2008-08-13

Autonomous and Intelligent Radio Switching

Quiyi Duan
Brigham Young University - Provo

AUTONOMOUS AND INTELLIGENT RADIO SWITCHING

by

Qiuyi Duan

A dissertation submitted to the faculty of

Brigham Young University

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Computer Science

Brigham Young University

December 2008

BRIGHAM YOUNG UNIVERSITY

GRADUATE COMMITTEE APPROVAL

of a dissertation submitted by

Qiuyi Duan

This dissertation has been read by each member of the following graduate committee
and by majority vote has been found to be satisfactory.

_____   _____
Date   Charles D. Knutson, Chair

_____   _____
Date   Daniel M.A. Zappala

_____   _____
Date   Michael A. Goodrich

_____   _____
Date   Scott N. Woodfield

_____   _____
Date   Quinn O. Snell

BRIGHAM YOUNG UNIVERSITY

As chair of the candidate's graduate committee, I have read the dissertation of Qiuyi Duan in its final form and have found that (1) its format, citations, and bibliographical style are consistent and acceptable and fulfill university and department style requirements; (2) its illustrative materials including figures, tables, and charts are in place; and (3) the final manuscript is satisfactory to the graduate committee and is ready for submission to the university library.

_____ _____
Date   Charles D. Knutson
               Chair, Graduate Committee

Accepted for the Department

_____ _____
Date   Kent E. Seamons
               Graduate Coordinator

Accepted for the College

_____ _____
Date   Thomas W. Sederberg
               Associate Dean, College of Physical and Mathematical Sciences

ABSTRACT

AUTONOMOUS AND INTELLIGENT RADIO SWITCHING

Qiuyi Duan

Department of Computer Science

Doctor of Philosophy

With the proliferation of mobile applications and the abundance of wireless devices, it is increasingly common for devices to support multiple radios. When two devices are communicating they should choose the best available radio based on user preference and application requirements. This type of "radio switching" should happen automatically, so that the system optimizes performance dynamically.

To achieve this objective, we design an Autonomous and Intelligent Radio Switching (AIRS) system to leverage the radio heterogeneity common in today's wireless devices. The AIRS system consists of three key components. First, we design a radio preference evaluation module to dynamically select the best radio according to users' preference, application's QoS requirements, and the device battery usage. Second, we propose a link quality measurement and prediction module to predict the radio quality under a variety of mobility and interference conditions. Third, we present a radio switching decision making module to switch to the preferred available radio intelligently, based on the preference and link quality evaluations.

The AIRS system maintains connectivity, as well as improves link quality, via dynamic and intelligent radio switching, regardless of interference or collisions from the interfaces of other devices. The radio preference evaluation module is able to generate and adjust a preference list dynamically. Multiple users' requirements are satisfied in a mutually beneficial manner and the selected radio is Pareto optimal. The link prediction module is able to achieve an accuracy above 90% under a variety of mobility and interference conditions. The module can dynamically increase the link measurement interval and significantly reduce its power consumption, without sacrificing accuracy. The decision algorithm uses several parameters to avoid switching radios too frequently, and is able to provide dynamic, but stable radio switching, while balancing the competing objectives of high throughput and low power consumption. Overall, the AIRS system is able to achieve high goodput (application level throughput) and long battery life as applied to handoff management in a frequently changing mobile environment.

ACKNOWLEDGMENTS

To my advisor, Dr. Knutson, thanks for teaching me the positive attitude of doing research and encouraging me as I'm upset. Thank you for loving me like your own child and the years of financial support.

To Dr. Zappala, thanks for your insightful comments and suggestions. Thank you for helping me through many tough problems and encouraging me as I'm frustrated. You teach me how to become a scientific researcher.

To Dr. Goodrich, thanks for your valuable suggestions and professional comments. Thank you for helping me with my first publication.

To my parents, thanks for your generous support and continuous encouragement. Your love leads me through many difficulties.

To my dear husband, Lei, thanks for always being there, supporting me and encouraging me. Your love makes me strong.

Thanks to all my friends for your valuable support.

Contents

Chapter 1

Introduction

Ubiquitous or pervasive computing has long been a vision of the computing community. Ideally, computing should be integrated into our environment, working behind the scenes to provide computation and communication, with graceful interaction with human users. Devices should be able to maintain connectivity, as well as improve link quality, in various situations.

Wireless devices play an important role in these visions, as radios can be embedded in many objects and have the potential to operate with a minimum of configuration. As mobile communication systems continue to evolve, mobile devices that support multiple physical transceivers are increasingly common. It is likely that wireless technologies will continue to proliferate and that devices will continue to support multiple radios and network stacks.

Complicating these visions, however, is the reality that no single wireless technology dominates the market nor provides the desired functionality in all situations. Each wireless technology presents certain strengths and weaknesses, and each shines within certain usage models. Cellular technology provides coverage over a wide area, but phone manufacturers are adding WiFi interfaces so that users can browse the web at a WiFi hotspot, with lower connection charges and possibly higher speeds. Likewise, laptops and cellphones, in addition to WiFi or cellular interfaces, have Bluetooth interfaces for exchanging data directly with other devices or peripherals, when

other network interfaces may be unavailable, too cumbersome, or consume too much power.

Because of this reality, we envision that devices ought to be able to seamlessly switch between available network connections on the fly in order to provide access to available services. For example, if a person wants to transfer images from a cellphone to a laptop, the devices should cooperate to make this happen however they can, regardless of whether the transfer utilizes a Bluetooth or WiFi connection. Furthermore, as the availability or quality of a connection changes due to the activity of other nodes or interference from other devices, devices should cooperate to switch to the best available interface, taking into account power and performance tradeoffs. In other words, wireless devices should exploit their heterogeneity in order to provide better service to end users. This kind of communication should "just work" rather than requiring the user to be involved.

Autonomously and intelligently optimizing connectivity between devices that support multiple radios is an emerging challenge in wireless networking. A multi-radio device should be able to evaluate radio configurations dynamically according to distinct user preferences, accurately predict future link quality rather than simple availability under a variety of mobility and interference conditions, and decide when to switch radios and which radio to choose on the fly. To be suitable for mobile devices, the radio switching mechanism should be computationally light, with modest communication overhead. In addition, this type of radio switching, typically classified as a soft, vertical handover, should allow each supported radio to have its own network stack.

To meet these challenges, our research group has developed the Quality of Transport (QoT) architecture [27], along with additional work exploring key components of this architecture [3,12,8,10,9]. The preliminary work with QoT demonstrates how devices with multiple radios can switch between radios, even if these radios use

different network stacks. What has been missing from QoT, however is the "brain" of the system. This includes: (a) periodic measurement of radios to determine their status; (b) future radio quality prediction; (c) decision making on switching to a different radio; and (d) preference negotiation between users.

In this dissertation, we develop an Autonomous and Intelligent Radio Switching (AIRS) system that provides all of this functionality. The AIRS system makes it possible for devices to switch between supported interfaces autonomously and intelligently, according to the dynamic performance characteristics of each interface. The AIRS system is designed to be compatible with the QoT architecture, but is not restricted to this application. The presented AIRS system is applicable to any architecture where smart and dynamic radio selection and switching are desired in a heterogeneous multi-radio environment.

1.1   Quality of Transport

Quality of Transport (QoT) [27] is an architecture that transparently and automatically manages session-layer protocol access to multiple radios in heterogeneous mobile environments. It functions as an intelligent layer inserted between the session/application layers (such as HTTP, FTP and OBEX) and transport layers (such as IrDA, Bluetooth, and TCP/IP), facilitating dynamic, transparent and intelligent radio switching for multi-radio devices in order to provide the highest quality data transfer capability within constantly changing mobile environments.

As Figure 1.1 demonstrates, QoT introduces upper and lower abstraction modules and provides transparency to existing protocols without requiring additional APIs. The upper module is referred to as the Transport Proxy Module (TPM) and appears to a session layer as if it were an interface to a specific radio. The lower module is referred to as the Transport Abstraction Module (TAM) and interacts with the transport layer as if it were an arbitrary (but indeterminate) session protocol.

3
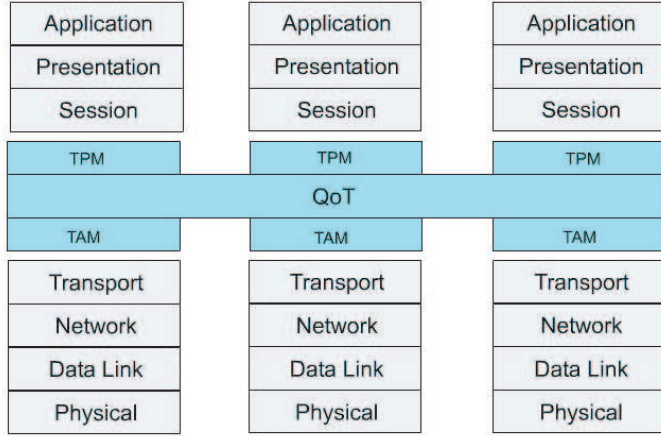
Figure 1.1: Data Exchange with QoT in Multi-Radio Environment

The TAM can also be viewed as presenting a consistent network interface to QoT, facilitating an extensible architecture from a radio perspective. The TAM manages radio connections and data transmission over the given radio.

The goal of QoT is to automatically manage the nature of the underlying data connection in order to maximize user experience and satisfaction [34]. If the active radio is disrupted, QoT attempts to connect over a less desirable (but available) radio without disturbing the session. We refer to this type of handoff as a *downgrade*. If QoT detects that a more preferred radio has become available, it attempts to connect over that radio in order to improve the link quality. We refer to this type of radio switching as an *upgrade*. In either case, the selected radio should be the optimal one among available radios, so that the user's requirements can be maximally satisfied during the communication.

Figure 1.2 illustrates a QoT-enabled data exchange between two devices using OBEX as the session layer protocol. The devices each support four radios, three of which are common (IrDA, Bluetooth, and IEEE 802.11b). Suppose that, at the time this figure presents, the most preferred link quality is provided by IrDA. In this situation, QoT would route the OBEX traffic via the IrDA stack (in dashed lines). As the user moves out of range of the active radio (IrDA) connection quality degradation
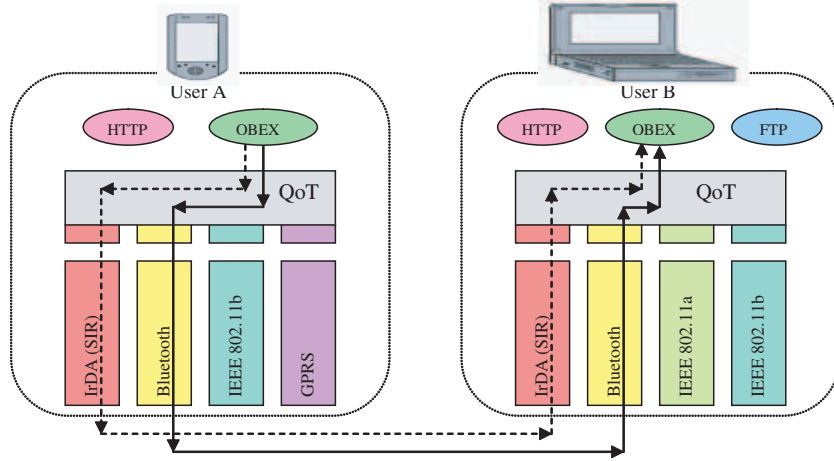
Figure 1.2: Data Exchange using QoT

is detected. In order to provide seamless connectivity with desired quality, the next preferred option, Bluetooth, would be selected. QoT would automatically switch the underlying radio to Bluetooth (in solid lines), without interrupting the connection or requiring user intervention.

1.2 Architecture of QoT

The basic architecture and preliminary implementation of QoT [27, 54], shown in Figure 1.3, comprises five primary modules, namely QoTCore, QoTBrain, DeviceManager, TPM, and TAM.

QoTCore is the "task handler" and "information provider" of the QoT architecture. It is the module that performs operations, such as making a connection, upgrading or downgrading the current radio, and transmitting data or control packets. QoTCore interfaces with the upper and lower abstraction modules (TPM and TAM), provides real-time information to the QoTBrain for decision making, and updates the dynamic radio information in the DeviceManager module.

QoTBrain serves as a "controlling administrator" in the QoT framework. It makes intelligent decisions autonomously based on the information provided by QoT-
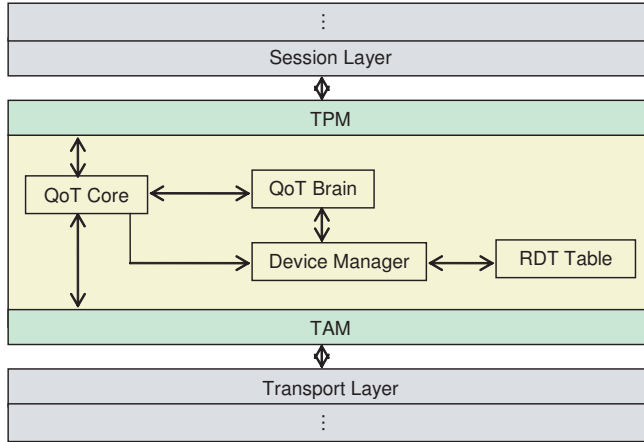
Figure 1.3: QoT Implementation Architecture

Core and DeviceManager, such as evaluating supported radios based on user preferences and application's QoS requirements, scheduling radio quality assessment queries with efficient query intervals, and initiating radio switchings (upgrade/downgrade) to a selected radio at an appropriate time. QoTCore takes these orders from QoTBrain and executes the given tasks. Meanwhile, QoTBrain periodically evaluates and predicts radio quality, then dynamically updates the corresponding information in the DeviceManager module.

DeviceManager dynamically maintains a Remote Device Table (RDT), functioning as a central repository of radio property and quality information, as shown in Figure 1.4. Radio information recorded in the RDT table includes radio connection parameters (such as IP addresses for TCP/IP), quality levels of relevant evaluation metrics, availability prediction, and other link descriptors (such as coverage, cost, and power consumptions).

1.3 Autonomous and Intelligent Radio Switching System

For this dissertation, we developed the Autonomous and Intelligent Radio Switching (AIRS) system, which acts as the QoTBrain in the QoT architecture. As illustrated
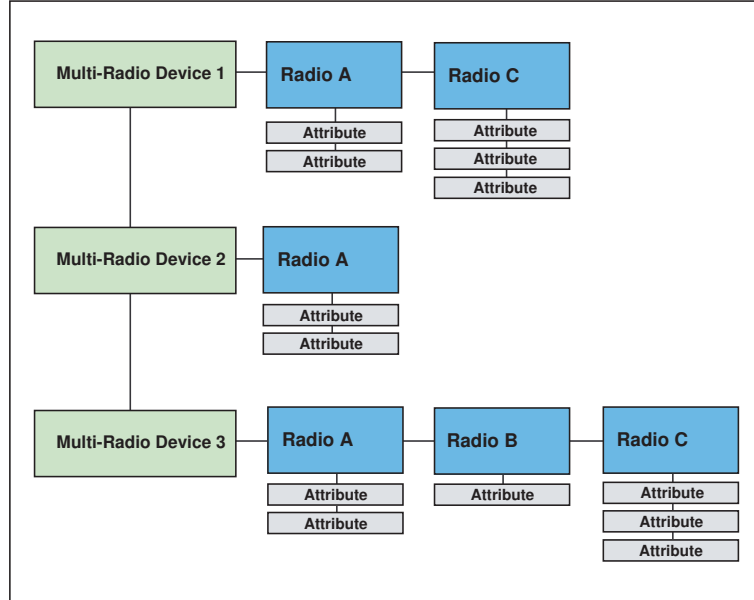
Figure 1.4: Remote Device Table (RDT)

in Figure 1.5, the AIRS system is composed of four key modules, which measure the radio status, predict link quality, and decide when to switch radios based on user preference, application requirements and remaining battery life.

1.3.1 Radio Preference Evaluation

The *Radio Preference Evaluation* module is designed to produce and dynamically maintain a preference list, ordering the available radios according to user preferences, the application's QoS requirement, and the usage of device battery. This module provides three evaluation modes: "high throughput", "power efficient", and "adaptive". In either "high throughput" or "power efficient" mode, the supported radios are ranked according to their performance on the concerned factor. For example, consider a device with WiFi, Bluetooth, and WirelessUSB. If the user selects the "power efficient" mode, then these three radios are evaluated based on their power consumption. Hence, the ranking list is WirelessUSB, Bluetooth, and WiFi, from the most preferred to the least preferred.
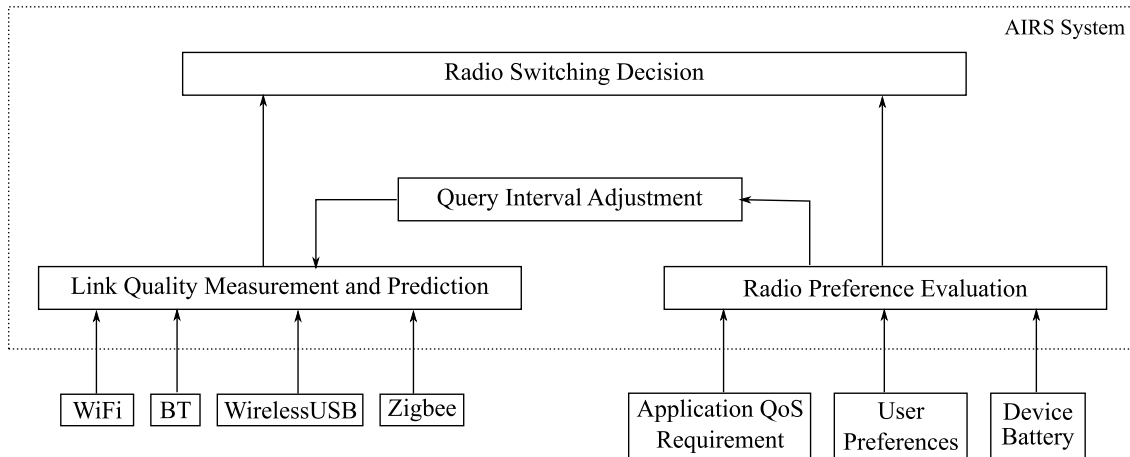
Figure 1.5: AIRS System

In "adaptive" mode, the radios are evaluated dynamically according to the device battery usage using an *Axiomatic Multi-transport Bargaining* algorithm [8]. The lower the battery, the more weight is placed on power consumption. Radios are first evaluated at the intra-device scope based on predefined utility functions, applying the *Utility Theorem*. Then, radio preference is negotiated at the inter-device scope applying *Nash's Axiomatic Bargaining* theory. "Social utility" is calculated for each radio, integrating connecting users' preferences in a mutually beneficial manner.

