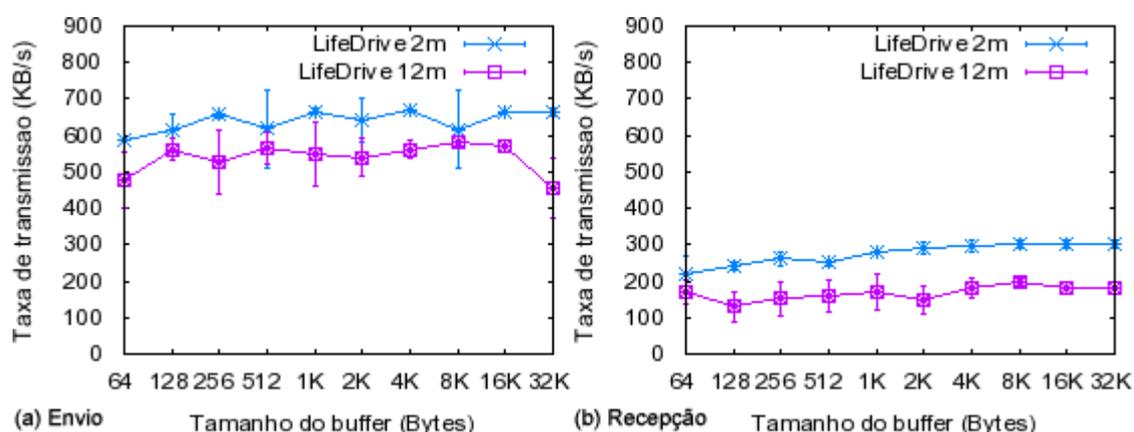


Figura 31. Taxa de transmissão do LifeDrive na rede Wi-Fi ad hoc.

Quanto ao consumo de bateria, o LifeDrive apresentou um comportamento oscilante em todos os casos, sendo que na recepção as potências médias variaram na faixa de 361 a 410 mW, a dois metros, e na faixa de 390 a 570 mW, a doze metros, conforme apresenta a Figura 32. Este comportamento, associado às taxas de transmissão inferiores no ponto mais distante, resultaram em consumos médios ligeiramente superiores neste ponto; médias de 0,3 e 0,4 J, a dois e doze metros, respectivamente. Já durante a recepção, as potências médias variaram na faixa de 360 a 455 mW, a dois metros, e na faixa de 365 a 550 mW, a doze metros. Este comportamento é bastante próximo ao observado durante o envio de dados, contudo as taxas de transmissão foram significativamente inferiores. Isso determinou um consumo de bateria maior durante a recepção de dados; média de 0,8 e 1,3 J, a dois e doze metros, respectivamente.

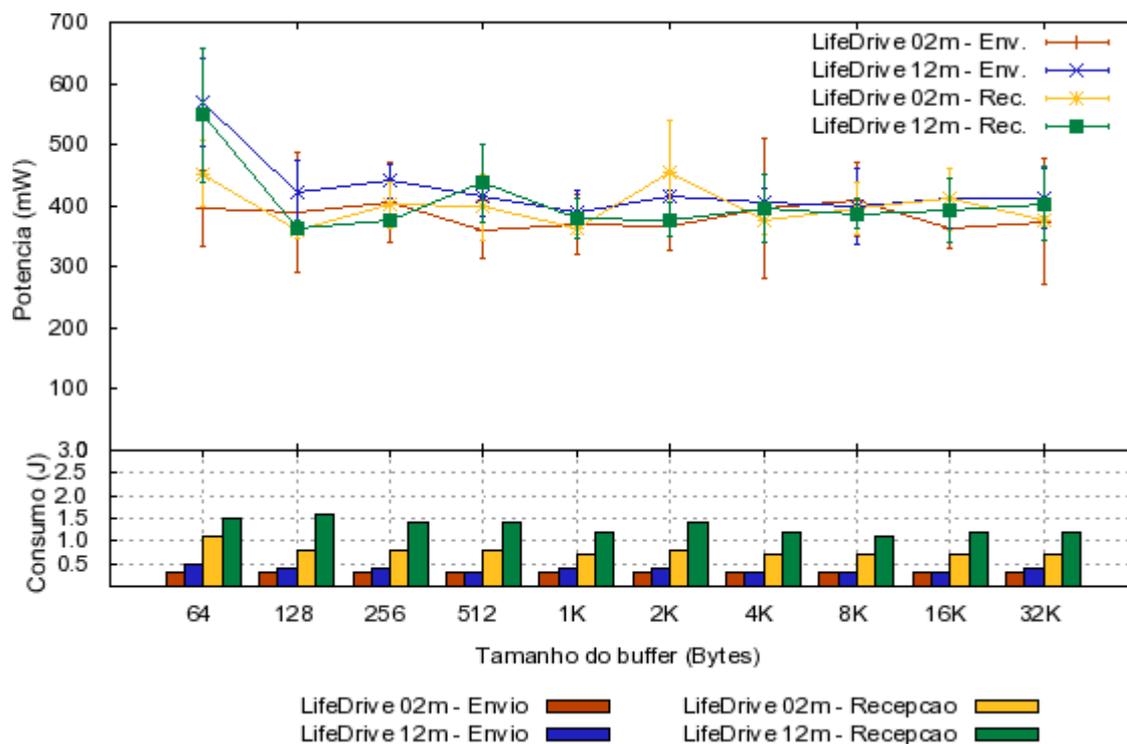


Figura 32. Consumo de bateria do LifeDrive na rede Wi-Fi *ad hoc*.

Em suma, o comportamento do Zire, durante o envio de dados, com *buffers* menores que 1 KB, obteve um desempenho inferior se comparado às taxas alcançadas com *buffers* maiores que 1 KB. Provavelmente isso ocorre devido ao número de *loops* necessário para transmitir os 3 MBytes de dados. Contudo, durante a recepção o efeito é contrário, a taxa de transmissão diminui conforme o tamanho do *buffer* aumenta. Isso provavelmente ocorre porque para o *laptop* é mais fácil transmitir *buffers* grandes e o PDA não consegue remontar os pacotes no mesmo ritmo. Já o LifeDrive não apresenta o mesmo comportamento pois tem capacidade de processamento superior à do Zire. Além disso, observa-se que as taxas obtidas no envio são significativamente maiores que as obtidas na recepção e, por consequência, o mesmo comportamento é observado no consumo de energia.

4.4 Comparação do Desempenho dos Dois PDAs

As seções 4.1, 4.2 e 4.3 apresentaram os resultados dos experimentos nas redes Bluetooth, Wi-Fi Infra-estrutura e Wi-Fi *ad hoc*, respectivamente. A análise foi feita por meio

da comparação entre os resultados obtidos, pelos PDAs, no ponto mais próximo e mais distante, destacando eventuais influências que o tamanho do *buffer* gerou na taxa de transmissão e no consumo de energia. Contudo, em nenhum momento o comportamento dos dois PDAs foi confrontado. Uma vez que este trabalho partiu da hipótese que as características de *hardware* do PDA influenciam seu comportamento durante a comunicação, faz-se necessária a comparação entre os dois equipamentos para tentar ratificar a hipótese. Neste caso, é importante lembrar que o LifeDrive tem um processador 25% mais rápido que o Zire 72s e, além disso, tem um *display* maior e um HD de 4 GB.

Dada a quantidade de resultados, optou-se por apresentar os resultados recuperados por meio da média aritmética dos valores obtidos em cada ambiente de rede, durante o envio e a recepção de dados, a dois e doze metros do ponto de acesso. Tais resultados apresentaram grandes variações em torno da média, mas são suficientes para uma comparação entre os dois equipamentos. A comparação será realizada por meio da análise das taxas de transmissão, das potências dissipadas, do consumo de energia e, por fim, da eficiência na comunicação.

Quanto às taxas de transmissão, o LifeDrive obteve valores médios superiores aos do Zire em todos os casos, conforme mostra o gráfico na Figura 33 (a). Neste gráfico, observa-se que o Zire apresentou taxas médias em torno de 14, 61 e 105 KB/s, e o LifeDrive, apresentou taxas médias em torno de 36, 338 e 405 KB/s, nas redes Bluetooth, Wi-Fi infra-estruturada e Wi-Fi *ad hoc*, respectivamente. A menor diferença foi na rede Bluetooth, onde o LifeDrive obteve taxas cerca de 2,5 vezes superiores as do Zire. Esta proximidade indica que o uso de TCP em um túnel PPP é custoso para os dois dispositivos. Já nas redes Wi-Fi, as diferenças são maiores, sendo que a superioridade do LifeDrive foi em torno de 5,5 e 3,8 vezes, nas redes infra-estruturada e *ad hoc*, respectivamente. Aparentemente, esta diferença ocorreu devido à capacidade de processamento superior do LifeDrive que, com isso, consegue manipular um volume de dados maior durante a comunicação.