Results demonstrate that radio selection using the Axiomatic Multi-Radio Bargaining algorithm is fair and Pareto optimal. The radio selected can satisfy connecting users' preference equally, and no alternative selection can provider better connection than the selected one for all users.

This work is published at the 2006 IEEE Wireless Communications and Networking Conference:

Qiuyi Duan, Lei Wang, Charles D. Knutson, and Michael A. Goodrich. Axiomatic Multi-Transport Bargaining: A Quantitative Method for Dynamic Transport Selection in Heterogeneous Multi-Transport Wireless Environments. IEEE Wireless Communications and Networking Conference (WCNC), Las Vegas, NV, April 2006.

The full paper is presented in Chapter 2.

1.3.2 Link Quality Measurement and Prediction

The *Link Quality Measurement and Prediction* module is used to predict whether a given link can meet application requirements in a constantly changing environment. To predict future link quality, the device periodically measures radio status using one or more metrics, including throughput, delay, and jitter [10]. The device maintains a window of past measurements in FIFO order, and predicts future link quality based on these measurements using a Weighted Least Square Regression (WLSR) algorithm. The module calculates the "availability probability" to indicate the radio's future performance considering the QoS requirements on all concerned metrics.

There are many cases where we could decrease the frequency of periodic queries on some radios to save more system power. For example, if two devices are communicating on a preferred radio, and the connection is stable, the query interval on other radios can be reduced. We dynamically adjust the query interval in the *Efficient Query Interval* module using Fuzzy Logic control theory [9], taking into account the radio preference and the link quality prediction, thus conserving battery life.

Our results show that the model is able to achieve an accuracy above 90% when predicting link qualities under a variety of mobility and interference conditions. We are also able to reduce the overhead of this prediction scheme by dynamically increasing the link measurement interval, while preserving its accuracy. We compare the WLSR method to several other prediction methods and show that WLSR outperforms them significantly.

This work is published at the 2008 IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks, and an extended version is submitted to IEEE Transactions on Mobile Computing:

Qiuyi Duan, Lei Wang, Charles D. Knutson, and Daniel Zappala. Link Quality Prediction for Wireless Devices with Multiple Radios. IEEE International Sympo-

sium on a World of Wireless Mobile and Multimedia Networks (WoWMoM), Newport Beach, CA, June 2008.

Qiuyi Duan, Lei Wang, Charles D. Knutson, and Daniel Zappala. Efficient Link Quality Prediction for Wireless Devices with Multiple Radios. Submitted to IEEE Transactions on Mobile Computing.

The extended version is presented in Chapter 3.

1.3.3   Radio Switching Decision Making

The *Radio Switching Decision Making* module decides which radio should be selected and when a switch should be performed. This module enables autonomous and intelligent radio switching, according to the dynamic performance characteristics of each interface. The decision making module integrates radio preference and link quality prediction by applying the *Expected Utility* theorem, and utilize hysteresis and link verification parameters to reduce frequent radio switches.

Our results show that the communicating devices are able to dynamically choose the best available radio, while balancing throughput and power. The AIRS system is able to achieve high goodput (application level throughput) and long battery life as applied to radio switching management in a frequently changing mobile environment. A connection managed by AIRS has better performance in terms of transparency and adaptiveness, compared to other alternative algorithms. Frequent radio switches can be significantly decreased with the application of hysteresis, link verification, and expected utility calculation.

This work is accepted at the 2008 IEEE International Workshop on Heterogeneous Multi-Hop Wireless and Mobile Networks:

Qiuyi Duan, Lei Wang, Charles D. Knutson, and Daniel Zappala. Autonomous and Intelligent Radio Switching for Heterogeneous Wireless Networks.

IEEE International Workshop on Heterogeneous Multi-Hop Wireless and Mobile Networks (MHWMN), Atlanta, Georgia, September 2008.

The full paper is presented in Chapter 4.

Chapter 2

Axiomatic Multi-Radio Bargaining: A Quantitative Method for Dynamic Radio Selection in Heterogeneous Multi-Radio Wireless Environments

Radio selection mechanisms are designed to facilitate seamless connectivity in heterogeneous multi-radio environments, allowing access to the "best" available radio according to user requirements. Evaluating radio configurations dynamically according to the user's preferences and Quality of Service (QoS) requirements is a challenging task. This paper describes a quantitative approach that applies the Utility Theorem and Nash's Bargaining solution to heterogeneous wireless environments. The mathematical model presented generates and adjusts the radio preference list dynamically depending on the degree to which a radio satisfies user preferences and the application's QoS requirements. We incorporate a negotiation engine using the Axiomatic Multi-Radio Bargaining algorithm to integrate local and remote users' requirements in a mutually beneficial manner as devices are connected via a peer-to-peer link. The radio selection model discussed in this paper is computationally light with modest communication overhead, making it suitable for mobile devices.

2.1   Introduction

Heterogeneous multi-radio devices are increasingly common. Devices in these environments typically possess multiple physical transceivers, such as IrDA, Bluetooth,

IEEE 802.11b/g, and cellular. Such intra-device heterogeneity can be exploited to optimize connection quality by selecting the "best" available radio, according to the users' preferences and the application's Quality of Service (QoS) requirements.

Quality of Transport (QoT) is a protocol that manages session/application layer access to multiple radios in heterogeneous wireless environments (Figure 2.1). QoT functions as an intelligent layer inserted between the session and transport layers, facilitating dynamic, transparent and autonomous radio switching for multi-radio devices in order to provide the highest quality data transfer capability [27]. If the active radio is disrupted, QoT attempts to connect over an available but less desirable radio without disturbing the session (referred to as a *downgrade*). If QoT detects that a more preferred radio has become available, it attempts to connect over that radio in order to improve the link quality (referred to as an *upgrade*). In either case, the selected radio should be the optimal one among available radios, so that the user's requirements can be maximally satisfied during the communication with QoS commitment.

The radio selection module in the QoT framework is designed to provide and dynamically maintain a radio preference list according to users' preferences and applications' QoS requirements. As a reference for QoT radio switching, such radio preference information allows traffic from session layers to be routed over the "best" available radio at any given time.

The rest of the paper is organized as follows. First, we introduce related work in Section 2.2. Second, we present a radio selection overview in Section 2.3, illustrating the role and functionality of our radio selection model. Next, descriptive criteria selection and user interface are discussed in Section 2.4 and Section 2.5 respectively. In Section 2.6, the radio selection model is introduced in detail. Dynamic preference adjustment is described in Section 2.7. Finally, conclusions are presented in Section 2.8.
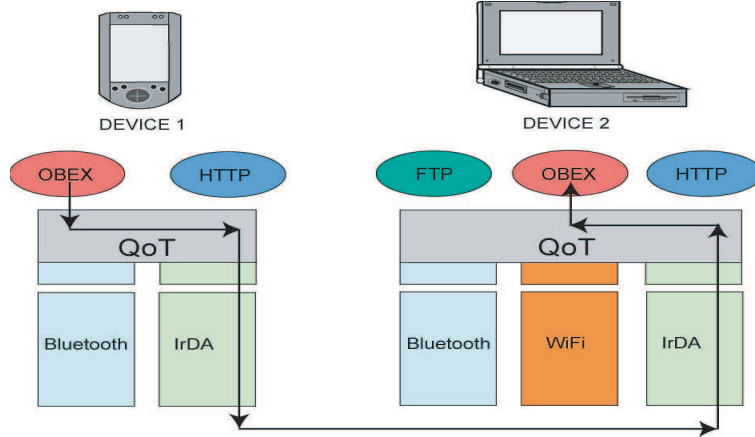
Figure 2.1: Data Exchange with QoT in Multi-Radio Environment

2.2 Related Work

A number of research projects have examined heterogeneous connection capabilities. The BARWAN project at UC Berkeley explored the use of vertical handoffs in wireless overlay networks as a mechanism for intelligently and dynamically maintaining an active TCP/IP connection to a network infrastructure [45]. The model assumed that networks with the smallest coverage provided the highest throughput, and hence were the "best." Such network selection may be inappropriate without considering user preferences and other relevant criteria.

The network selection mechanism for the BARWAN project was improved in [51], in which the author proposed a policy-based decision making scheme that relied on user input to determine tradeoffs between network cost, performance, and power consumption. However, no performance feedback information was provided, and the user was not enabled to adjust requirements after the initial setting. This policy-based approach was further improved in [58] by considering multiple active services, but the feedback problem still exists.

A segment selection algorithm based on fuzzy multiple objective decision making is discussed in [5]. Their model considers the trade-off between cost and quality in order to make a choice between terrestrial and satellite networks for a connection.

Figure 2.2: QoT Brain – Radio Switching Decision Making Model

The MosquitoNet [2] project at Stanford University was aimed at providing continuous Internet connectivity to mobile hosts through a mechanism of switching seamlessly between different network devices to take advantage of available connectivity, whether wired or wireless. Their work concentrates on Internet connectivity optimization, presuming that Internet access is the only essential usage model. The mechanism by which a connection is chosen is not discussed. The quality of connection and user's preferences are not considered in this model.

A Prioritized Soft Constraint Satisfaction (PSCS) scheme is proposed in [12] to select the "best" radio in a dynamic wireless radio switching system based on a user-established range of preferences and priority for criteria such as speed, power, range and cost. QoS requirements and mobility issues are not considered in this model. The user interface in this model is relatively complicated, requiring significant user involvement, and the final radio selection is decided almost entirely based on the user's inputs. Non-technical users of the PCSC interface may not understand the meaning of specific terms, or may lack the ability to intelligently specify such criteria. Such weakness may significantly degrade the applicability of the model in practice.

The PSCS radio selection model is extended in [11] for situations in which devices connect over a peer-to-peer link. A negotiation engine is added to generate

16

a preference list that is favorable to both users. However, QoS, mobility, and the complexity and dependency problems are inherited, making the model somewhat weak with respect to accuracy and applicability.

While leveraging some of the strengths of PSCS, the model discussed in this paper generates and adjusts the radio selection list based on the application's QoS requirement and the user's preferences. Performance feedback information is provided for the user's reference and for dynamic adjustment. This model is computationally light with modest communication overhead, and is less dependent on the user's inputs. Besides the intra-device radio evaluation and inter-device preference negotiation scenarios presented in this paper, the proposed model may be applicable to other multi-criteria selection problems in ad-hoc networks.

2.3   Radio Selection Overview

In order to better understand our model, we present an overview in this section, showing the performing environment, functional role and specific functionality of our radio selection model.

2.3.1   Mobility Issue

Connectivity quality (such as packet loss, error rate, latency, and jitter) changes as a user moves. For example, the signal quality degrades as a user leaves the service range. Hence, user mobility may impact radio selection.

However, computation and communication complexity increase significantly when mobility is considered. Radio selection models require periodic assessment of signal quality for all potential radios, so that the preference list may be updated based on this information. Devices connected over a peer-to-peer link must renegotiate their preference information constantly. Therefore, for complexity and overhead

considerations, we separate the measurement of such highly dynamic factors into an independent module.

2.3.2 Environmental Role of Radio Selection

The QoT Brain serves as a "trusted advisor" in the QoT framework. It makes intelligent radio switching decisions autonomously, so that the "best" available radio may be selected at any given time.

As illustrated in Figure 2.2, the QoT Brain is composed of four sub-modules. The Radio Preference Evaluation module is designed to produce and dynamically maintain a preference list, disclosing the desirability of each radio according to user preferences and the application's QoS requirement. The radio preference list is updated whenever a change of preference settings is detected.

The Link Quality Measurement and Prediction module is used to provide accurate status information for all supported radios via periodic radio performance measurements and quality assessments, in order to minimize the probability of incorrect radio switching. This module considers the QoS parameters that are not involved in the Radio Preference Evaluation module, such as packet loss, latency, and jitter. Further, it cooperates with the Radio Preference Evaluation module, providing requisite information for intelligent radio switching decision making.

Query Interval Adjustment module is designed to reduce system overhead by determining an efficient assessment query interval based on the radio desirability and its status records.

The Radio Switching Decision Making module integrates both radio desirability and status information, selects the "best" "stably available" radio at that given moment, and makes the final switching decision.

These four modules are interrelated and function in a cooperative manner. The system shown in Figure 2.2 makes it possible to keep the user connected seamlessly

over the "best" available radio at any given time. The radio selection model discussed in this paper achieves the functionality of the Radio Preference Evaluation module in Figure 2.2.

2.4 Descriptive Criteria

QoT-enabled devices rank and select radios transparently without the user's explicit involvement. The user may express specific preferences via a set of descriptive criteria provided by the system, and may make adjustments where desired. Our radio selection model collects such preference settings and translates them into a radio preference list.

Typical descriptive criteria include data rate, power consumption, signal range, service charge, signal quality (latency and reliability), jitter, etc. [12] [5] [51]. We consider two criteria to be critical in our radio selection model: data rate and power consumption.

- **Data rate** – User preferences may vary with respect to this criterion. Some usage models call for extremely high speed, while others may be satisfied as long as the throughput is sufficient to satisfy the application. The user's preference setting for data rate would be considered as one of radio selection rules during decision making.

- **Power consumption** – System power is a critical resource for mobile devices. Most users are concerned with the battery life of their mobile devices and some prefer power-efficient services with less data efficiency in order to achieve longer battery duration. Therefore, power consumption is selected as an imperative criterion for radio selection. Users may set specific preferences on this criterion to achieve desirable services.

This criteria set is simple and straightforward, while providing necessary preference information for decision making. Other criteria that we considered but did not include in this model are:

- **Service charge** – Most short-range wireless services do not require usage fees (for example, IrDA and Bluetooth). Wireless LANs are typically free of charge [51], although the proliferation of subscriber-based WiFi hot spots is changing that somewhat. Still, such hot spots generally operate on a flat-fee basis, rather than a per-byte usage charge. Assuming that a regular flat fee is charged for a user subscription, there is no particular cost savings in avoiding the service. Hence, our current model chooses to ignore service charge as a selection criterion.

- **Service range** – It is not necessary to consider signal range for mobile communications. For a certain wireless service, the device is either within the signal range or not. Such radio availability variations are considered under the mobility criterion through dynamic radio quality trackings. Furthermore, when the device is under the coverage of a preferred service according to the data rate and power consumption settings, a user's specific restrictions on signal range may even cause incorrect radio selections. Suppose that the user sets the preferred threshold at more than 1 meter, implying that IrDA is not a desirable candidate. If two devices communicate within 1 meter, and both users care a great deal about power consumption. IrDA is the optimal choice. However, it would not be selected due to the range restriction.

- **Mobility** – Signal quality[1] changes as a user moves. As discussed in section 2.3, even though mobility is an issue that may impact radio selection, due to its

---

[1]Signal quality, including packet loss, latency, and jitter etc., is not considered as a selection criterion, since it is difficult for the user to provide specific settings. QoS requirements on these metrics are considered in the mobility module.

computation and communication overhead, functionality for mobility detection and radio availability maintenance are combined in a separate module.

2.5   User Interface

The radio selection user interface supports two phases: *connection mode selection* and *preference adjustment.*

The interface for connection mode selection, represented in Fig. 2.3, allows the user to indicate the most important criterion. Selection of high-speed mode indicates that the user is generally more concerned with throughput than with power consumption. Hence, services with high data rate are preferred. The radio with the highest data rate is initially prioritized above other radios. Selection of power-efficient mode demonstrates that the user is more concerned with the device's battery life than with communication speed. Radios with low power consumption are desirable so long as such radios meet the application's throughput requirement. Within all options that are qualified on throughput, the radio with the lowest power consumption is initially prioritized.

Clearly, connection mode selection is not enough. Sometimes, even though a user chooses high-speed mode, the fastest radio may not be the one that is actually preferred. Thus, mode selection suggests only that the concern for throughput is greater than the concern for power consumption. We need to find a more favorable setting based on the user's specific desires on two criteria. Such information is obtained via the preference adjustment interface, which is illustrated in Figure 2.4.

Based on the user's connection mode selection at phase one, two sliders are set to the corresponding positions according to the performance features of the prioritized radio, with data rate and power consumption marked below the axes. Other achievable performance options for data rate and power consumption are distributed along the two axes respectively. If the current setting is satisfactory according to the

Figure 2.3: User Interface: Connection Mode Selection
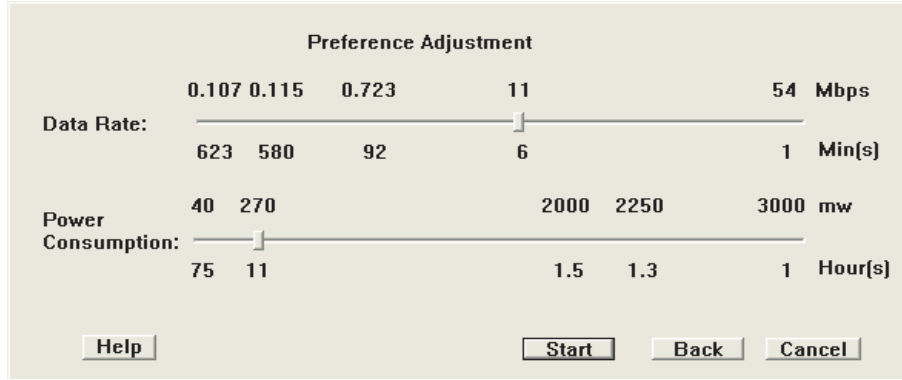
Figure 2.4: User Interface: Preference Adjustment

user's desire (high speed or power saving preference), communication can start immediately. Otherwise, the user can further adjust the setting by moving the slider of the corresponding criterion to other preferred options. For each criterion, the position of the slider indicates the user's preference setting.

Applications have specific Quality of Service (QoS) requirements. Such QoS information may be acquired via cross-layer communication techniques. For example, the application layer could communicate to other layers the application's QoS needs, including required throughput [39]. In order to guarantee a reasonable communication quality, we set the minimum data rate threshold as the QoS throughput requirement. Radios with data rate lower than the *minimum threshold* are incapable of completing the application, and thus should not be presented as options. Data rate options corresponding to such radios are locked against selection on the interface. The minimum threshold changes as the application's requirement changes.

22

Criteria are interrelated. A preference setting on one criterion may potentially restrict the selection on the other criterion. For example, in order to provide a data rate higher than the minimum throughput threshold, power consumption may not be made arbitrarily low. As discussed in [12], we use Shannon's Law to determine the minimum power threshold based on the minimum data rate requirement. Any option below the *minimum threshold* on the power criterion is incapable of meeting the application throughput requirement, and is locked against selection.

For the user's convenience, data rate is also presented as a time representation, namely the elapsed time required to download a one-hour length movie (typically 500 MB). The power consumption is also transformed into time format, namely the approximate battery life based upon the execution of the target application over the preferred radio. The system power information could also be gathered using cross-layer techniques such as those introduced in [39] [43]. The remaining battery life in our model has a different definition. The purpose of such a transformation is to provide the user an explicit perception of the power consumption features of all radios, instead of an accurate prediction of exactly how long the battery could last considering all tasks and system consumptions. Estimation methods can be found in [53] [40] [38].

2.6 Dynamic Radio Selection Model

The radio selection mechanism is designed to facilitate seamless connectivity while respecting user preferences. The model discussed in this paper dynamically maintains a radio preference list by balancing users' preferences on data rate and power consumption, so that the "best" available radio will be selected at any given time, according to user requirements.

2.6.1 Terminology

We denote $r_{\min}$ as the minimum data rate requirement, which is set based on the QoS throughput requirement gathered from the application layer. The value of $r_{\min}$ adjusts dynamically as the application's requirement changes.

We let $r_{\max}$ represent the maximum data rate, which is the highest data rate that radios can support. The value of $r_{\max}$ is set based on system information, and does not change unless a new transceiver with higher throughput is added.

The minimum and maximum power consumption thresholds are similarly defined, with $p_{\min}$ representing the minimum power threshold (which is set based on $r_{\min}$ using Shannon's Law) and $p_{\max}$ representing the maximum power threshold, which is the highest power consumption regardless of the availability status of radio, according to the system information.

r_{pref} is used to represent the data rate that the user prefers, gathered from the user interface (see Section 2.5). Any data rate in $[r_{\text{pref}}, r_{\max}]$ meets the user's preference for throughput. Any data rate in $[r_{\min}, r_{\text{pref}})$ is capable of satisfying the application, even though it is not fully desirable.

Similarly, we obtain the user's preference on power consumption from the interface and denote it as p_{pref}. Power consumption in $[p_{\min}, p_{\text{pref}}]$ meets user's preference, while power consumption in $(p_{\text{pref}}, p_{\max}]$ is an acceptable option.

$\mathcal{R} = \{R_1, R_2, ..., R_i, ..., R_n\}$ denotes the set of all radios. n is the number of radios that the device possesses. R_i represents radio i.

2.6.2 Intra-Device Radio Selection

The Axiomatic Multi-Radio Bargaining algorithm comprises two phases: intra-device radio preference evaluation and inter-device radio preference negotiation. We first

consider phase one[2], in which the radio is selected according to the local user's preference and the application's QoS requirements.

We apply the Utility Theorem [42] [36] to reflect variations in user's requirements. As the Utility Theorem states, if an agent (user) has a preference relation that satisfies the axioms of preference[3], then a real-valued utility function[4] can be constructed that reflects this preference relation such that the following hold:

$$U(R_i) > U(R_j), \text{ if } R_i \text{ is preferred to } R_j$$

$$U(R_i) = U(R_j), \text{ if } R_i \text{ and } R_j \text{ are equally preferred}$$

We first set a radio's utility based on the data rate criterion. Radios with data rate less than $r_{\min}$ do not qualify for the requested application. Hence user input is indifferent on those options and 0 is assigned as the data rate utility for all incapable radios. Radios within the range $[r_{\text{pref}}, r_{\max}]$ are equally desirable according to the user's requirement. We set their data rate utility as 9. Radios in $[r_{\min}, r_{\text{pref}})$ are capable with varying degrees of satisfaction. Their utility is given proportionally in range [1,9) according to their data rate, as shown in Equation 2.1.

$$\frac{r_{R_i} - r_{\min}}{r_{\text{pref}} - r_{\min}} = \frac{U_{rate}(R_i) - U(r_{\min})}{U(r_{\text{pref}}) - U(r_{\min})}$$

$$\Rightarrow U_{rate}(R_i) = \frac{r_{R_i} - r_{\min}}{r_{\text{pref}} - r_{\min}} * 8 + 1 \tag{2.1}$$

---

[2]The radio selection method at phase one also applies to a scenario in which the mobile device communicates with an infrastructure, such as a network access point.

[3]The axioms of preference refer to the properties of orderability, transitivity, continuity, substitutability, monotonicity, and decomposability. Details can be found in [42].

[4]The utility function is unique up to a positive affine transformation, which means that if a function $U()$ satisfies the axioms of preference for a particular agent, then so does $\alpha U() + \beta$ for $\alpha > 0$ and $\beta \in R$.
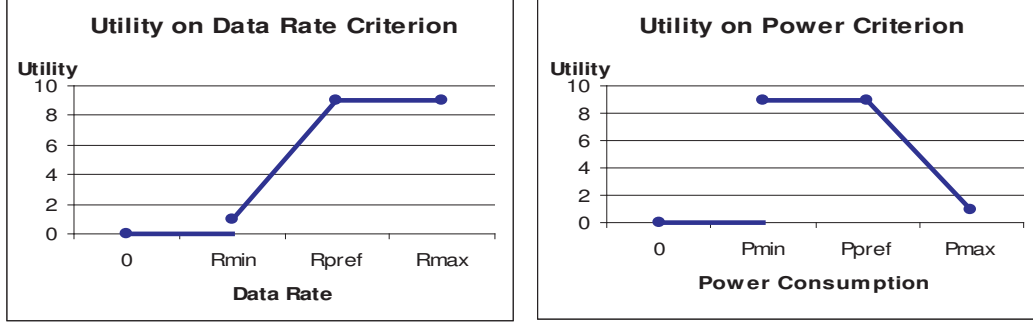
Figure 2.5: Relationship of Utility and Descriptive Criteria

Therefore, we define the utility function for the data rate criterion as Equation 2.2 to reflect the user's preferences on available radios. The relationship of utility and data rate is further illustrated in Figure 2.5.

$$
U_{rate}(R_i) =
\begin{cases}
0 & r_{R_i} < r_{\min} \\[2mm]
\frac{r_{R_i} - r_{\min}}{r_{\mathrm{pref}} - r_{\min}} * 8 + 1 & r_{R_i} \in [r_{\min}, r_{\mathrm{pref}}) \\[2mm]
9 & r_{R_i} \in [r_{\mathrm{pref}}, r_{\max}]
\end{cases}
\tag{2.2}
$$

Similarly, we create another real-valued function via Equation 2.3 to represent a user's well-behaved preferences on the power consumption criterion. The utility setting is also illustrated in Figure 2.5.

$$
U_{power}(R_i) =
\begin{cases}
0 & p_{R_i} < p_{\min} \\[2mm]
9 & p_{R_i} \in [p_{\min}, p_{\mathrm{pref}}] \\[2mm]
\frac{p_{\max} - p_{R_i}}{p_{\max} - p_{\mathrm{pref}}} * 8 + 1 & p_{R_i} \in (p_{\mathrm{pref}}, p_{\max}]
\end{cases}
\tag{2.3}
$$

For simplicity, we use 0 and real numbers from 1 to 9 to represent the utilities. Any real number will do, provided that the number assigned to the most preferred option is higher than the number assigned to the least preferred option and connected devices use the same scale.

Now, according to the user's preferences, each radio is evaluated by two independent criteria, data rate and power consumption, via the Equation 2.2 and 2.3 respectively. However, these two criteria are not equally important from the user's standpoint. We need to balance their degrees of importance when calculating the radio's overall utility. We do this by taking a weighted combination of the utilities of data rate and power consumption using a weight that matches the user's connection mode selection, where we can determine the criterion that is more important. In our model, this criterion is weighted 10 times more heavily than the less important criterion. The overall utility for each radio is calculated using Equation 2.4, where the value of the more important criterion is put to the tens position to represent its stronger influence on the outcome.

$$
U_{overall}(R_i) = \begin{cases} U_{rate}(R_i) * 10 + U_{power}(R_i) & \text{high-speed} \\ U_{power}(R_i) * 10 + U_{rate}(R_i) & \text{power-efficient} \end{cases} \tag{2.4}
$$

It is possible for us to choose other weight values to reflect the importance of the more important criterion. Experiments were conducted to determine the effects of the weight selection in Equation 2.4. Although not shown because of space limitations, we found that smaller weights were not strong enough to reflect the relative importance of the criterion in the utility, and larger weights neglected the less important criterion too much. Since the utility range for each criterion is $[0, 9]$, weights that are greater than 9 are sufficient to disclose the decisive influence of the more important criterion and provide similar balanced information. We empirically selected 10 as the balancing weight in our model.

The radio with higher overall utility ranks higher in the final radio preference list and is more preferred by the user. As implied in Equation 2.4, the radio with the higher utility on the more important criterion better meets the user's preference and QoS requirements. Hence, it is assigned a higher overall score and is ordered higher for

selection. Radios with the same utility evaluations on the more important criterion are ordered based on their utilities on the other criterion. The radio with a satisfying performance on the less important criterion may not be desirable – its probability of being selected depends on its performance on the more important criterion. The overall utility of radios that are unqualified on either criterion ($U_{rate}(R_i) = 0$ or $U_{power}(R_i) = 0$) is set to 0. Radio with a lower power consumption is preferred if there is a tie.

The overall utility explicitly represents the user's preference with QoS consideration. The radio is selected based on the degree to which it satisfies user requirements.

2.6.3 Inter-Device Radio Selection

As devices are connected over a peer-to-peer link with differing preferences on connection quality — a radio selection preferred by one user may not satisfy the other user's requirements. A radio preference negotiation mechanism is required to integrate both local and remote users' preferences. In this section, we consider the second phase of Axiomatic Multi-Radio Bargaining algorithm — inter-device preference negotiation.

Traditional turn-taking negotiation incorporates significant communication overhead and inherits potential deadlock risks. In order to solve this problem, QoT employs a third party arbitration handled through the arbitration engine on the primary device [11]. The engine works like a benevolent court system, which can fairly enforce the agreement reached by the agents.

The arbitration engine first gathers preference information from local and remote devices on common radios. Only the radio's overall utility is required, since it includes all relevant information, such as the connection mode selection, the application QoS requirement, and the user's specific preferences on each descriptive criterion. Based on such preference information, we apply Nash's Axiomatic Bargaining Solu-

tion [52] [36] [35] to evaluate all radio configurations and generate the commonly shared social preference list.

Nash's Axiomatic Bargaining is a mechanism that sets up the rules of the negotiation so that socially "good" things happen given any users' utilities. A radio that is mutually beneficial to both users is preferred. The better it meets both users' requirements, the higher it is listed.

As Nash's Axiomatic Bargain algorithm defines, there is a special outcome, called a fall-back solution, that can result if negotiation breaks down. As radio selection negotiation fails, no radio can satisfy both users' requirements simultaneously, and no connection can be set up. Therefore, we set the utility of fb solution as $U_{user1}(fb) = U_{user2}(fb) = 0$.

According to Nash's Axiomatic Bargaining algorithm, the best negotiation solution is the one that maximizes the benefits of the two agents, as demonstrated in Equation 2.5:

$$R_{best} = \arg\max_{R_i \in \mathcal{R}}[U_{user1}(R_i) - U_{user1}(fb)][U_{user2}(R_i) - U_{user2}(fb)] \qquad (2.5)$$

Since $U_{user1}(fb) = U_{user2}(fb) = 0$, Equation 2.5 can be simplified into Equation 2.6.

$$R_{best} = \arg\max_{R_i \in \mathcal{R}}[U_{user1}(R_i)][U_{user2}(R_i)] \qquad (2.6)$$

Applying Nash's Axiomatic Bargaining algorithm, we define a real-valued function, namely the Axiomatic Multi-Radio Bargaining algorithm, to calculate the social

utility for each radio using Equation 2.8 and make a social selection using Equation 2.7.

$$R_{best} = \arg\max_{R_i \in \mathcal{R}} U_{social}(R_i) \tag{2.7}$$

$$U_{social}(R_i) = U_{user1}(R_i) * U_{user2}(R_i) \tag{2.8}$$

Preferences of both local and remote users are equally considered during the negotiation. A radio's social utility is calculated based on its overall utility evaluated by the two connecting devices. The higher the utility a radio gets, the better it meets the users' preferences simultaneously. The radio with the highest utility value is the best negotiation solution as defined by the Axiomatic Bargaining algorithm.

Social selection should be fair with respect to both users' requirements. A fair solution in our model implies two aspects. One is that the negotiation process is unbiased, meaning that the connecting users' preferences are impartially considered. The other is that the selected solution is Pareto optimal, meaning that no alternative solution is better than the selected one for both users. A radio that simultaneously satisfies both users' requirements in a mutually beneficial manner is preferred.

Nash's Axiomatic Bargaining algorithm is fair according to its axioms. It is Pareto optimal, symmetric, independent of the utility scales of two agents, and independent of irrelevant alternatives [35]. The Axiomatic Multi-Radio Bargaining algorithm inherits such features and is able to provide a fair solution that satisfies the above fairness axioms. Our algorithm allows the "best" available radio to be selected via an impartial consideration of the preferences from both sides.

The social radio selection list is ordered from the radio with the highest social utility to the one with the lowest value. A radio with lower power consumption is preferred if there is a tie. If the active radio becomes unavailable, connecting

Transport	Data Rate(Mbps)	Power (mw)
IrDA (SIR)	0.115	40
Bluetooth	0.723	270
IEEE 802.11b	11	2250
IEEE 802.11a	54	2000
GPRS	0.107	3000

Figure 2.6: Performance Scenario Data of All Radio Configurations

devices could switch to the next available radio, which is currently the "best" radio among all available options. As a more preferred radio becomes available, connecting devices could upgrade to that radio to achieve a more favorable connectivity quality. Therefore, the most preferred available radio configuration could be selected at any given time, providing a mutually beneficial connectivity.

This radio selection model performs with reasonable computation and communication overhead because of its simplicity, making it suitable for mobile devices.

2.6.4 Performance Evaluation

We employed two QoT-enabled devices in our experiments, connecting via a peer-to-peer link in order to exchange files using the OBEX protocol. The commonly shared radios are IrDA (SIR), Bluetooth, IEEE 802.11b/a, and GPRS. We vary users' inputs to all possible mode and preference settings, and compared the performance of our algorithm with three other strategies: Always Power Efficient, Always High Speed, and Always the Largest Coverage. The minimum and maximum thresholds for data rate and power consumption were gathered from the application layer and system information using cross-layer techniques. Scenario data, shown in Figure 2.6, were collected from each radio's specification.

We use two metrics to evaluate the performances of various selection algorithms: desirability and fairness.

- The **desirability** metric reflects the degree to which a selected radio satisfies the users' requirements. We use a radio's overall utility for intra-device selection
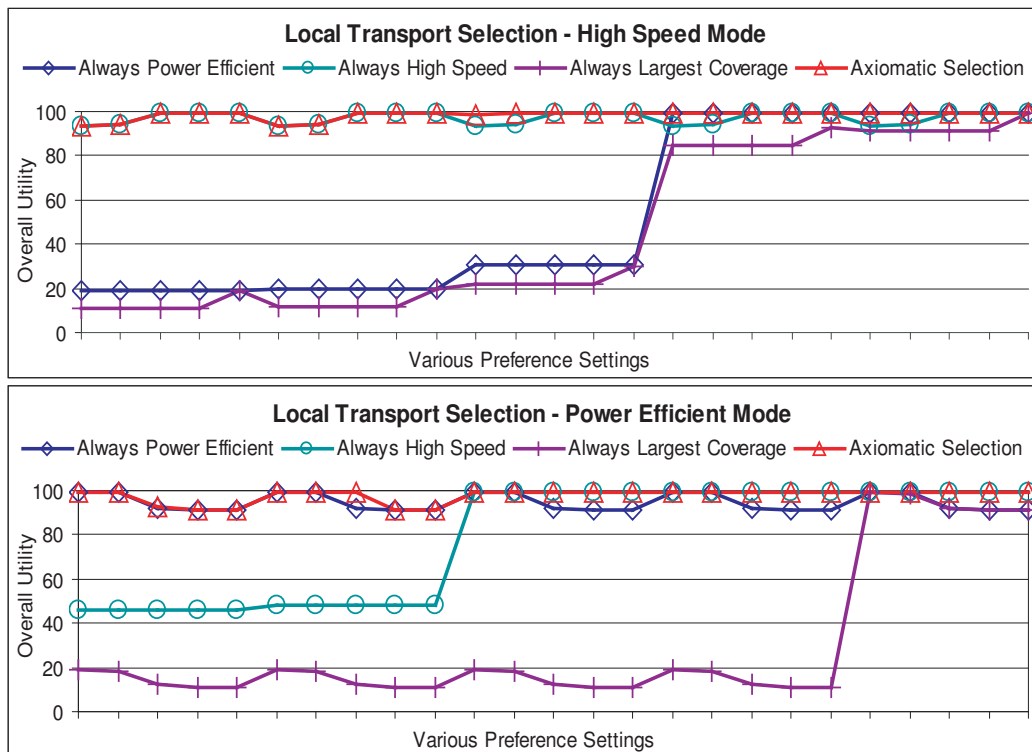
Figure 2.7: Intra-Device Radio Selection Comparisons on Various Preference Settings — Desirability of the Selection

evaluation, and the average overall utility of the radio over thousands of different preference settings from both users for performance comparison on inter-device selection.