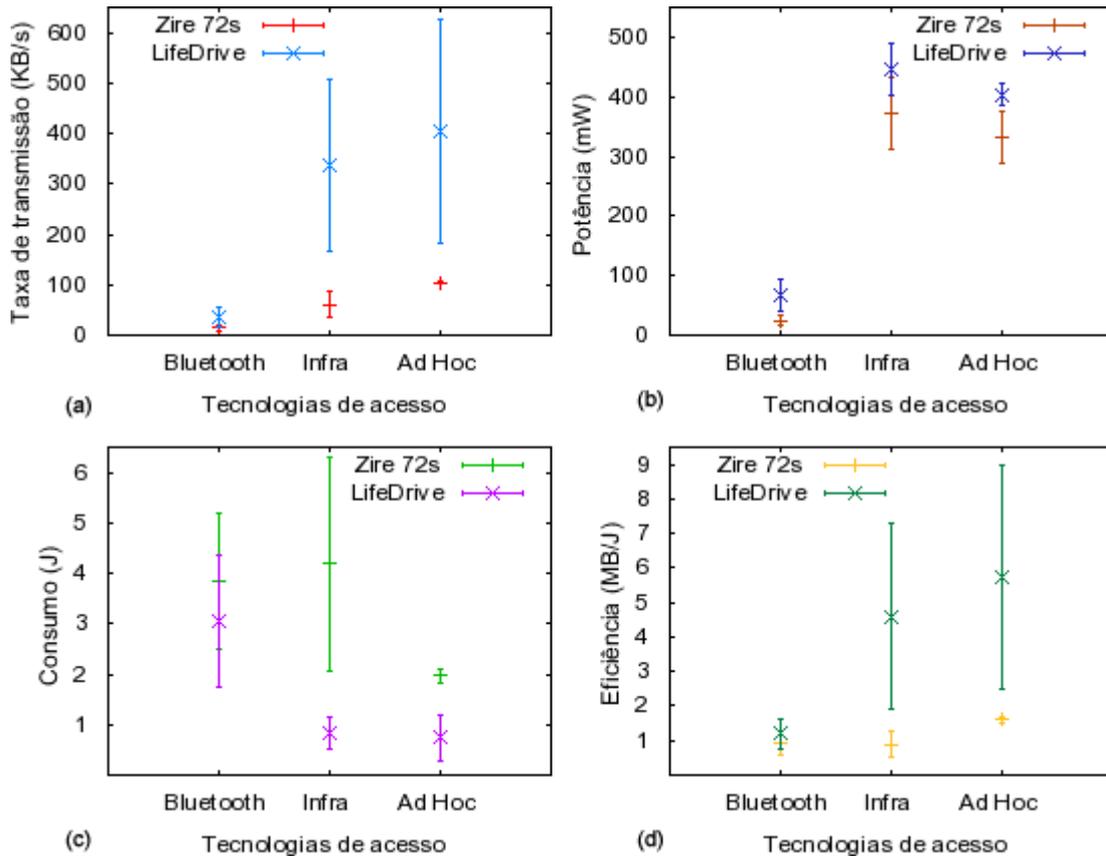


Figura 33. Comparação entre os dois PDAs nas três redes.

Ao contrário das taxas de transmissão, as potências médias dissipadas por cada PDA foram relativamente próximas, contudo o LifeDrive obteve maiores valores que o Zire nas três redes, conforme apresenta o gráfico na Figura 33 (b). Neste gráfico, observa-se que as potências médias apresentadas pelo Zire foram próximas de 25, 374 e 333 mW e as dissipadas pelo LifeDrive foram 67, 447 e 404 mW, nas redes Bluetooth, Wi-Fi infra-estruturada e Wi-Fi *ad hoc*, respectivamente. A maior diferença foi observada na rede Bluetooth, onde o LifeDrive dissipou, em média, uma potência cerca de 2,7 vezes maior que o Zire. Já nas redes Wi-Fi, as diferenças são de 19 e 21 %, nas redes infra-estruturadas e *ad hoc*, respectivamente. Este comportamento era esperado, uma vez que o LifeDrive tem uma tela de LCD maior, um processador mais rápido e um HD (o Zire não tem HD). Contudo, a diferença na rede Bluetooth foi bastante grande e sem uma justificativa trivial.

Quanto ao consumo médio de energia, o Zire apresentou valores superiores ao consumo apresentado pelo LifeDrive, em todos os casos, como pode ser visto no gráfico da Figura 33 (c). Neste gráfico, observa-se que o Zire consumiu cerca de 3,8, 4 e 2 J, e o LifeDrive consumiu aproximadamente 3, 0,8 e 0,7 J, nas redes Bluetooth, Wi-Fi infra-estruturada e Wi-Fi *ad hoc*, respectivamente. As maiores diferenças foram observadas nas duas redes Wi-Fi, onde o Zire consumiu cerca de 5 e 2,6 vezes mais energia que o LifeDrive, nas redes infra-estruturada e *ad hoc*, respectivamente. Já na rede Bluetooth, a diferença foi de apenas 26 %. É importante lembrar que as potências dissipadas pelos dois PDAs nas duas redes Wi-Fi foram próximas, contudo as taxas de transmissão foram significativamente distintas. Pode-se concluir que o maior consumo apresentado pelo Zire ocorreu devido às vazões de dados alcançadas, que, provavelmente estão fortemente relacionadas à capacidade de processamento do dispositivo.

Em suma, o LifeDrive foi mais eficiente que o Zire em todos os casos, conforme apresenta o gráfico na Figura 33 (d). Neste gráfico, observa-se que o Zire apresentou uma eficiência média de 0,9, 0,8 e 1,6 MB/J e o LifeDrive apresentou eficiência média de 1,1, 4,6 e 5,7 MB/J, nas redes Bluetooth, Wi-Fi infra-estruturada e Wi-Fi *ad hoc*, respectivamente. As maiores diferenças ocorreram nas redes Wi-Fi, onde o LifeDrive foi cerca de 5,7 e 3,5 vezes mais eficiente que o Zire. Já na rede Bluetooth a diferença foi de apenas 22 %.

Estes resultados mostram que os comportamentos dos dois PDAs experimentados foram distintos, ratificando a hipótese de que as características de *hardware* influenciam a capacidade de comunicação e o consumo de energia. Provavelmente, a capacidade de processamento fez a diferença nestes experimentos, pois observa-se que as potências dissipadas foram próximas, especialmente nas redes Wi-Fi; diferença de 19 e 21 %, redes infra-estruturada e *ad hoc*, respectivamente. A próxima seção apresenta os resultados das simulações e realiza uma comparação com os resultados experimentais discutidos até aqui.

4.5 Resultados da Simulação

As simulações foram realizadas com dois modelos de propagação *TwoRayGround* e *Shadowing*. O primeiro foi utilizado para simular um ambiente *outdoor*, sem obstáculos e sem linha de visada entre os nós. O segundo modelo foi utilizado para simular um ambiente *indoor* onde estão presentes diversos obstáculos que podem acelerar a atenuação do sinal. O ambiente de rede é composto por dois nós, sendo que, no instante inicial, um dos nós está estacionado a dois metros do ponto de acesso e a cada trinta segundos se movimenta para longe; cinco metros para Bluetooth e dez metros para Wi-Fi. Antes de cada movimento foi computada a taxa de transmissão alcançada no ponto em questão.

A Figura 34 apresenta os resultados da simulação na rede Bluetooth. Nesta figura, observa-se que nos dois casos, *indoor* e *outdoor*, a taxa de transmissão decresce conforme a distância aumenta, mas a variação é mais acentuada no ambiente *indoor*. Neste ambiente, as taxas médias foram 11, 9 e 3 KB/s nos pontos localizados a 2, 6 e 12 metros do ponto de acesso, respectivamente. Dessa forma, ocorreu um decrescimento de aproximadamente 22 %, entre 2 e 6 metros, e de 72 %, entre os pontos a 6 e 12 metros. Já no ambiente *outdoor*, observou-se taxas médias de 11 KB/s, a dois metros, e de 10 KB/s, a seis e doze metros do ponto de acesso. Dessa forma, observa-se um decrescimento de 10 % na taxa entre os pontos localizados a dois e seis metros, por outro lado, entre seis e doze metros a taxa foi constante. Portanto, como era esperado, o ambiente *outdoor*, permite manter a taxa de transmissão por uma área maior.

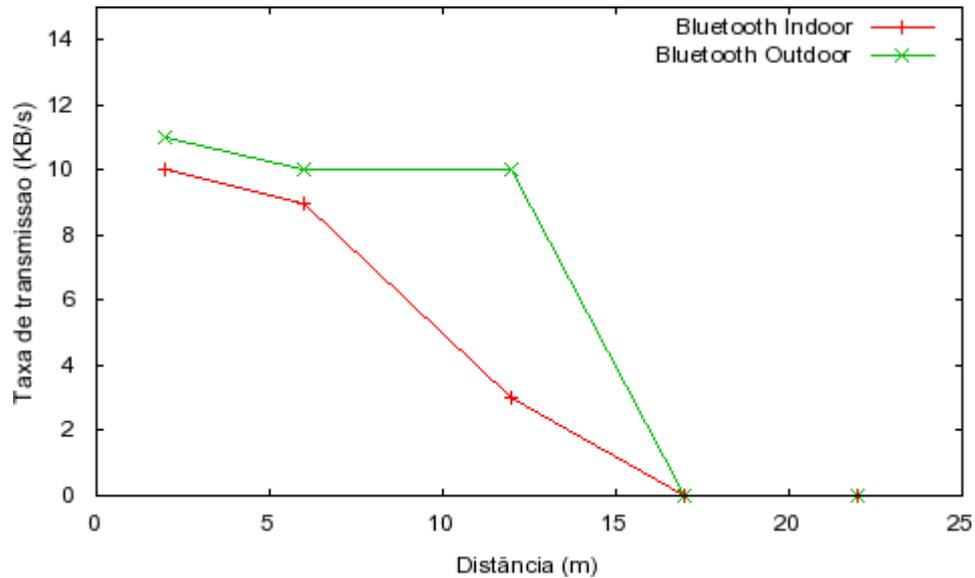


Figura 34. Resultados da simulação com Bluetooth.