- The **fairness** metric indicates the degree of impartiality with which users' preferences are considered during negotiation. Like desirability, we use the average overall utility of the radio over all possible preference settings from both users for such performance evaluation.

Figure 2.7 compares our algorithm with the other three strategies for local radio selection using the desirability metric[5]. As the result demonstrates, the radio selected using our algorithm is consistently more or equally desirable when compared

---

[5]The X-axis represents all possible preference settings under a specific connection mode selection. Patterns of the lines in Figure 2.7 depend on the experiment order on various settings, and are not essential to the motivations that underlie the research presented here.
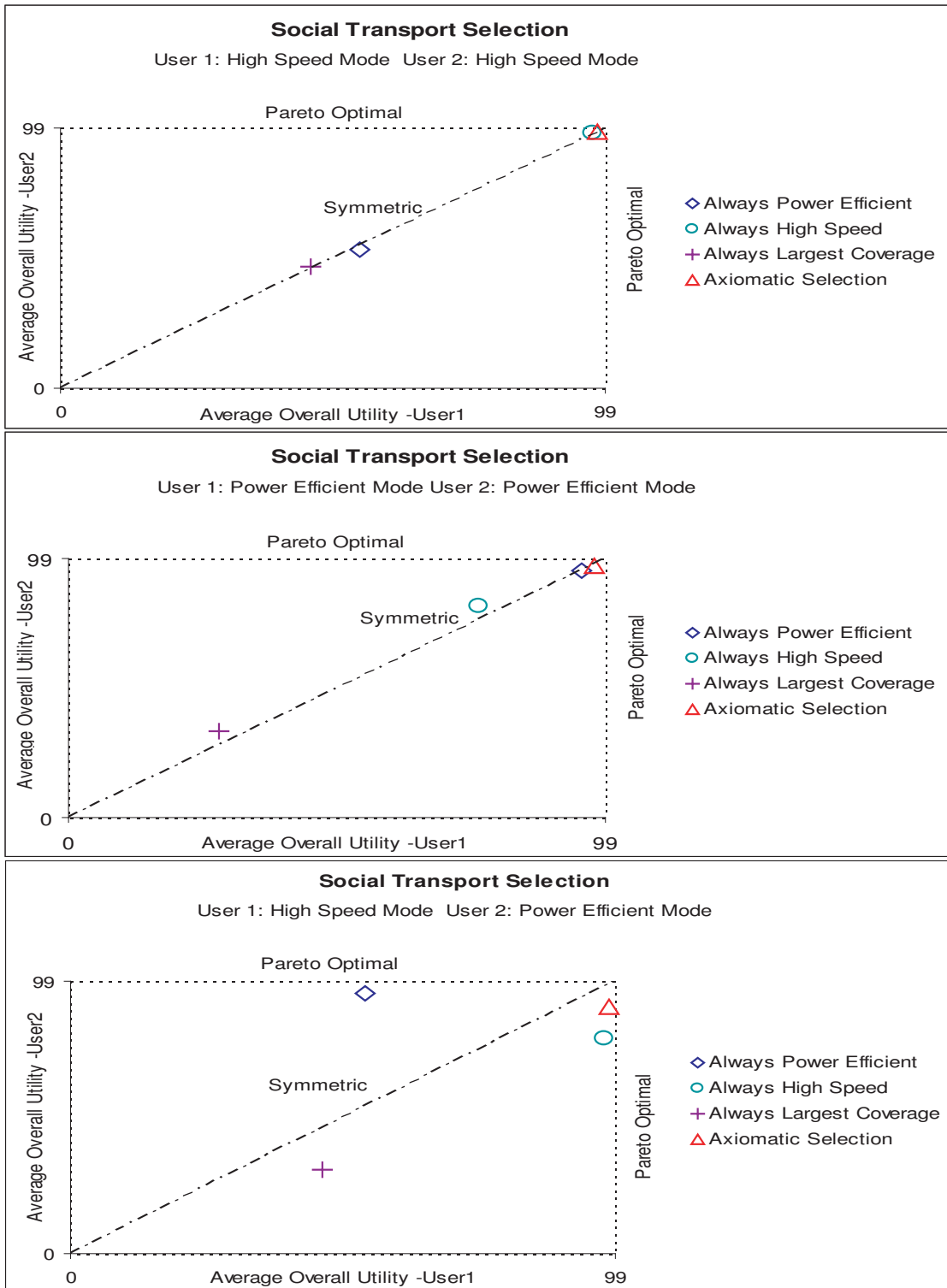
Figure 2.8: Inter-Device Radio Selection Comparisons on Various Mode and Preference Settings — Fairness and Desirability of the Selection

to other strategies. The "best" available radio could be selected following our radio preference list in either High-Speed or Power-Efficient mode.

Figure 2.8 compares the fairness and desirability of our algorithm with the other three strategies in three cases: both users care about speed; both users care about power consumption; users have different connection mode preferences. The result of each case is the average overall utility of the selected radio over all possible preference setting combinations with respect to that specific mode selection.

According to the line of symmetry, Figure 2.8 indicates that the radio selected using the Axiomatic Multi-Radio Algorithm is consistently fairer than the selection using the other strategies in all cases. When two users make the same connection mode selections, they have similar evaluations on the commonly shared radios. Hence, radio selection of any strategy meets the connecting users' requirements with a similar degree of satisfaction, either desirable to both sides or undesirable to both sides. As the figure shows, the radio selected using our algorithm is closer to the line of symmetry compared to the selection of other strategies, even though the advantage is not significant. When two users make different connection selections, their preferences may be significantly different. The Axiomatic Multi-Radio algorithm dynamically evaluates all shared radios according to both users' requirements and selects the option that is mutually beneficial to both sides. As the figure shows, the radio selected using our algorithm is much closer to the line of symmetry compared to the selection of other strategies.

Based on the Pareto optimal lines, Figure 2.8 demonstrates that the radio selection using the Axiomatic Multiple-Radio Bargaining algorithm is consistently more desirable when compared to the other three strategies in any case with all different preference setting combinations from the connecting peers. The Always High Speed and Always Power Efficient strategies only perform well when both users concern speed and both concern power consumption respectively. The radio selected using

the Axiomatic Multi-Radio algorithm is consistently the most desirable one compared to other strategies, because of its capability of choosing the Pareto optimal option through dynamic evaluations. The advantage is especially obvious when connecting users have different mode selections.

As the experimental results illustrate, our algorithm allows the "best" available radio in terms of fairness and desirability to be chosen, so that both users' connectivity requirements are mutually benefited during the communication.

2.7 Dynamic Preference Adjustments

The preference adjustment interface (Figure 2.4) also provides feedback information on connection quality. Positions of the two sliders are dynamically adjusted according to the current radio configuration, disclosing present connection performance information.

The user can adjust the preference setting at any time as requirements change, by moving the corresponding slider to the desired option or by changing the connection mode selection. Relevant variables, such as r_{pref}, p_{pref}, and connection mode, change accordingly. The radio selection model reevaluates all radio configurations based on the updated information, and generates a preference list according to the new preference setting. A QoT enabled device would then switch to the most desired available radio, so that the configuration selected is always favorable to the user's requirement.

The radio preference list also changes as the application's QoS requirement changes or as the system resources change.

As introduced in section 2.5, minimum thresholds of the two criteria are set based on an application's QoS requirements. As an application executes, QoS requirements change accordingly. With cross-layer information sharing, QoT could detect such changes and adjust $r_{\min}$ and $p_{\min}$ dynamically according to the new requirement.

Maximum thresholds of the two descriptive criteria, $r_{\max}$ and $p_{\max}$, are set based on system resource information. Their values do not changed unless a new transceiver with an even higher data rate or power consumption feature is added.

The radio selection model reevaluates all supported radios based on the newly updated criteria information. The social preference list is also updated through renegotiation as devices are connected over a peer-to-peer link.

2.8 Conclusion

This paper presents a dynamic radio selection model that applies the Utility Theorem and the Axiomatic Bargaining algorithm to heterogeneous wireless environments. The mathematical model generates and adjusts a preference list dynamically, facilitating seamless connectivity with user preferred quality. Multiple users' requirements are satisfied in a mutually beneficial manner using Axiomatic Multi-Radio Bargaining algorithm. The model is suitable for mobile devices with modest computation and communication overhead.

Future work in radio selection decision making could be focused on multiple radio utilization, such as inverse multiplexing, making the connectivity more favorable according to user preference.

Chapter 3

Efficient Link Quality Prediction for Wireless Devices with Multiple Radios

With the abundance of wireless devices available today, it is increasingly common for devices to support multiple radios, for example both WiFi and Bluetooth. Communication between these devices ought to be as simple as possible; they should be able to seamlessly switch between different radios and network stacks on the fly in order to better serve the user. To make this a possibility, we consider the challenging problem of predicting link quality – in terms of throughput, delay, and jitter – in a changing mobile environment. In this paper we present a link quality prediction algorithm that uses Weighted Least Square Regression to predict future availability based on past measurements of link quality. We use a simulation study to show that our prediction algorithm outperforms several alternatives, and is able to achieve an accuracy above 90% under a variety of mobility and interference conditions. We also show that our algorithm can significantly reduce its power consumption, without sacrificing accuracy, by increasing the link measurement interval. Finally, we demonstrate how a multi-radio system can improve throughput and power consumption by using our prediction algorithm to dynamically select the best available radio.

3.1 Introduction

Wireless devices ought to make it easier for users to communicate with each other. Complicating this vision, however, is the reality that no single wireless technology dominates the market nor provides the desired functionality in all situations. Cellular technology provides coverage over a wide area, but phone manufacturers are adding WiFi interfaces so that users can browse the web at a WiFi hotspot, with lower connection charges and possibly higher speeds. Likewise, laptops and cellphones, in addition to WiFi or cellular interfaces, have Bluetooth interfaces for exchanging data directly with other devices or peripherals, when other network interfaces may be unavailable, too cumbersome, or consume too much power. It is likely that wireless technologies will continue to proliferate and that devices will continue to support multiple radios and network stacks.

Because of this reality, we envision that devices ought to be able to seamlessly switch between available network connections on the fly in order to provide access to available services. For example, if a person wants to transfer images from a cellphone to a laptop, the devices should cooperate to make this happen however they can, regardless of whether the transfer utilizes a Bluetooth or WiFi connection. Furthermore, as the availability or quality of a connection changes due to the activity of other nodes or interference from other devices, devices should cooperate to switch to the best available interface, taking into account power and performance tradeoffs. In other words, wireless devices should exploit their heterogeneity in order to provide better service to end users. This kind of communication should "just work" rather than requiring the user to be involved.

In pursuit of this vision, we have developed a device architecture for seamlessly switching between available wireless interfaces [27], and we have examined several components of this architecture [3, 12, 8]. In this paper, we focus on one aspect of radio selection, the ability to predict the availability and performance characteristics

of each wireless interface [10]. This is a key issue, since in many cases, wireless devices have several different radios to choose from, and need some guidance as to which interface is likely to satisfy the application in the near future.

Predicting link availability has received significant attention in ad hoc wireless networks, where it is principally used to help routing protocols provide stable routes [23,17]. In this work, nodes use a single WiFi radio and cooperate to maintain network connectivity. Each node tries to predict the probability that a link to its neighbor will continue to be available for some time into the future. The routing protocol then uses this metric to compute routes that will remain available for the longest time; this has shown to be more effective than using shortest path routing.

The issues we face in designing for heterogeneous wireless devices differ from this previous work in several fundamental ways. The main difference is that we are interested in predicting link quality, rather than simply link availability. With a single interface and a network of homogeneous devices, maintaining connectivity is most important. However, with multiple available interfaces between two communicating devices, our goal is to choose the interface that can best meet application requirements. Accordingly, we try to predict whether a link will meet the throughput, delay, and jitter requirements of a particular application.

Another difference from previous work is that we consider devices in which each interface may potentially have its own network stack. For example, WiFi interfaces typically use a TCP/IP stack, but Bluetooth interfaces have their own stack. This means that we must devise a general algorithm that does not depend on a particular technology. We must also handle interference as a common occurrence, since different technologies may share the same frequencies.

The challenge in predicting the quality of a wireless link is that fluctuations occur due to mobility and contention from other wireless devices. In this paper, we predict future link quality using a sliding window of previous, periodic measurements.

We devise a prediction algorithm based on the Weighted Least Square Regression (WLSR) algorithm and determine the proper weighting of current versus past measurements using a full factorial simulation. We then reduce power consumption by using fuzzy logic to dynamically adjust to the frequency of the periodic measurements, based on radio status.

We use a simulation study to show that our algorithm is able to accurately and efficiently predict link quality under a variety of mobility and interference conditions. We compare prediction based on WLSR to several alternatives, as well as to an ideal algorithm that uses knowledge of the future to make predictions. Prediction based on WLSR outperforms the alternative algorithms, while achieving an accuracy above 90%. We also show that using dynamic link measurements reduces the overhead of the algorithm significantly, while preserving its accuracy. Finally, we illustrate the utility of our link quality prediction algorithm by using it to dynamically select the best radio for communication between two mobile devices, and show that it improves throughput while reducing power consumption.

3.2 Related Work

One of the motivating works in this field is a paper by Bahl et al. which argues that wireless devices ought to use multiple radios collaboratively to improve system performance and functionality [1]. In this paper, the authors prototype several new wireless devices with multiple radios. One uses a low-power radio to wake up the device and then uses a higher power 802.11 radio for standard WLAN communication. This strategy can significantly extend the battery life of a PDA. Another prototyped system uses multiple radios to provide higher capacity for wireless mesh networks. Similar work by Rodriguez et al. uses a mobile access router to provide improved data performance by using multiple service providers, technologies, and wireless channels [41].

This work illustrates the benefits of using multi-radio devices intelligently. Our work is complementary in that we are seeking to predict link quality so that a device can utilize those radios likely to provide good service in the near future. This is a challenging problem when the available radios may be intermittently available due to mobility and interference.

One of the most active related areas is concerned with mobility prediction for mobile ad hoc networks. In this area, each node has a single radio, so it is important to predict when nodes will move far enough away from each other that they can no longer communicate. At this point, the routing protocol must be invoked to find a new path. Doss et al. provide a good review of techniques in this area [7]. McDonald and Znati develop a probabilistic model of link availability to predict the future status of a wireless link, based on a random mobility model [32]. They then use this to improve routing by placing a bound on the probability of path failure. Jiang et al. provide a probability of continuous link availability for some period in the future, so that the routing protocol can then choose paths based on their stability [23]. Other work uses GPS devices to predict mobility [46]. One thing to be careful of, however, is that shadowing in an urban environment greatly affects link quality, causing several link quality prediction algorithms to perform poorly [16].

Several projects predict link availability or link quality based on past measurements. Gerharz et al. predict link stability based on a statistical evaluation of link lifetime from past observations [17]. Farkas et al. use pattern matching to predict link lifetime, using a circular buffer of past SNR measurements [15]. Other work uses use signal strength or success rate to characterize link quality in sensor networks [28, 26]. In sensor networks, since nodes are typically static, relatively few samples are needed to determine link quality, In our case, we need to sample more frequently because the devices we use are mobile.

	Application Type			
Metric	*VoIP*	*Live Video*	*Streaming Video*	*Data*
Throughput	64 kbps	384 kbps	256 kbps	128 bps
Delay	150 ms	150 ms	4 s	–
Jitter	30 ms	30 ms	–	–

Table 3.1: Application QoS Requirements

3.3 Measurement Framework

Consider two devices – a laptop and a PDA, for example – each with a WiFi and a Bluetooth interface. The key concept we explore in this paper is how to predict the quality of each interface so that the device can dynamically choose the one that is most likely to meet application requirements at a given time. The connection may be initially routed through a TCP/IP stack to the WiFi interface, but at some later time if the WiFi link suffers interference the devices should switch the connection to Bluetooth, without affecting the application or notifying the user.

We use three metrics to express application QoS requirements: throughput, delay, and jitter. The application specifies its requirements in terms of a threshold that must be met – if the link can support these requirements, then it is said to be "qualified" for that application. The device then selects the best qualified link based on user preference, which could for example favor performance or power savings. Table 3.1 lists example requirements, which are based on relevant business documentation and academic papers [25, 47, 37, 33]. In this paper we use only VoIP applications, since they express a QoS threshold for all three metrics, but our results are similar for other classes of applications.

To justify our use of all three metrics, we ran a simulation of a WiFi radio as it experiences changes in link quality due to a combination of interference and mobility. Fig. 3.1 shows each of the three metrics alone, as well as in combination. There are clearly times when a single metric alone indicates the performance of the link is satisfactory, while other metrics indicate the link is not qualified. Only by
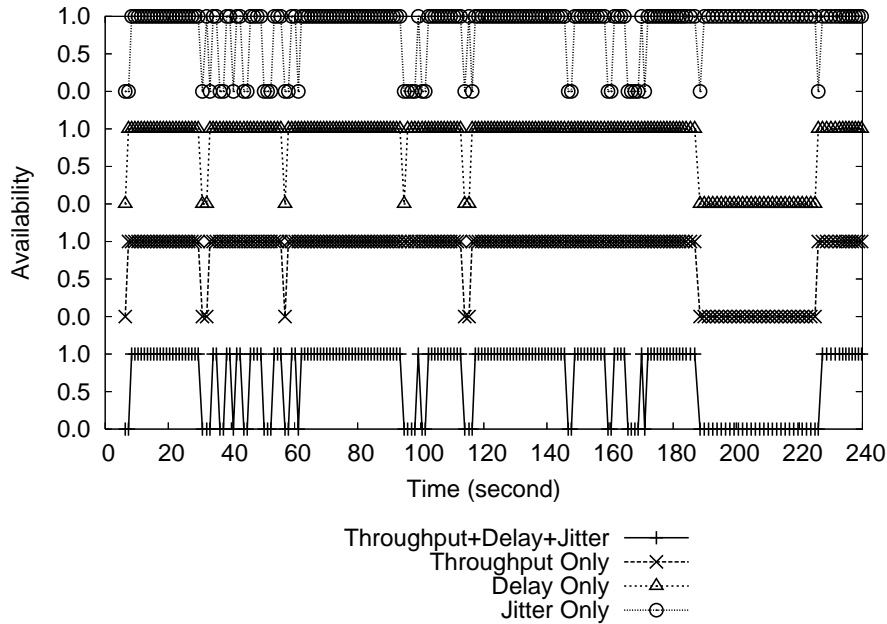
Figure 3.1: WiFi Link Quality

combining all three metrics are we able to ensure that the link can meet application QoS requirements.

3.3.1 Link Quality Measurements

To determine the quality of a link between two devices, we periodically send link-layer assessment queries from the primary device to the secondary device. This distinction is a natural fit for some technologies, such as Bluetooth; in others, we designate the initiator of communication as the primary device.

To make a query, the primary device sends a *link quality request* to the secondary device using the appropriate network stack, requesting its real-time quality information. Upon receiving the request, the secondary device measures it signal-to-noise ratio (SNR) and includes this in a *link quality response* using the same radio.

To compute the throughput of the link, the primary device averages its own SNR with the value in the response. It then uses this average to estimate the throughput of the link using the general Shannon's capacity formula in a Rayleigh Fading

Environment [30], which is the most applicable model when there is no dominant line of sight. Using this estimate provides an upper bound on capacity, ensuring that the system will never conclude that a link is unqualified when it is actually suitable. In addition, this method uses little power and overhead as compared to measuring the channel over a sustained period of time.

The primary device uses the round-trip time for the request and response as a measure of the link delay. The jitter is calculated as the difference between the delay for this request and the previous request.

To ensure the accuracy of our measurements, both the request and response are given priority in the device OS, so that the round-trip time gives an accurate measure of link delay, including any MAC negotiation. In addition, we pause active data traffic during the query to avoid any conflicts that might occur if multiple radios in a single device attempted to operate concurrently.

We begin by assuming that measurements are performed at regular intervals. We later show how the measurement interval can be varied dynamically, so that fewer measurements are performed during stable periods. Using dynamic measurements reduces overhead and conserves battery life, while still maintaining prediction accuracy.

3.3.2 Measurement Window

The wireless device keeps a window of past measurements and then tries to predict future link quality based on these measurements. The window contents are kept in FIFO order, so that a new measurement replaces the oldest measurement.

The size of the measurement window is an important parameter for the prediction algorithm. It should not be too large, since mobility can quickly cause old measurements to be outdated. The window should also not be too small, since interference can cause link quality to vary rapidly in a short period of time. Keeping enough measurements in the window can smooth out this behavior.

44

We use simulations to determine the appropriate window size in Section 3.7.

3.4   Prediction Algorithm

We develop a prediction algorithm based on the Weighted Least Square Regression (WLSR) algorithm [22]. This algorithm takes as input the window of current measurements for a given QoS metric, which occur at periodic intervals, and predicts the value of the metric for the next scheduled measurement period. Although this is a short period of time into the future (typically 1 second), this is enough time to enable the device to switch to a different radio if needed.

WLSR is an efficient prediction method that makes good use of small data sets. The only state required is the set of measurements considered (we use 5 to 30 measurements), and the algorithm can be implemented with about a hundred lines of code. No training or learning is required. Because WLSR applies weights to the measurements, we can treat them with different levels of importance according to their ages. This makes WLSR well-suited for a frequently changing mobile environment, since only recent performance measurements are useful in predicting future availability.

When predicting a future value for a metric, the WLSR algorithm calculates both a mean and a standard deviation for the prediction. This enables us to calculate a probability that the link will be able to meet the requested QoS threshold for that metric at the next measurement interval. We average the probabilities for each metric to arrive at an overall prediction for the link.

3.4.1   WLSR Weights

When using WLSR, it is critical that each measurement is weighted properly. In our case, we want to give more weight to more recent samples, in case the quality of the

link has changed recently. Accordingly, we number the samples from 1 to n, with n being the oldest, and set the weight for each measurement using:

$$\omega_i = \frac{1.0}{\kappa^i} \tag{3.1}$$

where κ is a fixed multiplier, ($\kappa \geq 1.0$). When $\kappa = 1$ all samples are treated equally, as in Ordinary Least Square Regression (OLSR).

The value for κ must be selected carefully. If the weight decreases too rapidly, old samples have very little contribution to the prediction, and only the most recent samples will be taken into consideration, ignoring any larger trend. On the other hand, if the weight decreases too slowly, all samples will be treated almost equally, making the prediction too dependent on old measurements.

We perform a full-factorial experiment to determine the proper value of κ in Section 3.7.

3.4.2 WLSR Method

We use the standard WLSR regression method [22, 13], given as:

$$\hat{m} = \hat{\alpha} * t + \hat{\beta} \tag{3.2}$$

where t is the measurement time, $\hat{m}$ is the value of a QoS metric (throughput, delay, or jitter) at time t, and $\hat{\alpha}$ and $\hat{\beta}$ are the regression parameters. The regression parameters are calculated by minimizing the Weighted Sum of Square Errors (WSSE) between the data in the measurement window and the performance level computed using the estimation model:

$$WSSE = \sum_{i=1}^{n} \omega_i e_i{}^2 = \sum_{i=1}^{n} \omega_i (m_i - (\hat{\alpha} * t_i + \hat{\beta}))^2 \tag{3.3}$$

Solving equations $\frac{\partial WSSE}{\partial \hat{\alpha}} = 0$ and $\frac{\partial WSSE}{\partial \hat{\beta}} = 0$ gives the WLSR estimations of $\hat{\alpha}$ and $\hat{\beta}$ as:

$$\hat{\alpha} = \frac{\sum_{i=1}^{n} \omega_i m_i t_i - n * \overline{\sqrt{\omega}t} * \overline{\sqrt{\omega}m}}{\sum_{i=1}^{n} \omega_i t_i{}^2 - n * \overline{\sqrt{\omega}t}^2} \tag{3.4}$$

$$\hat{\beta} = \overline{\sqrt{\omega}m} - \hat{\alpha}\overline{\sqrt{\omega}t} \tag{3.5}$$

where n is the size of the measurement window.

Once we calculate the regression parameters, we can then predict a future value for the metric by:

$$\hat{m}_p = \hat{\alpha} * t_p + \hat{\beta}, \tag{3.6}$$

which provides the predicted mean for the metric at time t_p. We also calculate he standard deviation of the prediction using:

$$S_{\hat{m}_p} = S_e * [1 + \frac{1}{n} + \frac{(t_P - \overline{\sqrt{\omega}t})^2}{\sum_{i=1}^{n} \omega_i t_i{}^2 - n * \overline{\sqrt{\omega}t}^2}]^{\frac{1}{2}} \tag{3.7}$$

where the standard deviation of the estimate, (S_e), is:

$$S_e = \sqrt{\frac{WSSE}{n-2}} \tag{3.8}$$

3.4.3 Qualification Probability

We define the qualification probability for a link as the likelihood that the predicted performance exceeds the QoS threshold specified by the application, such as those listed in Table 3.1.

To calculate the qualification probability, we construct a pdf for the metric at the prediction time, using the predicted mean and standard deviation. We use
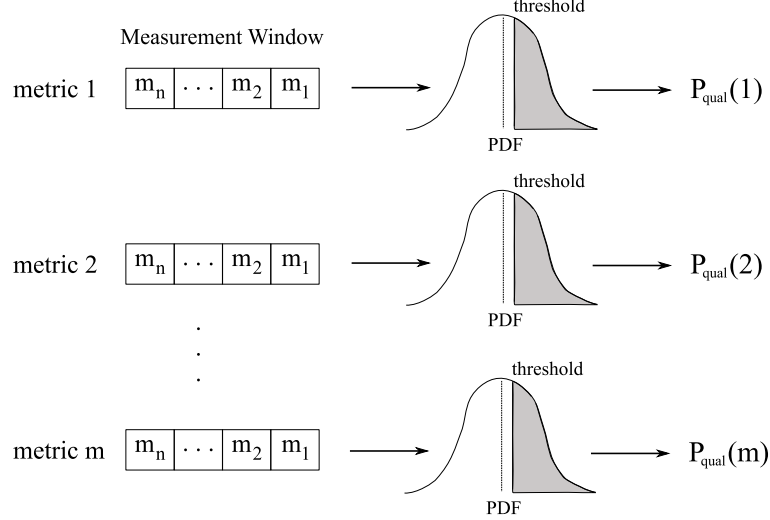
Figure 3.2: Qualification Probability

a t distribution for the pdf since the sampled data set is small. The qualification probability, P_{qual}, is the area under the pdf that meets the threshold (to the right for throughput and to the left for delay and jitter). We consider the link to be qualified for this metric if P_{qual} is at least 50% of the total area. This is equivalent to the mean of the pdf meeting the QoS threshold for the metric under consideration. This process is illustrated in Fig. 3.2.

The overall availability of the link, (P_{avail}), is given by averaging the qualification probability of each metric, where m is the number of metrics.

$$P_{avail} = \sum_{i=1}^{m} 1/m * P_{qual}(i), \tag{3.9}$$

We consider the link to be acceptable for the application if its predicted performance meets the QoS requirements of all relevant metrics. Accordingly, we rate the link as available if $P_{avail} \geq 50\%$. It is possible that the overall availability is greater than 50%, even when the link is not qualified for one or more of the metrics. In this case, we artificially assign the overall availability to 40%, so that the link is considered unavailable.

3.5 Dynamic Link Quality Measurements

In our previous discussion, we have assumed a fixed link measurement interval. In this section we describe a control system that uses fuzzy logic to vary the link measurement interval so that fewer measurements are made during periods of link stability. The challenge is to reduce overhead and conserve power, while still maintaining prediction accuracy.

Fuzzy logic has several advantages that make it particularly suitable for adjusting the link measurement interval [24,29,31]. First, fuzzy logic is capable of highly adaptive control, making it suitable for dynamic environments. It is able to control nonlinear systems that would be difficult or impossible to model mathematically, facilitating control systems that would normally be deemed unfeasible for automation. Second, fuzzy logic is very robust because it does not require precise, noise-free inputs, and the output is a smooth control function despite a wide range of input variation. We note that fuzzy logic has been applied to many control systems, ranging from simple, small, embedded micro-controllers to large, networked, multi-channel data acquisition and control systems.

When varying the link measurement interval, we make a distinction between two types of radio handoffs. An *upgrade* occurs when a more desirable radio becomes available and the device switches from the active radio to this better radio. A *downgrade* occurs when the active radio becomes unavailable and the device must switch to a less desirable radio. We consider these two cases separately. The active radio is always measured using a fixed interval of 1 second. We do not adjust the measurement interval for the active radio, so that the system can react quickly to any changes in its availability.

3.5.1 Downgrade Radios

The measurement interval for a potential downgrade radio $(I_{R_{down}})$ is a function of three factors:

$$I_{R_{down}} = f(P_{avail}(R_a), P_{avail}(R_{down}), Pref(R_{down})) \qquad (3.10)$$

The predicted availability of the active radio, $P_{avail}(R_a)$, is the primary determining factor. If $P_{avail}(R_a)$ is high, indicating that the radio currently in use is predicted to continue being available, the system can decrease the measurement frequency of downgrade radios to save power. On the other hand, If $P_{avail}(R_a)$ is low, suggesting that the active radio is at a risk of dropping off, the system needs to query downgrade radios more frequently in order to know their current status in case a radio switch is needed.

Secondary factors for the measurement interval are the predicted availability of the downgrade radio, $P_{avail}(R_{down})$, and the preference of the downgrade radio $(Pref(R_{down})$. The preference of the radio is based on its characteristics and whether the user prefers to favor throughput or power efficiency. For example, if the user prefers power efficiency, in order to maximize battery lifetime, then she will rank a Bluetooth radio above a WiFi radio. A radio that is preferred more than other downgrade radios, and that also has a high predicted availability, is likely to be selected for a downgrade. Link quality measurements should be performed more frequently on such a radio in order to accurately maintain its status. On the other hand, a radio that is low on the preference list or that has a low predicted availability should be measured less frequently to save system power.

To build a fuzzy logic control system, we define an input membership function for the input variables $P_{avail}(R_a)$ and $P_{avail}(R_{down})$, as shown in Fig. 3.3. This function maps the input variables to a degree of membership in three fuzzy sets: High, Medium,
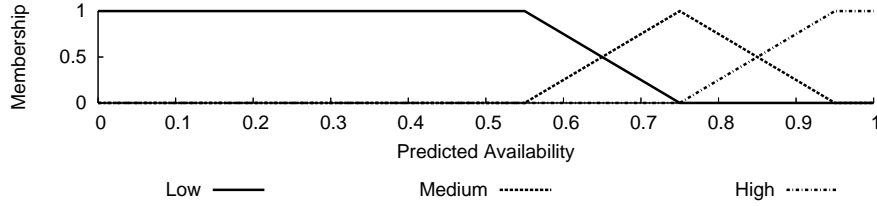
Figure 3.3: Input Membership Function for Downgrade Radios

and Low. Adding more fuzzy sets can improve resolution and provide more sensitive control, but causes extra complexity as well. Since 50% is the threshold used for the availability prediction, these three fuzzy sets fall in the range of (50%, 100%). To reduce computational overhead, we use the common triangle shaped membership function, with a typical overlapping of 50% of width [24].

To better understand the input membership function, consider an active radio with a predicted availability of 0.9 and a downgrade radio whose predicted availability is 0.7. First we use the input membership function for $P_{avail}(R_a)$ to find its membership values for the three fuzzy sets: 0.75 for "High", 0.25 "Medium", and 0 for "Low". Similarly, we calculate the membership values for $P_{avail}(R_{down})$, resulting in 0 for "High", 0.75 for "Medium", and 0.25 for "Low".

Given such a mapping, the second step of our control system processes the set memberships of all input variables using decision rules, as shown in Table 3.2, and calculates the system output. Based on different combinations of the two inputs, there are seven possible output sets. The membership value for a given set is the product of the memberships for the two input membership values.

Continuing our previous example, R_a has a positive membership in "High" and "Medium", and R_{down} has a positive membership in "Medium" and "Low". Applying the decision rules, we obtain four output fuzzy sets with positive memberships: "Larger" has a membership of $0.75 * 0.75 = 0.56$, "Largest" has a membership of $0.75 * 0.25 = 0.19$, "Normal" has a membership of $0.25 * 0.75 = 0.19$, and "Large"

$P_{avail}(R_a)$	$P_{avail}(R_{down})$	**Output Fuzzy Sets**
	High	Smallest
Low	Medium	Smaller
	Low	Small
	High	Small
Medium	Medium	Normal
	Low	Large
	High	Large
High	Medium	Larger
	Low	Largest

Table 3.2: Decision Rules for Downgrade Radio Measurement Interval
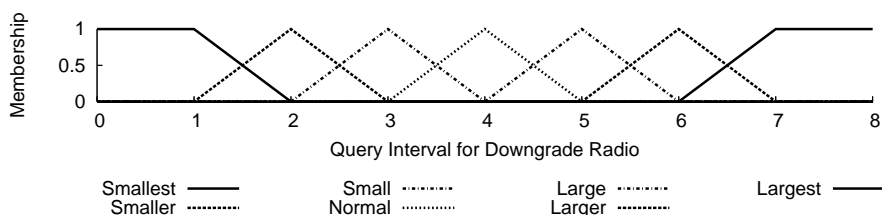

Figure 3.4: Output Membership Function for Downgrade Radios

with a membership of $0.25 * 0.25 = 0.06$. The memberships of all other output fuzzy sets is 0.

As a third step, our control system calculates the measurement interval for the radio using an output membership function, shown in Fig. 3.4. This function again uses the common triangle shape with a typical overlapping of 50%. The membership function is divided into $N_{down} = 7$ regions, corresponding to each output fuzzy set. The range of the function is from the minimum measurement interval, min_{down}, to the maximum measurement interval, max_{down}, in this case 1 to 7 seconds. Thus "Smallest" corresponds to a measurement interval of 1 second, and "Largest" corresponds to a measurement interval of 7 seconds. We later use simulations to determine the best settings for the range of this function.

The measurement interval for the radio is computed using the weighted average of all the output fuzzy sets, where the weight is the membership value. Continuing

52

our previous example, the measurement interval for the downgrade radio is $I_{Rdown} = 0.19 * 4 + 0.06 * 5 + 0.56 * 6 + 0.19 * 7 = 5.75$ seconds.

As a final step, we adjust the measurement interval based on the preference ranking of the radio. If there exists a more preferred radio we could downgrade to, and it has an equal or higher availability prediction than the R_{down} we are considering, that radio is more likely to be selected as a downgrade candidate. In this case, the measurement interval for R_{down} should be increased. Formally, we increase the measurement interval of R_{down} by $2 * ((max_{down} - min_{down} + 1)/N_{down})$, if there exists a radio R_i, such that $Pref(R_i) < Pref(R_{down})$ and $P_{avail}(R_i) >= P_{avail}(R_{down})$.

Concluding our example, suppose there is another potential downgrade radio that is more preferred and has a greater predicted availability. In this case, $I_{R_{down}} = 5.75 + 2 * (7 - 1 + 1)/7) = 7.75$.

3.5.2 Upgrade Radios

The measurement interval for a potential upgrade radio ($I_{R_{up}}$) is a function of two factors:

$$QI_{R_{up}} = f(P_{avail}(R_{up}), Pref(R_{up})) \tag{3.11}$$

Link quality measurements should be performed more frequently on a radio that has a high predicted availability and that is preferred more highly than other alternatives. Likewise, a radio should be measured less frequently if has a low predicted availability or if there are other more preferred radios that are likely to be available.