No ambiente experimental, o Zire apresentou as taxas médias 11,56 e 10,96 KB/s, no melhor caso, a dois e doze metros, respectivamente, conforme apresenta o gráfico na Figura 35 (a). Estes resultados indicam uma queda na taxa de aproximadamente 5 % e, dessa forma, apresentou um comportamento mais próximo ao observado no ambiente de simulação *outdoor*. Já no pior caso as taxas foram 11,13 e 9,8 KB/s a dois e doze metros, respectivamente, apresentando uma queda de aproximadamente 13 % na taxa de transmissão. Pode-se concluir que o comportamento do Zire foi mais próximo do comportamento observado na simulação *outdoor*, nos dois casos. Por outro lado, o LifeDrive apresentou taxas médias de 57,37 e 23,61 KB/s, no melhor caso, a dois e doze metros respectivamente, apresentando uma queda 58 % na taxa. Já no pior caso, o PDA apresentou taxas médias de 53,39 e 9,8 KB/s a dois e a doze metros respectivamente, apresentando uma queda de 81 % na taxa. Estes resultados apontam um provável problema na antena da interface de rede que está embutida na carcaça do dispositivo.

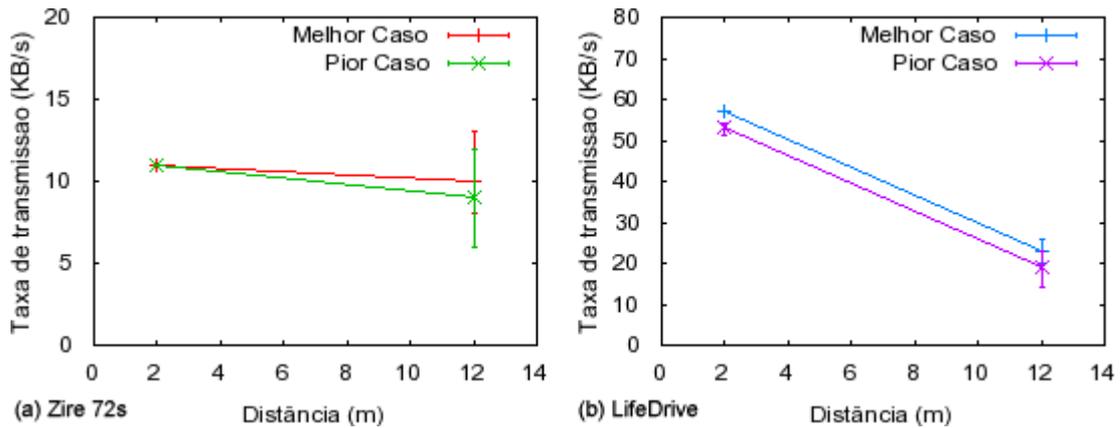


Figura 35. Resultados experimentais com Bluetooth.

Os resultados observados nas simulações na rede Wi-Fi são apresentados na Figura 36. Nesta figura, observa-se que o comportamento no ambiente *indoor* é decrescente em todos os casos. Por outro lado, no ambiente *outdoor*, observa-se uma pequena ascensão seguida de um comportamento constante. No ambiente *indoor*, as taxas médias foram 214, 203, 64 e 3 KB/s, a 2, 12, 22 e 32 metros do ponto de acesso, respectivamente. Tais resultados caracterizam um comportamento decrescente em função da distância, com uma queda na taxa de 5 %, entre os pontos a 2 e 12 metros, de 68 %, entre os pontos a 12 e 22 metros e, por fim, queda de 95 %, entre o ponto a 22 e 32 metros.

Já no ambiente *outdoor*, observou-se taxas médias próximas de 214 KB/s, a dois metros, próximas de 215 KB/s, a 12, 22, 32 e 42 metros, e cerca de 6 KB/s, a 52 metros. Estes resultados apontam um aumento de aproximadamente 4% na taxa entre os pontos localizados a 2 e 12 metros, já nos pontos 12, 22, 32 e 42 a taxa permaneceu praticamente constante e, por fim, entre os pontos 42 e 52 ocorreu uma queda de quase 97 % na taxa de transmissão. Assim como nas simulações com Bluetooth, pode-se concluir que o ambiente *outdoor* mantém o comportamento ótimo em uma área maior que no ambiente *indoor*.

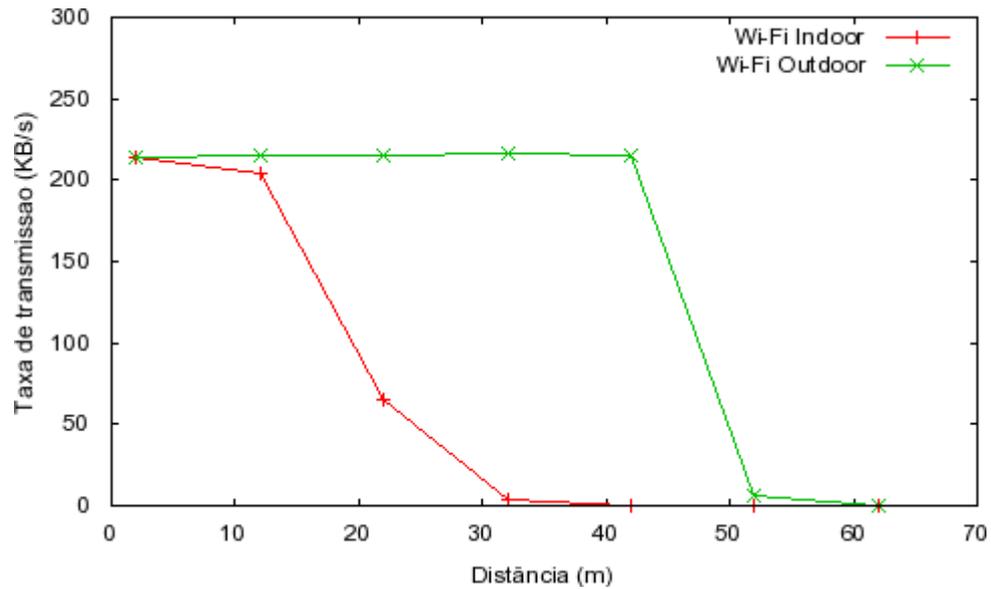


Figura 36. Resultados da simulação na rede Wi-Fi.

Para efeito de comparação, a Figura 37 apresenta os melhores e piores casos (MC e PC, respectivamente) nas redes Wi-Fi infra-estruturada e Wi-Fi *ad hoc*, nos dois pontos. Na rede *ad hoc*, o Zire apresentou taxas médias próximas de 126 e 128 KB/s, a dois e doze metros, respectivamente, no melhor caso. Já no pior caso, observou-se taxas médias de 75 e 76 KB/s, a dois e doze metros, respectivamente. Estes resultados indicam um aumento na taxa de aproximadamente 2 e 1 %, no melhor e pior caso, respectivamente. Pode-se concluir que o PDA teve um comportamento bastante similar aos resultados obtidos na simulação *outdoor*, apesar de os experimentos terem sido realizados em um ambiente *indoor*.

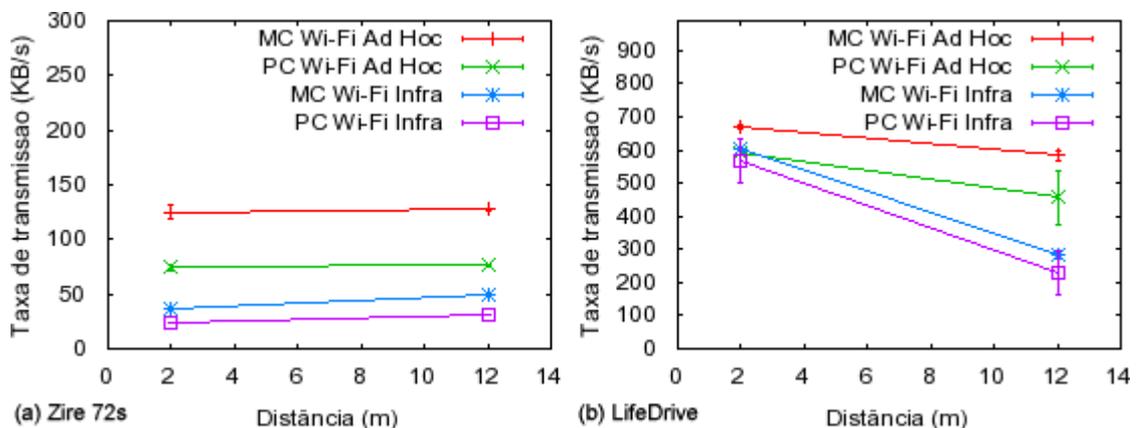


Figura 37. Resultados experimentais nas redes Wi-Fi.

Na rede infra-estruturada, o Zire obteve taxas inferiores se comparadas à rede *ad hoc*, conforme apresenta o gráfico na Figura 37 (a). Neste gráfico, observa-se o PDA apresentou taxas médias em torno de 37 e 50 KB/s, a dois e doze metros, e 23 e 32 KB/s, a dois e doze metros, no melhor e pior caso, respectivamente. Dessa forma, ocorreu um crescimento na taxa de cerca de 32 e 36 %, no melhor e pior caso, na devida ordem. Assim como na rede *ad hoc*, este comportamento foi mais próximo do comportamento observado no ambiente *outdoor*, contudo neste caso o crescimento na taxa entre o ponto a dois e doze metros foi expressivamente maior.