The input membership function for upgrade radios, shown in Fig. 3.5, is similar to the one used for downgrade radios. For upgrade radios we use $N_{up} = 5$ fuzzy sets: Very High, High, Medium, Low, and Very Low. We add two additional fuzzy sets for more sensitive control, since $P_{avail}(R_{up})$ is the only input.
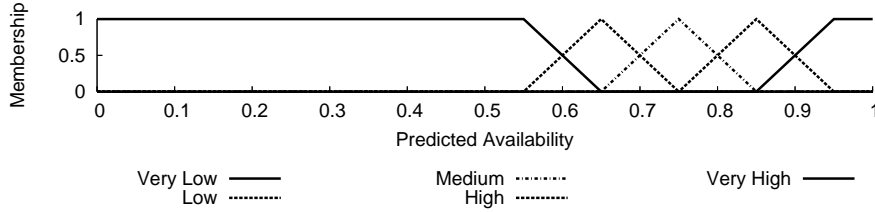
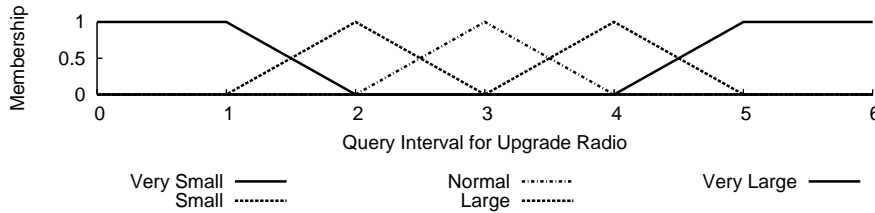Figure 3.5: Input Membership Function for Upgrade Radios

Figure 3.6: Output Membership Function for Upgrade Radio

Consider a potential upgrade radio with a predicted availability as 0.8. Using the given input membership function, we calculate the membership of R_{up} in the output sets: 0.5 for "High", 0.5 for "Medium", and 0 for all other sets.

The system does not need decision rules in this case since there is only a single input variable. The output membership function maps the output sets into a measurement interval as shown in Fig. 3.6. In this example, the minimum interval, min_{up}, corresponding to "Very High", is 1 second, and the maximum interval, max_{up}, corresponding to "Very Low", is 5 seconds.

The system output is the weighted average of all the output fuzzy sets, where the weight is the membership is the membership value. Using our example, the measurement interval $I_{R_{up}} = 0.5 * 1 + 0.5 * 2 = 1.5$ seconds.

The preference ranking of R_{up} also affects how frequently measurements should be performed. If a more preferred radio performs equally well or better, we increase the measurement interval of R_{up} to save more power. We use the same procedure as for downgrade radios.

3.5.3 Effect on Prediction Algorithm

The prediction algorithm uses past measurements to predict future availability. What should it do when measurements are spaced farther apart, due to the dynamic measurement intervals? In our work, we assume that measurements always occur at regular 1 second intervals. If a measurement is "skipped" because the interval has been increased, the algorithm assumes the value of this measurement is the same as the preceding measurement. This enables us to compare performance of the same algorithm, both with and without the dynamic measurement intervals.

3.6 Simulation Methodology

We perform a simulation study to evaluate accuracy and efficiency of the prediction algorithm using ns-2.28 [14]. We implemented an interface interference model, the link measurement mechanism, and the prediction algorithm.

3.6.1 Topology

We use a topology, shown in Fig. 3.7, that includes two mobile devices with WiFi, Bluetooth, WirelessUSB, and ZigBee as common radios. The two devices use a VoIP application running over UDP. We note that the choice of application and transport protocol does not affect the prediction accuracy. We consider different transport protocols when evaluating overhead.

The simulation topology is designed so that the link quality of the hour radios varies due to both mobility and interference. To simulate mobility we move the multi-radio devices in and out of range of each other. To simulate interference we turn on and off a set of Bluetooth devices and a set of WiFi devices. The scenarios we use are sufficient to show how well our prediction algorithm works for both frequent and infrequent changes due to mobility, as well as high volatility due to interference. We
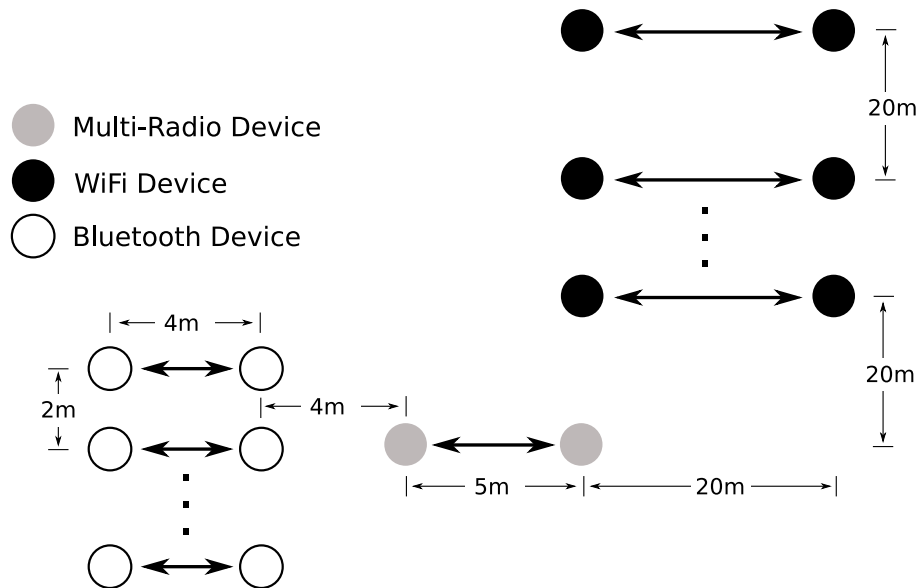
Figure 3.7: Simulation Topology

use only a VoIP application in our simulations because it has QoS requirements for all three of the link quality metrics – throughput, delay, and jitter. Our results are the same for other application types; the only difference is that the prediction algorithm takes into account fewer metrics.

3.6.2 Prediction Accuracy

To evaluate the accuracy of our prediction algorithm, we compare the predicted availability to a series of *ideal predictions* generated with the benefit of hindsight of all measurements, both future and past. Choosing an ideal prediction for a given moment depends on the tradeoff between fast response time versus stability. Some applications or users may want to switch immediately when a single measurement indicates a link has become unavailable, to minimize disruption or to maximize throughput. Other applications or users may prefer to stay with a particular radio for some period of time, to avoid the disruption or overhead that may occur with frequent switching. To balance this tradeoff, we define a period τ during which the user prefers to stay
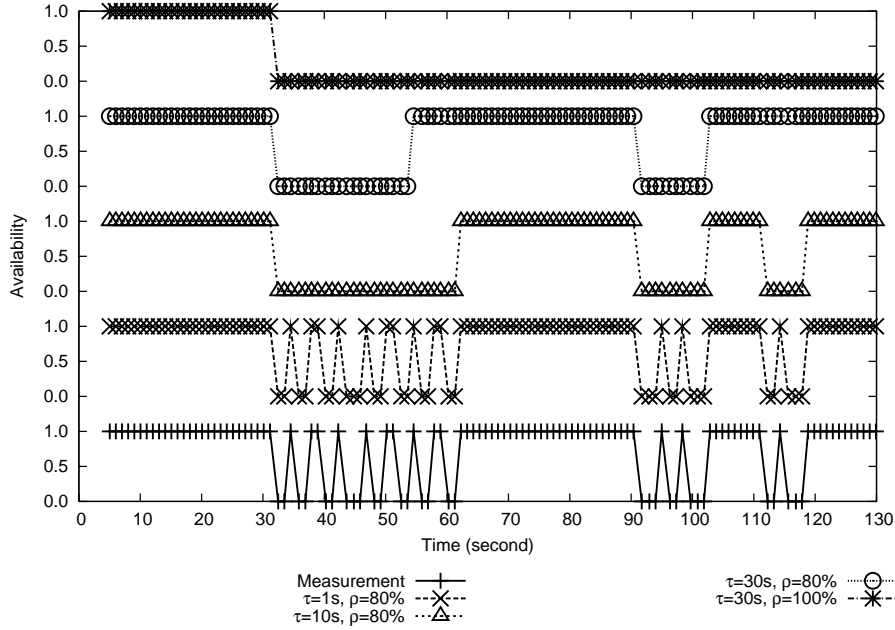
56

Figure 3.8: Measurements and Ideal Predictions for Bluetooth Radio

with a single radio. If the user wants to switch aggressively, then τ could be 1 second, whereas if she prefers stability it could be 20 seconds.

Another consideration for the ideal prediction is the user's patience with interference. We define persistence, ρ, as the percentage of measurements that must indicate the link is qualified during the period τ. For example, some users may insist that the link never suffer from interference, for a ρ of 100%, while others may be comfortable if the link is qualified 80% of the time during the measurement period.

To illustrate the different ideal predictions that are possible, Fig. 3.8 plots curves for various settings of τ and ρ. The measurements are taken from a simulation of a Bluetooth radio for the two multi-radio devices. Both devices are initially stationary, then one of them moves in and out of Bluetooth range (but still in WiFi range) for the periods from $30s$ to $60s$, $90s$ to $100s$ and $110s$ to $115s$. Notice that curves with a small τ follow the measurement pattern closely, while using a large τ and ρ is much more stable.

Given an ideal prediction, we then compare it to an actual prediction and calculate the *prediction error rate*, which is the ratio of the number of incorrect predictions to the total number of predictions made during the simulation. The prediction is defined as incorrect if the link is estimated to be unavailable while it is actually available based on the ideal prediction, and vice versa.

3.6.3 Switching Accuracy

While prediction accuracy is important, ultimately what matters most is that the multi-radio device is able to consistently choose the best radio. Based on the ideal prediction for all available radios, we generate an *ideal radio switching* curve, which chooses the best available radio at every single moment. If, at a given time, there are n available radios, as determined by the ideal prediction algorithm, then ideal radio switching chooses the radio that delivers the highest throughput or the best power savings, depending on user preference. Likewise, we construct a radio switching curve based on predicted availability, so that we can determine how well our system performs compared to the ideal system.

Given the actual and ideal switching curves, we measure the *switching error*, which is the amount of time that our switching algorithm is using a different radio than the ideal switching curve. There may be times when our prediction algorithm is incorrect, since it is difficult to predict the future, but if the switching error is small, then the prediction error is not significant.

3.7 WLSR Weights and Window Size

We begin our evaluation of the prediction algorithm by determining the proper settings for the WLSR weights and the measurement window size. For these experiments, we use a fixed measurement interval of 1 second, then perform a full factorial experiment using different WLSR weights and measurement window sizes.

Measurement Window Size			5			10			15		
WLSR Weight			1.0	1.3	1.6	1.0	1.3	1.6	1.0	1.3	1.6
$\tau = 1s$ $\rho = 80\%$	BT Mobility	WiFi	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		BT	17.65%	20.23%	20.84%	22.07%	19.76%	21.90%	29.62%	22.25%	23.33%
	WiFi Mobility	WiFi	10.00%	8.33%	10.00%	9.27%	7.09%	7.09%	11.00%	6.00%	5.80%
		BT	6.67%	5.00%	5.00%	9.09%	5.45%	5.45%	11.40%	6.00%	6.00%
	Interference	WiFi	17.41%	16.66%	16.66%	22.79%	19.67%	18.03%	29.67%	21.96%	19.82%
		BT	11.14%	8.28%	8.28%	12.70%	11.08%	9.28%	12.77%	12.23%	10.46%
Average			10.48%	**9.75%**	10.13%	12.65%	10.51%	10.29%	15.74%	11.41%	10.90%
$\tau = 5s$ $\rho = 80\%$	BT Mobility	WiFi	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		BT	7.51%	10.12%	10.73%	11.04%	8.73%	10.87%	17.59%	10.23%	11.30%
	WiFi Mobility	WiFi	10.08%	8.40%	10.08%	9.27%	7.09%	7.09%	11.00%	6.00%	5.80%
		BT	6.72%	5.04%	5.04%	9.09%	5.45%	5.45%	11.40%	6.00%	6.00%
	Interference	WiFi	15.25%	14.18%	15.40%	17.03%	14.57%	15.55%	23.73%	16.41%	16.75%
		BT	11.22%	9.26%	9.26%	13.35%	11.72%	10.26%	13.83%	13.65%	11.53%
Average			8.46%	**7.83%**	8.42%	9.96%	7.93%	8.21%	12.93%	8.72%	8.56%
$\tau = 10s$ $\rho = 80\%$	BT Mobility	WiFi	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		BT	7.81%	10.52%	11.16%	11.13%	8.80%	10.96%	17.59%	10.23%	11.30%
	WiFi Mobility	WiFi	10.53%	8.77%	10.53%	9.36%	7.16%	7.16%	11.00%	6.00%	5.80%
		BT	7.02%	5.26%	5.26%	9.17%	5.50%	5.50%	11.40%	6.00%	6.00%
	Interference	WiFi	15.86%	15.05%	16.33%	18.17%	15.68%	16.35%	24.46%	17.48%	17.47%
		BT	11.50%	10.39%	10.39%	13.93%	12.62%	11.15%	14.68%	14.16%	12.39%
Average			8.79%	8.33%	8.94%	10.29%	**8.29%**	8.52%	13.19%	8.98%	8.83%

Table 3.3: Full Factorial Experimental Results

In selecting scenarios for this experiment, our goal is to have enough variation in radio availability so that the WLSR parameters we choose will work across a wide range of possible situations. Accordingly, we use scenarios that include times when the radio is continuously available, times of periodic unavailability, and times of high volatility.

We use the topology shown in Fig. 3.7, and the multi-radio devices share both a WiFi and a Bluetooth radio. We run experiments with the following three scenarios, each of which lasts for 120 seconds:

- **BT Mobility:** One of the multi-radio devices stays stationary, while the other moves in and out of Bluetooth coverage at a speed of $4m/s$ during the following periods: $30s$ - $60s$, $90s$ - $100s$ and $110s$ - $115s$.

- **WiFi Mobility:** One of the multi-radio devices stays stationary, while the other moves in and out of WiFi coverage at a speed of $30m/s$ during the following periods: $30s$ - $65s$ and $75s$ - $110s$.

- **Interference:** The multi-radio devices stay stationary. The 5 pairs of Bluetooth nodes and the 10 pairs of WiFi nodes generate traffic during the periods $30s$ - $60s$, $90s$ - $100s$ and $110s$ - $115s$.

For each scenario, we run an experiment with each possible combination of the WLSR weight κ and the measurement window size n. The weight κ varies from 1.0 to 1.6, in increments of 0.1, and the window size n varies from 5 to 30, in increments of 5. We repeat each experiment five times. Because of space constraints, we show only the most relevant results. With other parameters we tested, the prediction error rate either increases or remains the same.

Table 3.3 shows the results of the full factorial experiment. Each row lists the prediction error for both the WiFi and Bluetooth radios for a given experiment under the listed scenario. The first group of experiments compares the prediction to an ideal curve with τ equal to 1 second and ρ equal to 80%, with subsequent groups compared against other settings for the ideal curve. For each group we mark the best average prediction in bold.

When responsiveness is preferred (τ is small), a small measurement window and a larger weight work best. This gives the highest weight to the most recent of a small number measurements, so that the prediction is likewise more responsive. As τ increases, a larger window and smaller weight begin to perform better. Using these settings enables the prediction algorithm to be more stable.

Based on these experiments, we believe a measurement window size of 5 or 10 and a WLSR weight of 1.3 provides a good balance between reactivity and stability. These settings perform well across all the experiments.

In subsequent simulations, we use $\tau = 10s$ and $\rho = 80\%$ for the ideal curve. Based on Fig. 3.8, this provides good immunity to interference while also reacting to changes due to mobility. Subsequent simulations also use a measurement window of 10 and WLSR weight of 1.3.

3.8 Comparison to Other Algorithms

We next evaluate the WLSR prediction algorithm by comparing it to two alternative algorithms: Exponential Weighted Moving Average (EWMA) and a prediction based on the signal strength only. We continue to use a measurement interval of 1 second.

We use the topology shown in Fig. 3.7, and the multi-radio devices share both a WiFi and a Bluetooth radio. We run experiments with the following three scenarios:

- **Mobility:** One of the multi-radio devices stays stationary, while the other other moves in and out of Bluetooth coverage at a speed of $4m/s$ during the following periods: $30s$ - $60s$, $90s$ - $100s$, $110s$ - $115s$. This same node also moves out of WiFi coverage at 150s, and then moves back at 190s, at a speed of $30m/s$

- **Interference:** The two multi-radio devices stay stationary. The 5 pairs of Bluetooth nodes and the 10 pairs of WiFi nodes generate traffic during the periods $30s$ - $60s$, $90s$ - $100s$ and $110s$ - $115s$. The Bluetooth nodes also generate traffic from $190s$ - $220s$ and the WiFi nodes also generate traffic from $140s-170s$.

- **Combined:** The combination of both the mobility and the interference scenarios.

We run each simulation for 240 seconds, with one link measurement query per second for each radio. We repeat each simulation five times and combine the results.

We show the average prediction error presented for this experiment in Table 3.4. In almost all scenarios, the WLSR algorithm is able to predict link quality more accurately than the other two algorithms. The only exception is the prediction for WiFi in the mobility scenario, where the error rate of WLSR is only slightly higher than that of signal strength model. This is a good result for WLSR, since signal strength prediction is mainly useful for mobility prediction, and WLSR does just as well. On average, more than 90% of the predictions using WLSR are correct, as compared to the ideal curve.

Prediction Algorithm		WLSR	EWMA	Signal
Mobility	WiFi	1.62%	2.71%	1.44%
	BT	6.13%	13.62%	11.09%
Interference	WiFi	12.30%	17.75%	22.45%
	BT	9.18%	9.35%	9.68%
Combined	WiFi	10.90%	15.35%	18.64%
	BT	6.62%	12.52%	12.71%
Average		**7.79%**	**11.88%**	**12.67%**

Table 3.4: Prediction Accuracy Comparison
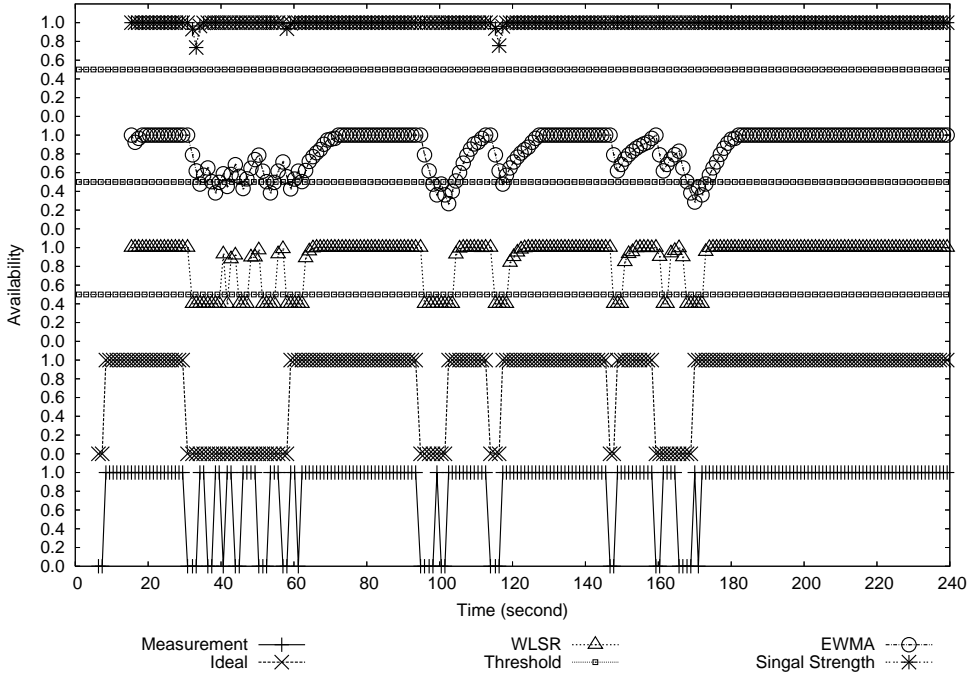
Figure 3.9: Link Quality Prediction of WiFi – Interference Only

To further illustrate the prediction accuracy of the WLSR algorithm, we plot the simulation results in Fig. 3.9- 3.12 , each with a randomly selected replication seed. The measurement curve represents the actual link status considering all relevant QoS metrics, either available (shown as 1.0) or unavailable (presented as 0.0). The ideal curve is generated from the measurement and based on user's preference ($\tau = 10s$ and $\rho = 80\%$). We then show the predictions for all three algorithms. On each prediction we plot the threshold at 0.5; if a prediction is above the threshold then the link is predicted to be available during that period.
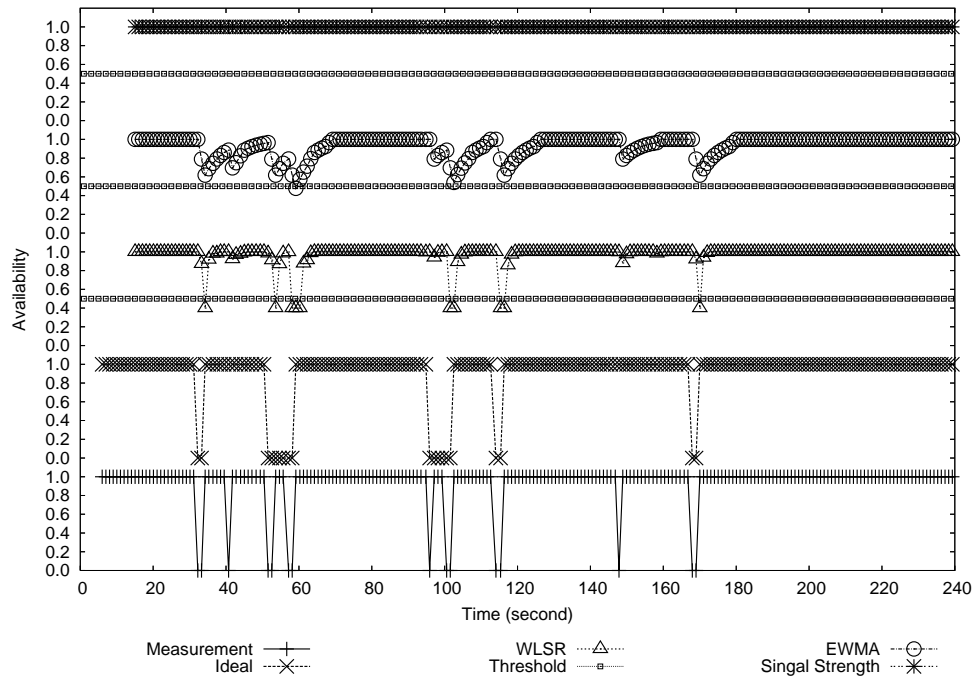
62

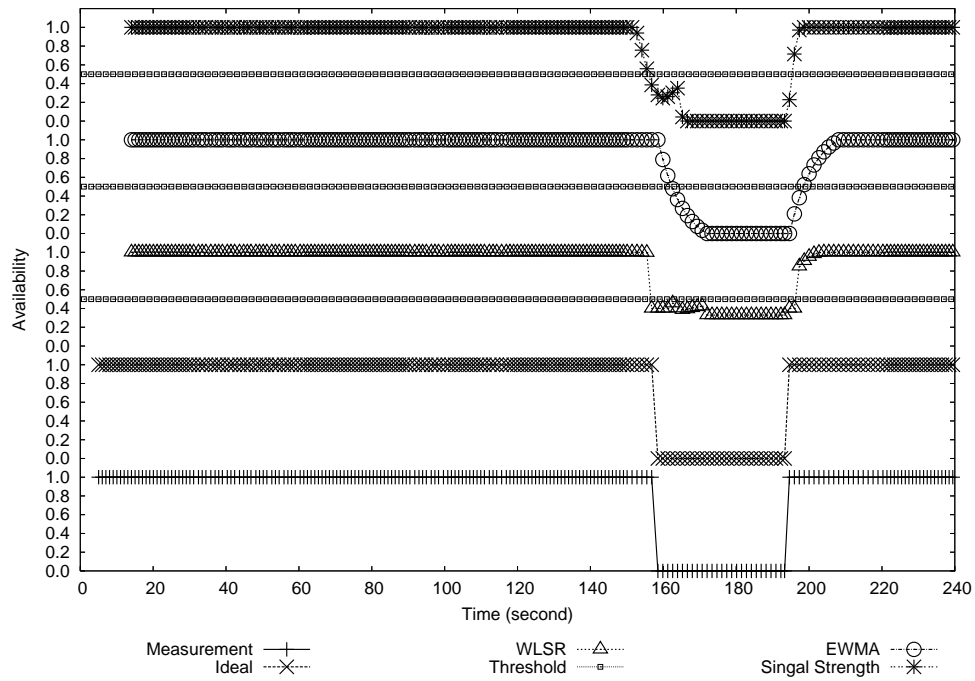Figure 3.10: Link Quality Prediction of BT – Interference Only

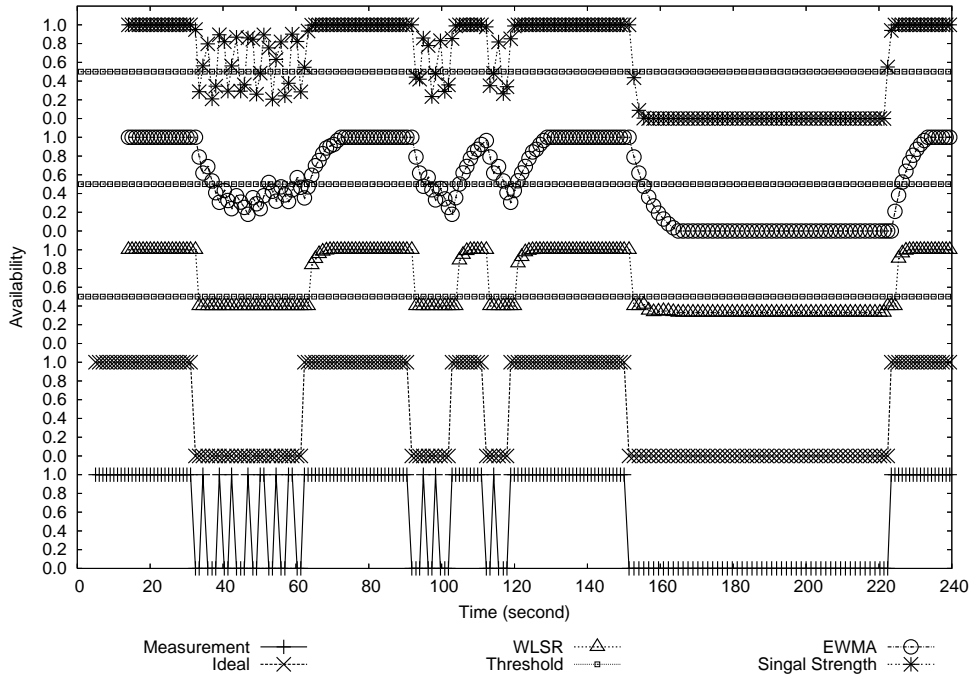Figure 3.11: Link Quality Prediction of WiFi – Mobility Only

Figure 3.12: Link Quality Prediction of BT – Mobility Only

These figures show why the WLSR prediction is more accurate than the EWMA and signal strength algorithms. It is able to closely match the ideal curve, with very little latency when changes occur. The only difficulty WLSR encounters is during longer periods of WiFi interference, when several consecutive good measurements briefly fool it into thinking the interference is gone.

In general, the EWMA algorithm reacts more gradually to changes in link status, taking an extra second or two to react. As a smooth function, EWMA does not react quickly to sudden changes. This causes particular difficulty during short periods of interference, as it may never declare the link unavailable. During longer periods of WiFi interference it also is fooled into thinking the interference is gone, but appears to have more difficulty than WLSR.

Using only the measured signal strength works well for mobility-induced changes. However, it is so sensitive to rapid changes in Bluetooth mobility that it does not provide the stability we prefer during these periods. The signal strength

algorithm also does not handle interference well, perceiving that the link is available the entire time.

Another interesting point about the WLSR algorithm is that it often produces a probability of 40%. Recall that this value is arbitrarily assigned when the overall probability would be above 50% but one of the metrics by itself disqualifies the link. This frequently occurs when jitter is unacceptable, even though throughput and delay are met.

Finally, note that even if a link is obviously unavailable due to mobility, the WLSR prediction does not reach 0. This is because the predicted availability is a probability calculated based on the pdf functions of all relevant metrics. Even though the predicted performance is much lower than the threshold, the probability computed will not be 0.

3.9 Overhead

Periodic link quality measurements impose overhead in terms of throughput and power consumption. To evaluate overhead we generate UDP traffic under three workloads – light (10 packets/s), medium (100 packets/s) and heavy (1000 packets/s), with 128 byte packets. For a fourth workload we generate a constant stream of TCP traffic. In all cases we measure throughput both with and without queries and then calculate throughput loss as the ratio of $throughput_{with} - throughput_{without}$ to $throughput_{without}$. We calculate the power consumed as a percentage of the overall battery life per minute, and we do this separately for the queries and the background workload. The initial battery level is 10 watt-hours, which is the typical battery level for PDAs. Table 3.5 shows the average overhead as computed from five replications of the experiments.

These results show that the overhead for link quality measurements is very low, with almost negligible throughput loss and power consumption. In all cases, the

Transport Protocol	UDP			TCP
Workload	Light	Medium	Heavy	–
Throughput	-0.05%	-0.12%	-0.40%	-0.19%
Power (queries)	0.0034%	0.0034%	0.0032%	0.0030%
Power (workload)	0.0066%	0.0144%	0.0592%	0.0660%

Table 3.5: Link Measurement Overhead

throughput loss is below 0.5%, and the power consumed is generally small compared to the workload. The throughput loss during light UDP loads is smaller because the link is usually not busy when queries are sent.

3.10   Using Dynamic Link Quality Measurements

We can further decrease the overhead of periodic link quality measurements by adaptively increasing the measurement interval during stable periods. To measure the effectiveness of this approach, we run experiments that compare the dynamic measurement intervals to fixed measurement intervals.

Recall that the dynamic algorithm uses fuzzy control, with the downgrade radio bounded by min_{down} and max_{down} and the upgrade radio bounded by min_{up} and max_{up}. We vary the minimum intervals from 0.25 to 1 seconds and the maximum intervals from 2 to 10 seconds. We compare the dynamic algorithm to the one that uses a fixed measurement interval, with values from 1 to 10 seconds.

We use two simulation scenarios that take into account different user preferences. In the high throughput scenario, the user prefers a radio with a high data rate; in the power efficient scenario, the user prefers a radio with low power consumption:

- **High Throughput Scenario**: One of the multi-radio devices stays stationary, and the other moves in and out of Bluetooth coverage at a speed of $4m/s$. Ten pairs of WiFi nodes and five pairs of Bluetooth nodes communicate with each other, causing interference on these radios. To maximize throughput, the multi-radio systems may downgrade from WiFi to Bluetooth, or from Bluetooth to

WirelessUSB, and they may upgrade from WirelessUSB to Bluetooth, or from Bluetooth to WiFi.

- **Power Efficient Scenario**: One of the multi-radio devices stays stationary, and the other moves in and out of WirelessUSB coverage at a speed of $11m/s$. Ten pairs of Bluetooth nodes communicate with each other, causing interference on the Bluetooth radio. To maximize power savings, the multi-radio systems may downgrade from WirelessUSB to Bluetooth, or from Bluetooth to WiFi, and they may upgrade from WiFi to Bluetooth, or from Bluetooth to WirelessUSB.

In both cases, we initiate mobility and interference 3 - 5 times during the simulation, with a random starting time. The duration of each mobility or interference event lasts from 30 - 50 seconds. We use a typical battery life for PDAs of 10 watt-hours. Each simulation runs for 300 seconds, and average our results over 50 replications.

Our results indicate that a minimum measurement interval below 1 second does not significantly improve prediction accuracy, while consuming more battery power. Because of this finding, we only show results here with minimum measurement interval of 1 second.

In Fig. 3.13 and Fig. 3.14 we plot the cumulative switching error versus the power consumption for our link quality prediction algorithm. We label the points for the dynamic measurement interval with the tuple (max_{down}, max_{up}). We likewise label the points for the fixed measurement interval with the time between measurements. The circles on the graph indicate clusters of points. For example the cluster labeled $(2.0 - 10.0, 2.0 - 3.0)$ have max_{down} in the range from 2 to 10 seconds and max_{up} in the range from 2 to 3 seconds.

The most important result shown in these figures is that the dynamic measurement interval provides a better tradeoff between switching accuracy and power
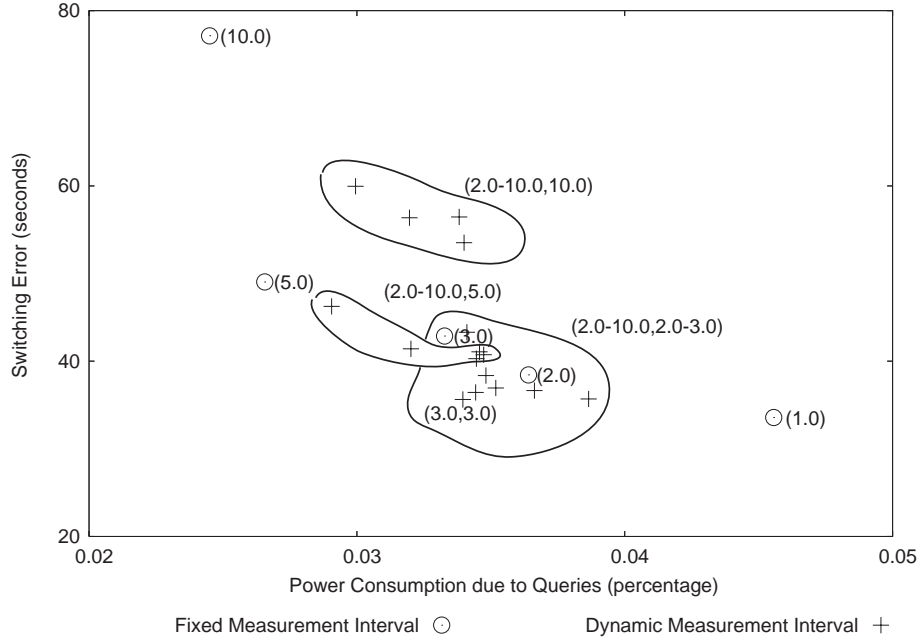
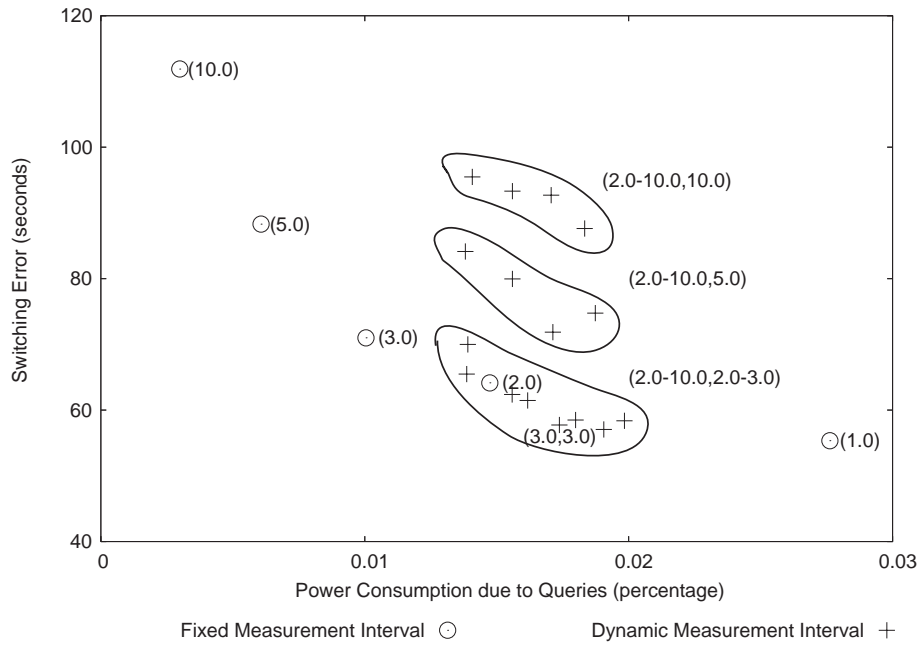Figure 3.13: Switching Error vs. Power Consumption: High Throughput Scenario

Figure 3.14: Switching Error vs. Power Consumption: Power Efficient Scenario

68

consumption. Using a fixed measurement interval allows the system to directly trade off better radio switching for more power consumption, depending on the measurement frequency. Using a dynamic measurement interval that may vary from 1 to 3 seconds allows the system to maintain the same accuracy as a fixed interval of 1 second, but with lower power consumption similar to using a fixed interval of 3 seconds. The best setting is labeled with $(3.0, 3.0)$ on both graphs.