Não é possível justificar este comportamento com a granularidade e o teor dos experimentos realizados. Contudo, pode ter ocorrido algum fenômeno no ponto mais próximo que resultou nesta diferença sutil; é importante destacar que as taxas observadas foram muito menores que a vazão nominal da rede, então este comportamento também pode estar relacionado a outras sobrecargas (*e.g.* do protocolo TCP). O mesmo comportamento não foi observado com o outro PDA.

Assim como o Zire, o LifeDrive obteve os melhores desempenhos na rede *ad hoc*, conforme mostra o gráfico na Figura 37 (b). Neste gráfico, observa-se que as taxas obtidas pelo PDA, na rede *ad hoc*, foram próximas de 669 e 584 KB/s, a dois e doze metros, e de 590 e 457 KB/s, a dois e doze metros, no melhor e pior caso, respectivamente. Estes resultados apontam um comportamento decrescente no ponto mais distante, com diferença de aproximadamente 14 e 29 %, no melhor e pior caso, respectivamente. Dessa forma, observa-se uma queda na taxa de transmissão, no ponto mais distante, significativamente maior que a observada no ambiente de simulação *indoor*. Já na rede infra-estruturada, o LifeDrive apresentou taxas médias próximas de 601 e 283 KB/s, a dois e doze metros, e 569 e 227 KB/s, a dois e doze metros, no melhor e pior caso, na devida ordem. Dessa forma, observa-se, no

ponto mais distante, uma queda na taxa de cerca de 2 e 2,5 vezes, no melhor e pior caso, na devida ordem. Esta diferença não era esperada, especialmente na rede infra-estruturada onde a diferença de dez metros pode ser considerada pequena.

As interfaces de rede dos dois PDAs são idênticas, a única diferença é que a interface Wi-Fi do Zire é externa e o LifeDrive tem as duas interfaces, Bluetooth e Wi-Fi embutidas, conforme apresentou a Figura 18; o Zire 72s, à esquerda, e o LifeDrive, à direita. Pode-se concluir que provavelmente a acomodação das duas interfaces na carcaça do LifeDrive tenha influenciado seu comportamento no ponto mais distante.

4.6 Considerações Finais

Neste capítulo, em um primeiro momento, foi apresentada uma análise da taxa de transmissão e do consumo de bateria obtida com os dois PDAs nos três ambientes de rede; Bluetooth, Wi-Fi infra-estruturada e Wi-Fi *ad hoc*. Esta análise foi feita por meio de comparações entre os resultados obtidos no ponto localizado a dois metros e no ponto a doze metros. Em um segundo momento, foi realizada uma comparação entre o comportamento dos dois PDAs. Observou-se que o comportamento dos dois dispositivos foram distintos e que o PDA com maior capacidade de processamento foi mais eficiente em todos os cenários. Em seguida foram apresentados os resultados obtidos nos ambientes de simulação e realizada uma comparação com os resultados obtidos na experimentação.

5 Sistema de Suporte à Decisão

Vários fatores técnicos dificultam a implementação de um mecanismo de seleção automática de acesso, no nível da aplicação, de forma eficiente e conveniente. Dentre tais fatores merecem destaque as limitações do sistema operacional e a descoberta de pontos de acesso. Os PDAs experimentados neste trabalho utilizam o sistema operacional Palm OS. Este SO não permite a criação de *threads* no espaço de usuário, dessa forma, tem-se um fluxo de execução único para a aplicação. Neste ambiente, seria necessário parar a execução da aplicação para executar o mecanismo de seleção de acesso. Outro problema é a descoberta dos pontos de acesso, uma vez que em ambientes reais podem existir várias redes ou nenhuma. O dispositivo deve estar ciente da existência da rede antes de tentar juntar-se a ela, isso pode resultar em uma tarefa bastante custosa, desperdiçando recursos preciosos.

Neste contexto, observa-se que, hoje, a seleção do PDA e da tecnologia de acesso deve ser realizada na fase de projeto da aplicação. Dessa forma, este trabalho propõe um SSD para auxiliar esta escolha. Segundo Sprague *et al.* (1982), um SSD auxilia seus usuários à utilizar dados e modelos para solucionar problemas. Neste sentido, os requisitos funcionais do sistema são: (i) Dado um cenário, recuperar os PDAs e tecnologias de comunicação que atendem ao cenário; (ii) Dado um PDA e um cenário, listar as tecnologias de comunicação, suportadas pelo dispositivo, que atendam ao cenário; (iii) Dado um PDA, uma tecnologia de comunicação e um cenário, verificar se a dupla “PDA + tecnologia” atendem ao cenário. Neste caso, um cenário é caracterizado por duas variáveis: taxa de transmissão e autonomia da bateria. A taxa de transmissão refere-se à vazão mínima de dados necessária para atender uma aplicação móvel em questão; já a autonomia da bateria refere-se ao tempo de uso da aplicação sem a necessidade de recarregar a bateria do PDA.

O presente capítulo descreve o desenvolvimento do SSD denominado *PDA Chooser*, cujo principal objetivo é atender os requisitos funcionais supracitados. A seção 5.1 apresenta a modelagem da base de conhecimento do sistema. A seção 5.2 apresenta as funcionalidades da implementação do SSD. Finalmente, a seção 5.3 traz as considerações finais do capítulo.

5.1 Modelagem do SSD

Quanto à organização, um SSD é tipicamente formado por três subsistemas: de diálogo, de dados e de modelo. O subsistema de diálogo serve de interface entre o usuário e o sistema. O subsistema de dados é responsável pela criação, alteração e recuperação dos dados. E, por fim, o subsistema de modelo implementa um mecanismo de inferência que utiliza a base de conhecimento (subsistema de dados) para gerar uma solução.

Durante a modelagem do SSD, utilizou-se a ferramenta GNU Prolog [Diaz, 2003] que disponibiliza os subsistemas que compõem um SSD típico em um único *software*. Esta ferramenta fornece um *shell* para realização de consultas (subsistema de diálogo), implementa uma máquina de inferência (subsistema de modelo) e utiliza a linguagem lógica Prolog para definir as regras e fatos da base de conhecimento (subsistema de dados).

A base de conhecimento foi construída a partir dos resultados apresentados no capítulo 4. Tais resultados permitem caracterizar os PDAs avaliados em função da taxa de transmissão e da autonomia da bateria, a Figura 38 sintetiza estes dados. Nesta figura, são definidas áreas delimitadas pela taxa de transmissão e pela autonomia dos dispositivos observada nos experimentos. Para verificar se PDA e tecnologia atendem a um cenário, basta representar o cenário como um ponto onde a abscissa é a autonomia da bateria e a ordenada é a taxa de transmissão. Se o ponto estiver localizado dentro da área definida pelos limites de um PDA e tecnologia de comunicação, significa que esta dupla atende os requisitos da aplicação em questão.

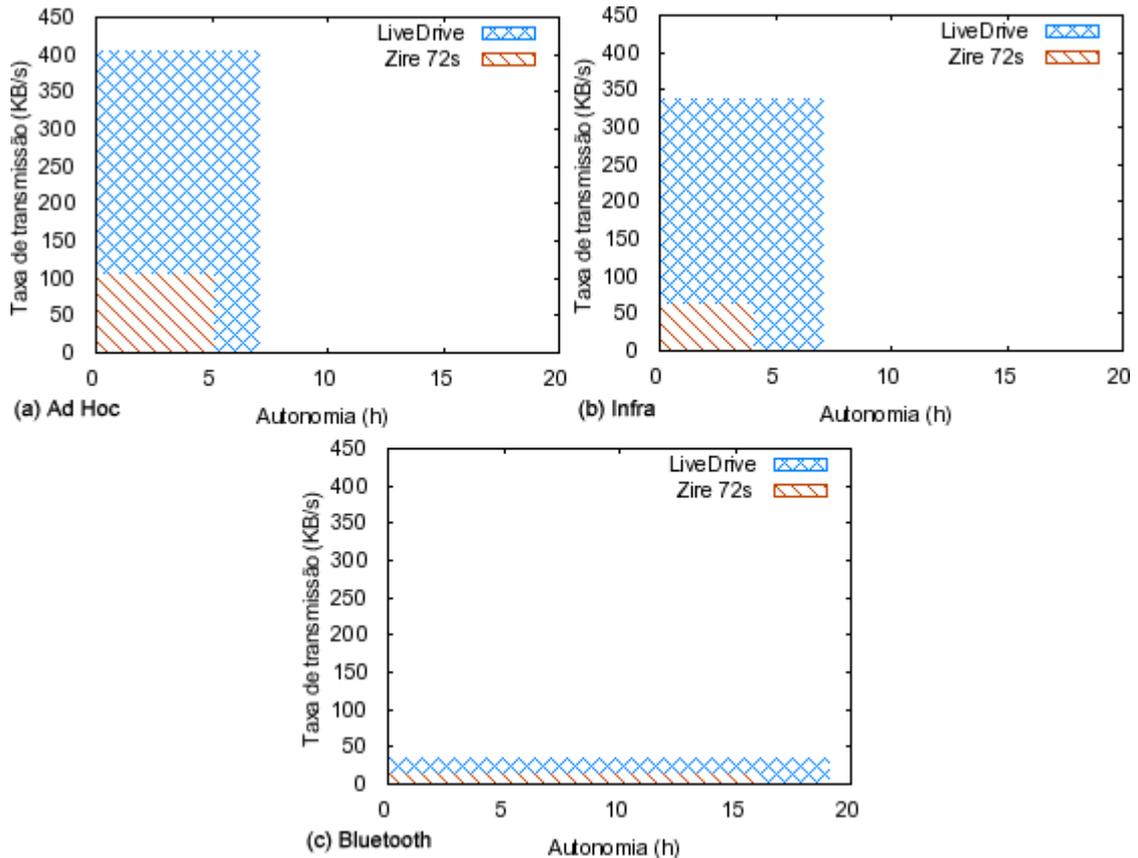


Figura 38. Caracterização dos PDAs e tecnologias de acesso.

Os requisitos funcionais supracitados podem ser escritos no formato de questões, as quais o SSD deve fornecer resposta, elas são: (i) Quais PDAs e tecnologias de comunicação conseguem prover T KB/s de taxa de transmissão com uma autonomia de H horas? (ii) O PDA P com a tecnologia X consegue prover uma taxa de T KB/s e H horas de autonomia? (iii) Quais tecnologias de comunicação do PDA P conseguem prover T KB/s de taxa de transmissão e H horas de autonomia?

A primeira tarefa foi representar os resultados mostrados na Figura 38 como fatos na base de conhecimento. Há dois fatos: $autonomia(\langle Autonomia \rangle, \langle PDA \rangle, \langle Tecnologia \rangle)$ e $txTransmissao(\langle Taxa \rangle, \langle PDA \rangle, \langle Tecnologia \rangle)$. A Figura 39 apresenta os principais predicados da base de conhecimento. Para ilustrar o mapeamento de resultados para fatos foram utilizados os resultados do Zire na rede Wi-Fi *ad hoc*, que são apresentados pela Figura 38 (a). Em média, o Zire tem uma autonomia próxima de 5 horas, $autonomia(5.03, zire, wi-$

fi_adhoc) (linha 38) e taxa de transmissão em torno de 105 KB/s, *txTransmissao*(105.33, *zire*, *wi-fi_adhoc*) (linha 45). O mesmo mapeamento é realizado para as outras redes, conforme mostra a Figura 39 linhas 38 a 50.

01	atendeRequisitos (Taxa, Autonomia):-	27	<i>verificaTaxaTransmissao</i> (_,_) :- !.
02	<i>verificaAutonomiaBateria</i> (Autonomia),	28	
03	<i>verificaTaxaTransmissao</i> (Taxa),	29	<i>verificaAutonomiaBateria</i> (PDA, Autonomia) :-
04	<i>write</i> ('Requisitos atendidos em:'),	30	<i>autonomia</i> (Aut, PDA, Tecnologia),
05	<i>mostra</i> PDA's,	31	Aut >= Autonomia,
06	<i>abolish</i> (autonomia/3),	32	<i>assert</i> (bateria(Aut, PDA, Tecnologia)),
07	<i>abolish</i> (transmissao/3).	33	fail.
08		34	
09	prove (PDA, Tecnologia, Taxa, Autonomia) :-	35	<i>verificaAutonomiaBateria</i> (_,_) :- !.
10	<i>bateria</i> (Aut, PDA, Tecnologia),	36	
11	Aut >= Autonomia,	37	% Fatos
12	<i>txTransmissao</i> (Tx, PDA, Tecnologia),	38	<i>autonomia</i> (5.03, <i>zire</i> , <i>wi-fi_adhoc</i>).
13	Tx >= Taxa.	39	<i>autonomia</i> (4.96, <i>zire</i> , <i>wi-fi_infra</i>).
14		40	<i>autonomia</i> (16.9, <i>zire</i> , <i>bluetooth</i>).
15	tecnologias (PDA, Taxa, Autonomia) :-	41	<i>autonomia</i> (7.90, <i>lifedrive</i> , <i>wi-fi_adhoc</i>).
16	<i>verificaTaxaTransmissao</i> (PDA, Taxa),	42	<i>autonomia</i> (7.58, <i>lifedrive</i> , <i>wi-fi_infra</i>).
17	<i>verificaAutonomiaBateria</i> (PDA, Autonomia),	43	<i>autonomia</i> (19.6, <i>lifedrive</i> , <i>bluetooth</i>).
18	<i>mostra</i> Tecnologias,	44	
19	<i>abolish</i> (transmissao/3),	45	<i>txTransmissao</i> (105.33, <i>zire</i> , <i>wi-fi_adhoc</i>).
20	<i>abolish</i> (autonomia/3).	46	<i>txTransmissao</i> (61.42, <i>zire</i> , <i>wi-fi_infra</i>).
21		47	<i>txTransmissao</i> (14.10, <i>zire</i> , <i>bluetooth</i>).
22	verificaTaxaTransmissao (PDA, Taxa) :-	48	<i>txTransmissao</i> (405.73, <i>lifedrive</i> , <i>wi-fi_adhoc</i>).
23	<i>txTransmissao</i> (Tx, PDA, Tecnologia),	49	<i>txTransmissao</i> (338.21, <i>lifedrive</i> , <i>wi-fi_infra</i>).
24	Tx >= Taxa,	50	<i>txTransmissao</i> (36.00, <i>lifedrive</i> , <i>bluetooth</i>).
25	<i>assert</i> (transmissao(Tx, PDA, Tecnologia)),		
26	fail.		

Figura 39. Fatos e regras da base de conhecimento.

O primeiro requisito (i) é atendido pela regra *atendeRequisitos*(Taxa, Autonomia), conforme mostra a Figura 39 (linhas 1 a 7). Esta regra recebe como parâmetros a taxa de transmissão (KB/s) e autonomia (h) e utiliza a regra *verificaAutonomiaBateria*(Autonomia) e a regra *verificaTaxaTransmissao*(Taxa) para percorrer os fatos e verificar quais PDAs e tecnologias têm desempenhos maiores ou iguais aos parâmetros fornecidos.

O segundo requisito (ii) exigiu a criação de uma regra que prove que um PDA *X* com a tecnologia de comunicação *Y* consegue atender uma taxa média *T* e uma autonomia *A*. A regra *prove*(PDA, Tecnologia, Taxa, Autonomia) (linhas 09 a 13) atende o requisito supracitado retornando sim ou não.

O terceiro requisito (iii) especifica que dado o PDA, a Taxa e Autonomia o sistema deve retornar a(s) tecnologia(s) de comunicação que atendem o cenário. Para tanto, foi

implementada a regra *tecnologias(PDA, Taxa, Autonomia)* (linhas 15 a 20) que retorna uma lista com as tecnologias que atendem o cenário.

5.2 PDA Chooser

Com o objetivo de prover uma interface com usuário mais simples e intuitiva que o *shell* do GNU Prolog, foi desenvolvida a aplicação denominada *PDA Chooser*. Esta aplicação é um SSD que utiliza a base de conhecimento desenvolvida a seção anterior para atender os requisitos funcionais supracitados. Tomando a organização de um SSD típico, a base de conhecimento desenvolvida representa o subsistema de dados. O subsistema de diálogo foi implementado com os componentes de interface disponibilizados pelo SDK da plataforma Java SE. E, por fim, foi utilizada uma implementação de referência da JSR 94 (*Java Specification Request*) que define uma máquina de inferência (*Rule Engine*) para máquina virtual Java.

Para realizar consultas a base de conhecimento o SSD disponibiliza um formulário onde o usuário pode compor as consultas por meio da definição do cenário e/ou seleção do PDA e tecnologia de acesso. Para ilustrar, a Figura 40 apresenta o formulário de consulta e define um cenário, cuja taxa de transmissão é de 30 KB/s e autonomia de 3 horas, em outras palavras, esta consulta deseja recuperar todos os PDAs e tecnologias de acesso que conseguem prover no mínimo: taxa de transmissão de 30 KB/s e autonomia de 3 horas. O resultado da consulta é apresentado na Figura 41.

Figura 40. Formulário de consulta.

. O formulário de resultados utiliza um plano cartesiano semelhante ao apresentado na Figura 38, onde são apresentadas áreas delimitadas pela taxa de transmissão e pela autonomia da dupla formada pelo PDA e tecnologia de acesso. O cenário é representado por um ponto onde a abscissa é a autonomia e a ordenada é a taxa de transmissão.

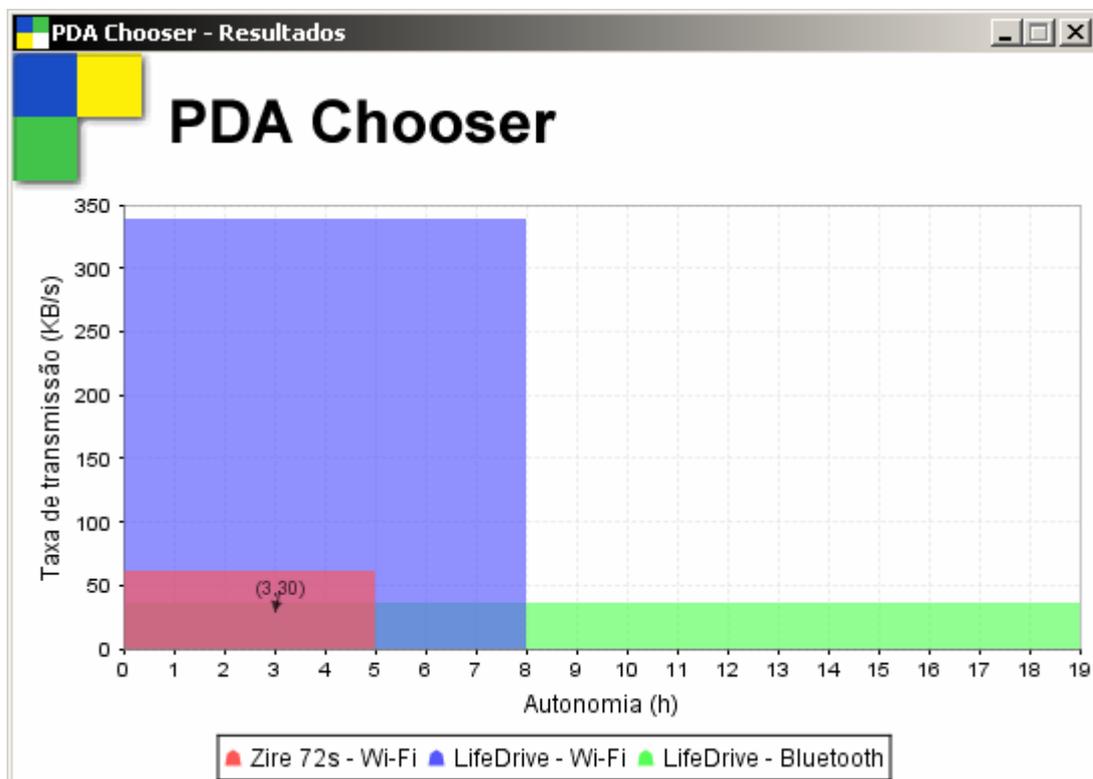


Figura 41. Formulário com os resultados da consulta.

5.3 Considerações Finais

Este capítulo descreveu a modelagem e implementação do SSD denominado *PDA Chooser*, cujo principal objetivo é auxiliar a escolha do PDA e da tecnologia de acesso para determinado cenário. Para tanto, os resultados obtidos nos experimentos realizados neste trabalho foram mapeados para fatos na base de conhecimento do SSD e, além disso, foram criadas regras que utilizam os fatos para atender os requisitos funcionais do sistema. Esta base de conhecimento foi criada com uma linguagem lógica e, dessa forma, permite fácil consulta, atualização e manutenção. Em outras palavras, um especialista humano pode inserir resultados de experimentos com novos PDAs, rastrear uma solução e realizar manutenção.

6 Conclusões

O desenvolvimento da próxima geração das redes de acesso sem fio é um dos principais desafios da Computação Móvel. Estas redes provavelmente serão complexos sistemas heterogêneos compostos por diversas redes com tecnologias de acesso sem fio. Hoje, existe uma crescente demanda científica para melhorar a experiência dos usuários móveis nestes ambientes de rede. Entre as frentes de trabalho, merece destaque a interoperabilidade entre as tecnologias de acesso sem fio para permitir que o usuário permaneça sempre conectado à melhor rede disponível, em termos de custo, consumo de energia, QoS e segurança. Deseja-se que este serviço seja provido de forma transparente, integrada e conveniente.

Inicialmente, o presente trabalho tinha o objetivo de implementar a seleção automática da tecnologia de acesso, na camada de aplicação, visando à maximização da autonomia do PDA. Dessa forma, um mecanismo de seleção recuperaria a quantidade de energia disponível no dispositivo em um determinado instante, estimaria a autonomia com cada tecnologia de acesso suportada e, por fim, escolheria a tecnologia de acesso que permitisse a maior autonomia e que atendesse os requisitos de QoS da aplicação em questão. Contudo, observou-se que PDAs com características de *hardware* diferentes apresentavam comportamentos distintos, durante a comunicação em redes sem fio. Esta observação inicial gerou a hipótese que as características do *hardware* do PDA influenciam seu comportamento durante a comunicação. Portanto, iniciou-se um esforço para investigar este comportamento.

O foco desta investigação foi a avaliação quantitativa da capacidade de comunicação e do consumo de bateria. Uma vez conhecida estas duas variáveis é possível determinar a eficiência da dupla PDA e tecnologia de acesso. Para tanto, o Capítulo 3 apresentou os

métodos adotados, o ambiente experimental e o ambiente de simulação. Foram experimentados dois PDAs com características de *hardware* distintas. Os resultados dos experimentos foram apresentados no Capítulo 4. Ainda neste capítulo, foi realizada uma comparação entre os dois PDAs experimentados e foi apresentada uma discussão confrontando os resultados experimentais com os resultados obtidos nas simulações. Os resultados experimentais indicaram uma diferença significativa no comportamento dos dois PDAs, ratificando a hipótese levantada. O PDA com maior capacidade de processamento obteve os melhores resultados, contudo esta desigualdade não é proporcional à diferença na capacidade de processamento, especialmente nas redes Wi-Fi. O LifeDrive tem um processador 25 % mais rápido que o do Zire, contudo foi cerca de 6 e 3 vezes mais eficiente que o Zire, nas redes Wi-Fi infra-estruturada e *ad hoc*, respectivamente. Contudo, na rede Bluetooth, o LifeDrive foi cerca de 22 % mais eficiente que o Zire. Pode-se concluir que a sobrecarga da utilização de conexões TCP/IP encapsuladas em um túnel PPP sobre um enlace Bluetooth é custosa.

Devido a diversas restrições técnicas, a implementação de um mecanismo de seleção automática de acesso eficiente na camada de aplicação é inviável, especialmente em sistemas operacionais mono *thread* como o Palm OS. Dessa forma acredita-se que hoje a escolha do dispositivo e tecnologia de acesso deve ser feita na fase de projeto da aplicação. Para auxiliar esta escolha foi modelado e implementado um SSD denominado *PDA Chooser*, cujo principal objetivo é auxiliar a escolha do PDA e tecnologia de acesso para um determinado cenário. A base de conhecimento do sistema foi desenvolvida a partir dos resultados experimentais, utilizando uma linguagem lógica. Com isso, os resultados são disponibilizados de forma objetiva e de fácil acesso, manutenção e atualização.

Em suma, as contribuições deste trabalho são o estudo e levantamento dos principais problemas acerca da implementação do conceito ABC; a metodologia para avaliação

quantitativa da capacidade de comunicação e consumo de energia de PDAs durante a comunicação; e, por fim a modelagem e implementação de um sistema de suporte à decisão, cuja base de conhecimento foi concebida a partir dos resultados experimentais. Diante desta metodologia de medição elaborada, a disponibilidade de novos dispositivos para teste deve ampliar o auxílio ao usuário. Espera-se que, futuramente, este sistema possa fornecer informações para algoritmos de seleção automática de acesso.

6.1 Trabalhos Futuros

Como sugestão para trabalhos futuros recomenda-se o estudo e implementação de um método para avaliação da capacidade de comunicação mais refinado e não intrusivo, que obtenha o número de erros e atrasos médios dos pacotes na camada de transporte. Além disso, seria interessante monitorar o nível do sinal recebido pelo dispositivo, para relacionar as taxas de transmissão ao nível do sinal.

Os experimentos foram realizados em ambientes ótimos quanto à concorrência ao acesso ao meio, pois em todos os casos, só havia um PDA no ambiente de rede. Um estudo sobre o impacto da concorrência ao acesso ao meio na eficiência dos dispositivos móveis também é interessante. Uma vez que o desempenho da rede sem fio decresce em função do número de elementos computacionais comunicantes.

A implementação do mecanismo de seleção automática acesso visando à maximização da autonomia do dispositivo pode ser feita em um ambiente de simulação. Seria interessante especializar o mecanismo de seleção para levar em consideração o estado atual das redes disponíveis. Dessa forma, seria necessária a implementação de um módulo para o NS2 que permitisse a recuperação do estado das redes, em termos de qualidade do sinal e percentual de utilização da rede. Com isso, tem-se um mecanismo de seleção automática de acesso ciente do contexto.

A base de conhecimento do *PDA Chooser* utiliza dados médios das taxas de transmissão e do consumo de bateria obtidos durante o envio e recepção de dados. Portanto, tem-se uma caracterização geral do comportamento dos dispositivos. Uma especialização da base de conhecimento visando à representação mais precisa do comportamento dos equipamentos é relevante.

Referências Bibliográficas

- Akyildiz, I. F., Wang, X., Wang, W. (2005) “Wireless mesh networks: a survey”. *Computer Networks Journal* (Elsevier), Vol. 47, pp. 445-487.
- Bluetooth SIG. (2002) “Specification of the Bluetooth system”. Specification Version 1.1. Bluetooth SIG.
- Bhagwat, P. (2001) “Bluetooth: Technology for Short-Range Wireless Applications”, *IEEE Internet Computing*, v. 5, n. 3, p. 96-10.
- Crow, B. P., Widjaja, I., Kim, J. G., Sakai, P. T. (1997) “IEEE 802.11 Wireless Local Area Networks”. *IEEE Communications Magazine*, vol. 37, p. 116-126.
- Das A., Ghose A., Razdan A., Sarany H. and Shorey R. (2001) “Enhancing Performance of Asynchronous Data Traffic over the Bluetooth Wireless Ad-hoc Network”, *IEEE Infocom*, vol. 1, p 591-600.
- Dias, K. L., Cavalcanti, D. A. T., Costa, J. C. W. A. e Cordeiro, C. M. (2006) “Gerenciamento de Mobilidade e Arquiteturas para Redes 4G”. Capítulo 1 dos Anais dos Minicursos do SBRC2006 p. 3-51.
- Diaz, D. (2003) “The GNU Prolog”. Disponível em: <http://gnu-prolog.inria.fr/>, acessado em 09/2006.
- Fall, K., Varadhan, K. (2006) “The ns Manual”. Disponível em: http://www.isi.edu/nsnam/ns/doc/ns_doc.pdf, acessado em 12/2006.
- Feeney, L. M., Nilson, M. (2001) “Investigating the Energy Consumption of a Wireless Network Interface in an Ad Hoc Networking Environment”. *IEEE INFOCOM* p. 1548-1557.
- Fodor, G., Eriksson, A. and Tuoriniemi, A. (2003) “Providing QoS in Always Best Connected Networks”, *IEEE Communications Magazine*, Vol. 41, No. 7, p. 154-163.
- Fodor, G., Furuskär, A., Lundsjo, J. (2004) “On Access Selection Techniques in Always Best Connected Networks”, *Proceedings of the 16th ITC Specialist Seminar on Performance Evaluation of Wireless and Mobile Systems*, Antwerp, Belgium.
- Foster, L. R. (2002) “Palm OS Programming Bible”. John Wiley & Sons, Inc. New York, NY, USA.
- Furuskär, A., Zander, J. (2005) “Multi-service Allocation for Multiaccess Wireless Systems”, *IEEE Transactions on Wireless Communication*, Vol. 4, No 1, p. 174-184.

- Furuskär, A. (2002) "Allocation of Multiple Services in Multi-Access Wireless Networks", in proceedings of the IEEE Mobile and Wireless Communication Networks, MWCN '02, p. 261-265.
- Gast, M. 802.11. (2002) "Wireless Networks: The Definitive Guide". O'Reilly.
- Gustafsson, E. and Jonsson, A. (2003) "Always Best Connected", IEEE Wireless Communications, Vol. 10, No. 1, p. 49-55.
- Hager, C. T. R., Midkiff, S. F., Park, J., Martin, T. L. (2005) "Performance and Energy Efficiency of Block Ciphers in Personal Digital Assistants". Third IEEE International Conference on Pervasive Computing and Communications.
- Harold, E. R. (2000) "Java Network Programming". O'Reilly, Second Edition.
- Imielinski, T., Badrinath, B. R. (1994) "Mobile wireless computing". Communications of ACM Vol. 30, p. 18-28.
- Indulska, J. and Balasubramaniam, S. (2003) "Vertical handover supporting pervasive computing in future wireless networks". Computer Communications – Advances in Future Mobile/Wireless Network and Services, Vol. 27, p. 708-719.
- Java SE (2006) "Java SE - Overview", página Web disponível em <<http://java.sun.com/javase/>>. Acessado em março de 2006.
- Junior, F. C., Gaspary, L. P., Pfitscher, L. L., Cavalheiro, G. G. H., Barbosa, J. L. V., Ramos, J. D. G. (2005) "Avaliando o Impacto na Recepção de Dados e no Consumo de Energia do Uso de Dispositivos Móveis para Acessar Redes Wi-Fi de Forma Segura". SBRC2005 p. 45-55.
- Kleinrock, L. (1995) "Nomadic computing – an opportunity". ACM SIGCOMM Computer Communication Review Vol. 25, p. 26-40.
- LabPalm (2006) "Projeto LabPalm", página Web disponível em <<http://labpalm.dc.ufscar.br>>. Acessado em dezembro de 2006.
- Maier, D., Warren D. S. (1988) "Computing with Logic: Logic Programming with Prolog". Benjamin/Cummings.
- Mateus, G. R. e Loureiro, A. A. F. (2004) "Introdução à Computação Móvel". Versão preliminar da segunda edição. Disponível em: <<http://homepages.dcc.ufmg.br/~loureiro/cm-rssf.html>>.
- Muller, N. J. (2000) "Bluetooth Demystified". McGraw-Hill Professional Publishing.
- NS2 - Network Simulator. <<http://www.isi.edu/nsnam/ns>>. Acessado em novembro de 2005.
- Prasad, R., Dovrolis, C., Murray, M., Claffy, K. (2003) "Bandwidth estimation: metrics, measurement, and tools", Network IEEE p27-35.

- Postel, J., Garlick L. L., Rom R. (1981) “Transmission Control Protocol”, RFC 793.
- Proakis, J. G. (2001) “Digital Communications”. 4. ed. New York: McGraw-Hill.
- Simpson, W. (1994) “The Point-to-Point Protocol (PPP)”, RFC 1661.
- Sprague, R.H. Jr., Carlson, E.D. (1982) “Building Effective Decision Support Systems”, Prentice-Hall, Englewood Cliffs, NJ.
- Stein, J. C. (2003) “Indoor Radio WLAN Performance”, Intersil Corporation.
- Stuart W. (2000) “IrDA: Past, Present and Future”, IEEE Personal Communications.
- Thomas, G. B. (2002) “Cálculo – Volume 1”. 10 ed. São Paulo: Addison Wesley.
- Vanni, R. M. P., Moreira, E. S., Goulart G. (2006) “DOHand: An ontology to support building services to exploit handover information in mobile heterogeneous networks”. 5th International Information and Telecommunication Technologies Symposium, Vol. 1, p. 105-112.
- Wi-Fi (2006) “Wireless Fidelity Alliance”, página Web disponível em <<http://www.wi-fi.org/>>. Acessado em março de 2006..
- Wilson G., Ostrem J., Bey C, Dugger M. (2003) “Palm OS® Programmer’s API Reference”. PalmSource, Inc.
- Wilson, G.; Ostrem, J.; Bey, C. (2004) “Palm OS Programmer’s Companion, Volume I”, PalmSource.

Anexos

Implementação dos Experimentos

As próximas subseções descrevem os principais pontos da implementação dos dois módulos, *Mobile* e *Server*, da aplicação *PDA Stress*.

PDA Stress Mobile

O módulo *Mobile* foi implementado com a linguagem de programação C para o sistema operacional (S.O.) Palm OS [Foster, 2002]. Este S.O. tem um SDK (*Software Development Kit*) para desenvolvimento de aplicações, em C/C++, que disponibiliza uma API (*Application Programming Interface*) chamada 68K [Wilson *et al.*, 2004]. A API 68K possui uma biblioteca para criação de *sockets* TCP/IP, chamada NetLib, onde estão implementadas funções parecidas com a API de *sockets* de Berkley.

O experimento de envio de dados foi implementado pelo trecho de código apresentado pela Figura 42. Nesta figura, tem-se um laço de repetição (linha 02) mais externo que é o responsável por realizar cada uma das dez etapas do experimento. Um *switch* (linha 03) verifica qual é a etapa atual, aloca o *buffer* do tamanho definido para etapa vigente e define o número de iterações necessárias, no laço de envio, para somar 3 MBytes com *buffer* alocado. A função *conectar()* (linha 20), recebe como parâmetro o endereço IP (*Internet Protocol*) e a porta onde o módulo *Server* está executando, realiza a conexão e retorna o *socket* da conexão. Dentro de um laço de envio (linha 25), a função *enviar()* recebe como parâmetro o *socket* da conexão, o *buffer* e o tamanho do *buffer* e escreve o *buffer* no *socket*, esta função é apresentada em detalhes na Figura 43. Terminado o envio de dados, a função *desconectar()* (linha 29) é chamada para fechar a conexão com o módulo *Server* e, em seguida, o *buffer* é


```

49         tamanhoBuffer - bytesEnviados, // Tamanho.
50         0, // Flags.
51         NULL, // Endereço do receptor.
52         0, // Tamanho do endereço.
53         TIMEOUT, // Timeout em centésimos de seg.
54         &error); // Ponteiro para código do erro.
55
56         // Verifica se o volume de dados enviados é maior que zero.
57         if (bytes > 0)
58             bytesEnviados += bytes;
59         else
60             break;
61     } // while
62 }

```

Figura 43. Trecho do código fonte da função *enviar()*.

O experimento de recepção possui uma implementação parecida com o experimento de envio apresentado na Figura 42; a principal diferença ocorre na linha 24 onde foi chamada a função *receber()* no lugar da função *enviar()*. A função *receber()* é responsável pela leitura de dados em um *socket*; para tanto, ela recebe como parâmetro o *socket* da conexão, o *buffer* de recepção e o tamanho deste *buffer*. Para realizar a recepção a função utiliza um laço de repetição cuja condição de parada é uma tentativa frustrada de leitura no *socket*, *i.e.* o módulo *Server* encerrou a conexão. Dentro do laço de recepção, é chamada a função *NetLibReceive()* (linha 69) que recebe vários parâmetros, entre eles o *socket* da conexão, o *buffer*, *offset* e *timeout.*, e realiza a leitura dos dados no *socket*.

```

63 // Recebe dados por meio de um socket.
64 void receber(NetSocketRef socket, CharPtr buffer, Int32 tamanhoBuffer){
65     bytes = 0; // Volume de bytes enviados em cada iteração.
66     bytesRecebidos = 0; // Total de bytes enviados.
67
68     do{
69         bytes = NetLibReceive(AppNetRefnum, // Referência para NetLib.
70                             socket, // Referência para o socket.
71                             (buffer + bytesRecebidos), // Offset.
72                             (tamanhoBuffer - bytesRecebidos),
73                             0, // Flags.
74                             NULL, // Endereço do receptor.
75                             0, // Tamanho do endereço.
76                             TIMEOUT, // Timeout em cetesimos de seg.
77                             &error); // Ponteiro para código do erro.
78         // Incrementa a quantidade de bytes recebidos
79         if(bytes > 0)
80             bytesRecebidos += bytes;
81
82         // Verifica se o buffer está cheio.
83         if(bytesRecebidos >= tamanhoBuffer)
84             bytesRecebidos = 0;
85
86     } while (bytes > 0);
87 }

```

A Figura 44 apresenta o formulário principal do módulo *Mobile* sendo executado em um simulador do Palm OS. Neste formulário, é possível definir o endereço IP (1) do

computador onde está executando o módulo *Server*, as portas de envio/recepção e resultados (2 e 3) e o *delay*, em milissegundos, entre cada experimento. Também são apresentados os seguintes dados: número de *ticks* por segundo que o sistema operacional está utilizando, tempo em segundos no início da etapa do experimento, tempo em segundos no final da etapa do experimento, tempo transcorrido e *log*.



Figura 44. Interface do módulo Mobile.

PDA Stress Server

O módulo *Server* foi implementado com a linguagem de programação Java, especificamente com o SDK da plataforma Java SE (*Java Standard Edition*) [Java SE, 2006], utilizado para criação de aplicações para computadores convencionais. Os experimentos são implementados por duas classes: *PDASTressServer* e *ThreadClient*. A primeira classe é responsável por esperar uma conexão do módulo Mobile em uma porta dada, já a segunda classe é responsável por atender a requisição realizando as etapas dos experimentos de envio e recepção de dados.

A Figura 45 apresenta o principal trecho de código da classe *PDASTressServer*. Nesta figura tem-se, na linha 88, a instanciação de um objeto *socket* do tipo *ServerSocket* que recebe

como parâmetro a porta que o servidor aguardará as conexões. Depois é definido um laço infinito para aguardar as conexões do módulo *Mobile*. Dentro deste laço, é chamado o método *accept()* do objeto *socket*. Tal método implementa a primitiva ACCEPT do protocolo TCP que fica bloqueada até receber um CONNECT. Quando o módulo *Mobile* efetua a conexão é instanciado um objeto *thread* do tipo *ThreadClient* que recebe como parâmetro o *socket* da conexão.

```

88  socket = new ServerSocket(porta);
89
90  // Laço infinito para aguardar conexões
91  while(true) {
92      // Primitiva bloqueante que aguarda conexões
93      cSocket = socket.accept();
94
95      // Cria uma thread para atender a conexão
96      ThreadClient thread = new ThreadClient(cSocket);
97      thread.start();
98  } // while

```

Figura 45. Trecho de código da classe *PDAStressServer*.

A classe *ThreadClient* implementa efetivamente o lado servidor dos experimentos propostos. Se o experimento for de envio de dados será executado o trecho de código apresentado na Figura 46. Nesta figura, tem-se a instancição de um *buffer* de leitura e de um objeto de escrita a partir do *socket* da conexão (linhas 127 e 128). O objeto de leitura é utilizado para recuperar o número da etapa a ser executada (linha 132). Uma estrutura condicional do tipo *switch/case* (linhas 134-146) identifica a etapa vigente define o número de iterações e aloca o *buffer*. Um laço de repetição realiza a escrita no socket até completar 3 MBytes (linhas 149-153). O buffer é enviado por meio do método *print()* do objeto de escrita (linha 151). Para finalizar, o *socket* é fechado (linhas 156-158).

```

127 //Instancia um buffer de leitura e escrita a partir do socket.
128 is = new BufferedReader(new InputStreamReader(cSocket.getInputStream()));
129 ps = new PrintStream(cSocket.getOutputStream());
130
131 // Recupera o número da etapa a ser realizada.
132 etapa = is.readLine();
133
134 switch(Integer.parseInt(etapa)){
135     case 1: // Etapa 1 - 64 bytes
136         iteracoes = 49152;
137         alocaBuffer(64);
138         break;
139
140     // ... demais etapas (2, 3, 4, 5, 6, 7, 8 e 9)
141

```

```

142         case 10: // Etapa 10 - 32 K
143             iteracoes = 96;
144             alocaBuffer(32768);
145             break;
146     }
147
148     // Laço de envio de dados
149     while(i < iteracoes){
150         // Escreve no socket
151         ps.print(buffer);
152         i += 1;
153     }
154
155     //Fecha o socket
156     cSocket.shutdownInput();
157     cSocket.shutdownOutput();
158     cSocket.close();

```

Figura 46. Trecho de código da classe *ThreadClient* - Envio.

Se o experimento for de recepção, o trecho de código apresentado na Figura 47 será executado. Nesta figura, tem-se a chamada do método *alocaBuffer()* que recebe como parâmetro o tamanho do *buffer* de recepção, neste caso o tamanho do *buffer* é três MBytes. Um objeto do tipo *BufferedReader* é instanciado a partir do *socket* da conexão (linha 103). Recupera-se o tempo antes de iniciar a recepção (linha 106). Um laço de repetição realiza a leitura sistemática do *socket* por meio do método *read()* do *buffer* de leitura. Terminada a recepção, computa-se o tempo total do experimento (linha 122) e fecha-se a conexão (linhas 125 e 126).

```

99     //Aloca buffer
100     alocaBuffer(tamanhoBuffer);
101
102     // Instancia um buffer de leitura a partir do socket.
103     is = new BufferedReader(new InputStreamReader(cSocket.getInputStream()));
104
105     // Recupera o tempo antes de iniciar a recepção
106     inicio = System.currentTimeMillis();
107
108     // Laço para leitura dos dados no socket
109     do { // while
110
111         // Lê dados no socket.
112         bytes = is.read(buffer, subTotal, tamanhoBuffer - subTotal);
113
114         if(bytes > 0)
115             subTotal += bytes;
116         else
117             break;
118     }while (subTotal < tamanhoBuffer); // while
119
120
121     // Recupera o tempo no final do experimento
122     tempo = System.currentTimeMillis() - inicio;
123
124     // Fecha o socket
125     cSocket.shutdownInput();
126     cSocket.close();

```

Figura 47. Trecho de código da classe *ThreadClient* - Recepção.

O módulo *Server* é executado em computadores convencionais, e comunica-se, enviando e recebendo dados, com o módulo *Mobile*, executado no PDA. O módulo *Server* ainda é responsável por armazenar *logs* da comunicação. A Figura 8 apresenta os dois formulários da aplicação. O primeiro formulário apresenta os resultados de cada etapa do experimento e o segundo formulário apresenta a etapa do experimento que está sendo executada.

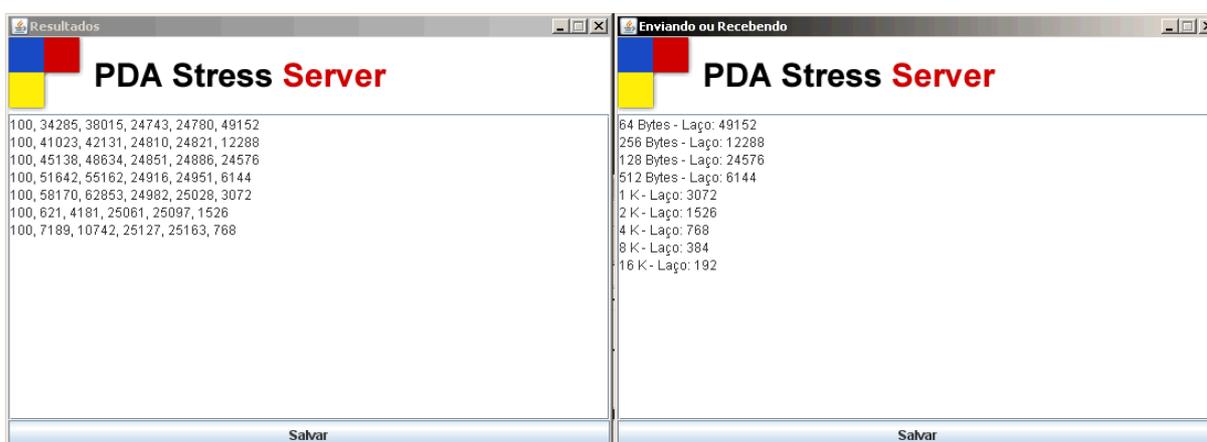


Figura 48. Interface gráfica do módulo *Server*.

Os *logs* de cada etapa dos experimentos são apresentados conforme a Tabela 8. Nesta tabela tem-se a etapa do experimento, o número de *ticks* por segundo do sistema operacional, número de *ticks* no início da etapa, número de *ticks* no final da etapa, tempo em segundos no início da etapa, tempo em segundos no final da etapa e o número de iterações no laço de repetição durante o envio ou recepção de dados na rede.

Tabela 8. Formato dos logs da aplicação PDA Stress Server.

Etapa	Ticks/s	Ticks início	Ticks fim	Tempo início	Tempo fim	Número de repetições
1	100	4395	7897	56876	56911	49152
2	100	10906	14362	56941	56976	24576
3	100	17372	20794	57006	57040	12288
4	100	23803	27314	57070	57105	6144
5	100	30323	34239	57136	57175	3072
6	100	37248	41373	57205	57246	1526
7	100	44382	48529	57276	57318	768
8	100	51538	55831	57348	57391	384
9	100	58840	63216	57421	57464	192
10	100	691	5112	57495	57539	96