Another result seen in these graphs is that high maximum measurement intervals provide generally the same power savings but with decreased switching accuracy. Thus there is no advantage to using a maximum measurement interval as large as 10 seconds.

To further illustrate how the dynamic measurement interval works, we randomly select one experiment from both of the simulation scenarios. These figures, shown in Fig. 3.15 and Fig. 3.16, show how our system selects a different radio over time, based on predicting link availability. In each figure, the lower half shows ideal radio switching, based on knowing future availability. The upper half shows the radio switching performed by our system, along with the periodic link measurements. Note that the measurement interval for each radio increases when it is not being considered for use in the near future, and decreases when it is the active radio or is likely to be used for an upgrade or downgrade. Overall, the radio switching for our system matches the ideal system fairly closely.

3.11   System Performance

To illustrate the utility of our prediction algorithm, we examine the performance of a multi-radio system that chooses the best available radio based on link quality predictions. For this simulation we use the same, simple radio switching algorithm as in the previous section. If, at a given time, there are n available radios, as determined
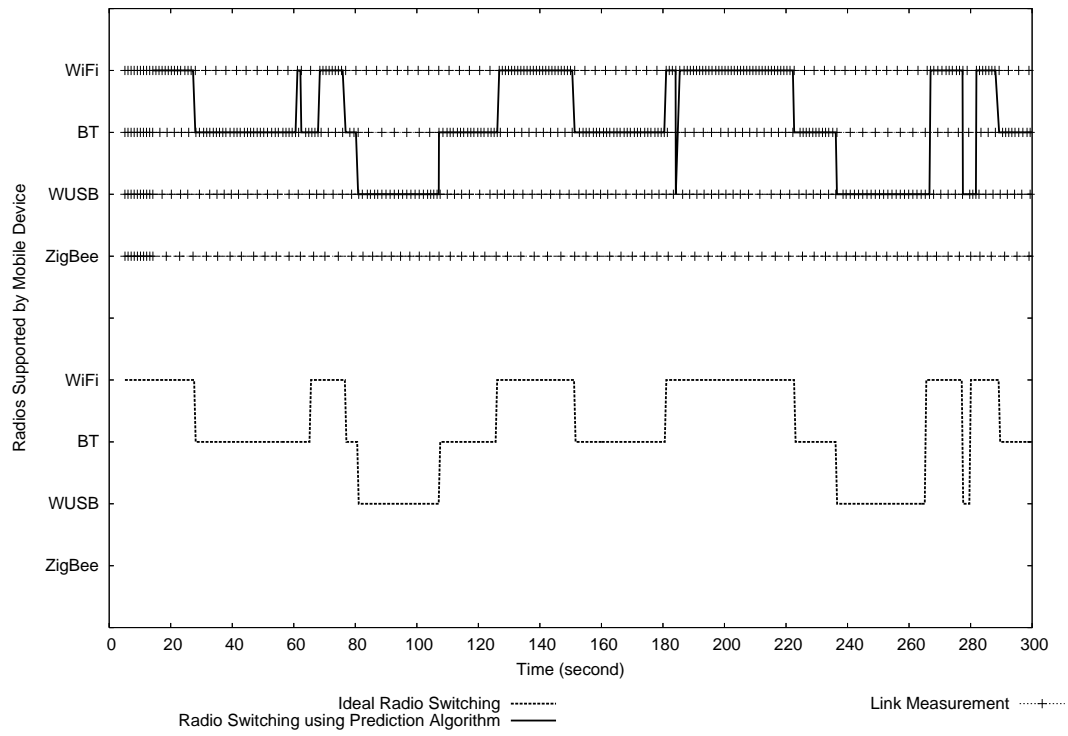
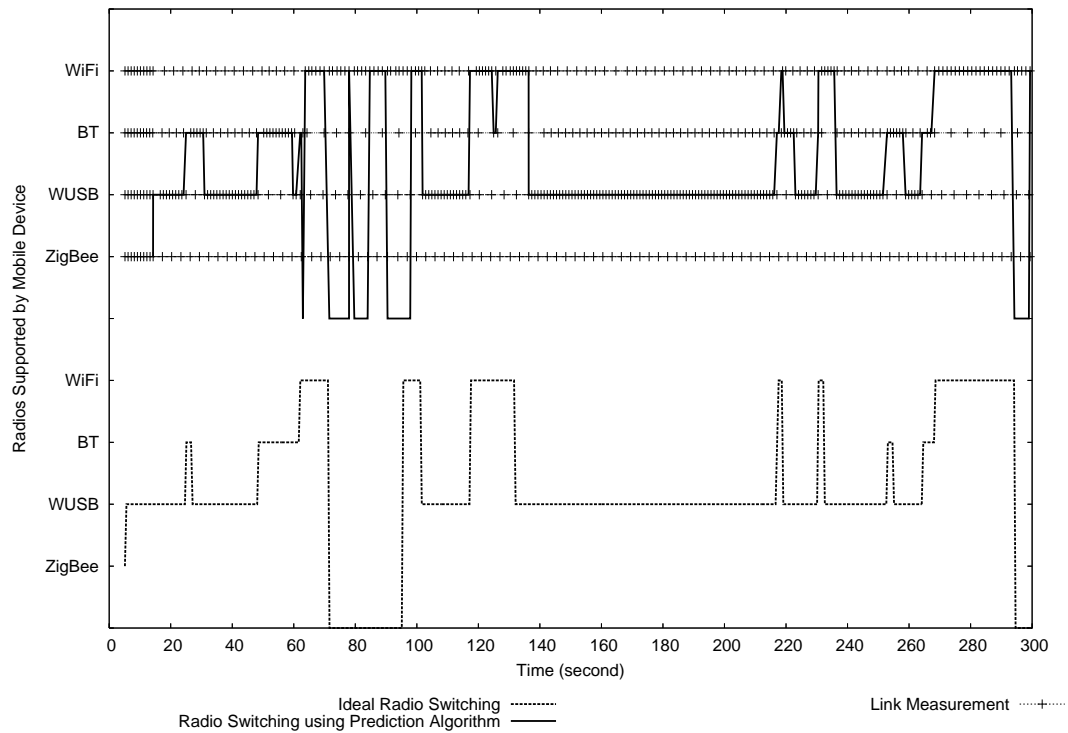Figure 3.15: Radio Switching: High Throughput Scenario

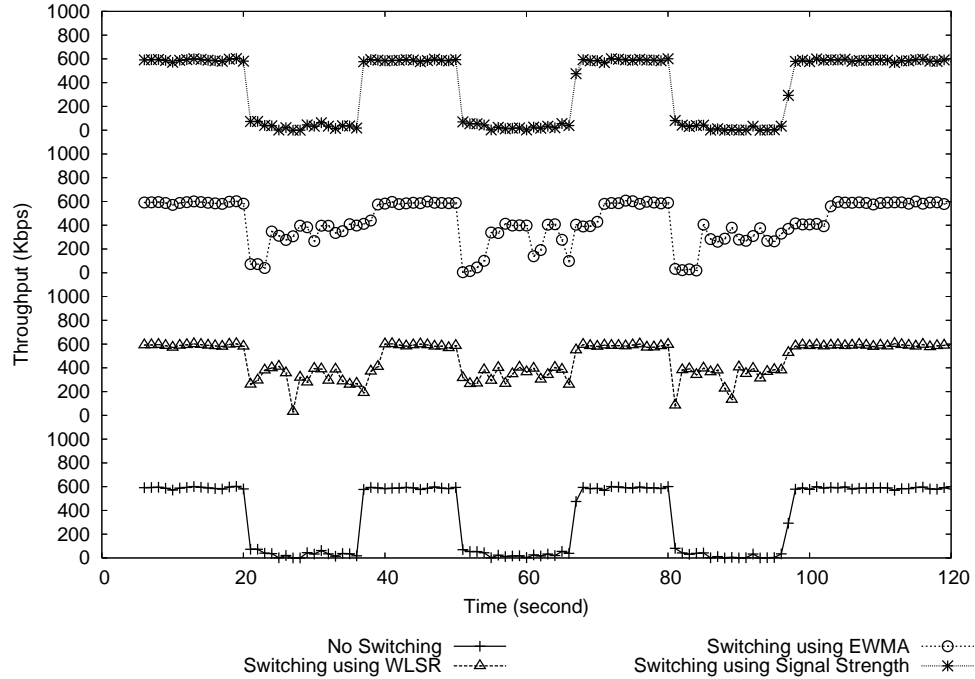Figure 3.16: Radio Switching: Power Efficient Scenario

70

Figure 3.17: Dynamic Radio Switching

by the prediction algorithm, then the system chooses the radio that delivers the highest throughput or the best power savings, depending on user preference.

The first scenario we consider is when the user prefers high throughput. While the multi-radio devices communicate, 20 pairs of WiFi nodes transmit nearby, simulating intensive interference for a 15 second interval, at 20s, 50s, and 80s. The simulation runs for 120 seconds. The multi-radio devices should be able to switch to Bluetooth when the WiFi radio becomes unavailable.

Fig. 3.17 compares the performance of radio switching when using either the WLSR, EWMA, or signal strength prediction algorithms. We include the case for no radio switching (staying with WiFi the entire time) to illustrate the overall benefit of using both radios. Both the WLSR and EWMA algorithms improve performance significantly, whereas switching based on signal strength has almost no effect. Because the measured signal strength still satisfies the transceiver capture level, it does not handle interference well.

71

Prediction Method	None	WLSR	EWMA	Signal Strength
Total MBytes	5.33	7.07	6.52	5.33
Throughput (kbps)	358.88	475.49	438.05	358.88
% Gain	-	32.49%	22.06%	0.00%

Table 3.6: Throughput Improvement via Radio Switching

Prediction Method	None	WLSR	EWMA	Signal Strength
Power (Joules/s)	21.28	12.45	12.88	12.08
% Savings	-	-41.50%	-39.47%	-43.23%

Table 3.7: Power Savings via Radio Switching

When comparing WLSR to EWMA, WLSR has shorter latency in the prediction algorithm, allowing radio selection to happen faster. Thus, although both algorithms result in throughput dropping for a short period, using WLSR has a tangible benefit. The overall benefit is reflected in Table 3.6, which summaries the throughput gain seen by each mechanism. Using WLSR improves throughput by 32% in this case, as compared to EWMA, which gains 22%. Though not shown here, our results also show about a 10% improvement in throughput when our WLSR algorithm uses a dynamic measurement interval instead of a fixed interval.

The second scenario we consider is when the user prefers power savings. While the multi-radio devices communicate, they move out of range of the Bluetooth radio for a period of 30 seconds. This happens twice, once at 15s and again at 75s, with the simulation running for 120 seconds. We again compare the three prediction algorithms.

As shown in Table 3.7, all three algorithms provide significant power savings when compared to using the WiFi radio the entire time. In this case, using signal strength alone works very well, but the WLSR algorithm is almost as good. Since WLSR also handles interference well, these results show it is a good fit for a multi-radio system.

3.12 Conclusion and Future Work

In this paper, we present a link quality prediction algorithm using Weighted Least Square Regression (WLSR). The algorithm allows mobile devices with multiple radios to statistically predict link quality based on a series of past measurements. Because the algorithm is sensitive to regression weights and the measurement window size, we use a simulation to determine good settings for these parameters. We also develop a method to compute an ideal prediction curve so that we can compare prediction accuracy of various algorithms.

Our simulation results demonstrate that the WLSR algorithm is able to predict link quality accurately and stably in a frequently changing mobile environment. The prediction algorithm is more accurate than alternative algorithms, and the overhead caused by the link measurements is negligible in terms of throughput and power consumption.

We further show that the overhead of periodic link measurements can be reduced using a dynamic measurement interval. This allows the system to conserve power by measuring radios less frequently when they are not active and not likely to be used in the near future.

We demonstrate the utility of our prediction algorithm by using it in a radio switching architecture that enables multi-radio devices to switch to a different radio when the current one becomes unavailable due to mobility or interference. This combination allows devices to increase throughput or lower power consumption, based on user preference.

Our future work will focus on efficient radio switching algorithms. In this paper, our system switches to the most preferred radio whenever it is available, but this can result in frequent switching when radios are sporadically available. We plan to develop a switching algorithm that is responsive to application requirements but avoids frequent switches.

Chapter 4

Autonomous and Intelligent Radio Switching for Heterogeneous Wireless Networks

As wireless devices continue to become more prevalent, heterogeneous wireless networks – in which communicating devices have at their disposal multiple types of radios – will become the norm. Communication between nodes in these networks ought to be as simple as possible; they should be able to seamlessly switch between different radios and network stacks on the fly in order to better serve the user. To make this a possibility, we consider the challenging problems of when two communicating devices should decide to switch to a different radio, and which radio they should choose. We design an *Autonomous and Intelligent Radio Switch* (AIRS) decision algorithm that uses predicted radio availability and user profiles to choose the best available radio for two adjacent devices. The decision algorithm uses several parameters to avoid switching radios too frequently. We use a simulation study to evaluate the best settings for several parameters, then show that the AIRS system performs better than several alternative algorithms. AIRS is able to provide dynamic, but stable radio switching, while balancing the competing objectives of high throughput and low power consumption.

4.1 Introduction

As wireless devices continue to evolve, systems that support multiple radios are becoming increasingly common, because no single wireless technology provides the desired functionality in all situations. Cellular technology provides coverage over a wide area, but phone manufacturers are adding WiFi interfaces so that users can browse the web at a WiFi hotspot, with lower connection charges and possibly higher speeds. Likewise, laptops and cellphones, in addition to WiFi or cellular interfaces, have Bluetooth interfaces for exchanging data directly with other devices or peripherals when other network interfaces may be unavailable, too cumbersome, or consume too much power.

Likewise, wireless networks are likely to be composed of heterogeneous devices in the future. Mesh networks will need multiple radios, so they can communicate with mobile devices that may switch among different radios to conserve power or provide greater throughput. Ad hoc networks will be composed of many heterogeneous devices, and will need to find ways to adapt to radio availability when these devices move. In both cases, devices ought to be able to seamlessly switch between available radios on the fly in order to provide continuous access to available services. Communication ought to "just work", rather than requiring the user to be involved.

One of the key challenges for a heterogeneous wireless network is deciding when to switch radios and which radio to choose. In a multi-hop network, a flow may span several hops, and each pair of adjacent devices in the flow may experience different amounts of interference, mobility, and competing traffic. Hence, the radio switching decisions for a given flow can be decomposed into a series of negotiations between adjacent nodes. For each pair of nodes, several radios may be available, so the devices must choose the one that will provide the best performance. This type of radio switching is typically classified as a *soft, vertical handover*, meaning that

multiple radios are available and that each radio typically has a different network stack.

In this paper, we develop an *Autonomous and Intelligent Radio Switching* decision algorithm that has several unique features. First, it takes as input the predicted link quality of each radio link, rather than using only current measurements of availability. Second, it also takes as input user preference, so that it can make decisions based on whether the user wants to optimize throughput or battery power. Third, it can choose the best available radio according to preference ranking (based on throughput or power savings) or by calculating expected utility, which provides a balance between throughput and power. Finally, the algorithm includes mechanisms to avoid the overhead of frequently switching radios when their availability is sporadic.

We evaluate the AIRS decision algorithm using a simulation study of heterogeneous wireless devices. First, we determine the appropriate settings of several parameters that help the decision algorithm to avoid frequent switches. We illustrate its effectiveness by showing how the decision algorithm avoids using radios that are only sporadically available, as well as ignoring brief periods of unavailability for a preferred radio. Finally, we show that the algorithm provides better throughput and power savings compared to alternative algorithms.

4.2   Related Work

The concept of seamless handoff between different wireless interfaces has been explored in a number of contexts. Network layer approaches typically assume an IP stack for all interfaces, and try to preserve IP connectivity as hosts move [45, 2, 57, 56, 55, 48, 41]. Session layer approaches operate above the transport layer, while still making radio switching transparent to the application layer [6, 50, 19, 18, 20, 4, 44]. Switching at the session layer enables devices to utilize many different types of ra-

dios. However, much of the work in this area is very preliminary, with many problems not yet addressed.

Several decision algorithms have been developed for deciding when to perform a handover or which interface to use for a particular flow. Singh et al. describe how to optimally assign flows to different access networks, assuming that all interfaces are always available, but characterized by variable delay and bit rate [44]. Wang et al. describe a handoff system that allows users to express policy about what is the "best" wireless system at the current moment, with the goal of balancing network load among networks with similar performance [51]. Handoffs are only performed if the network has been consistently available for some time. Chen et al. propose a vertical handoff decision making scheme using a score function on three criteria: expense, link capacity, and power consumption [6]. A few projects have proposed decision algorithms based on fuzzy logic and neural networks [57, 21, 49]. Much of this work reacts to current network conditions, rather than predicting future availability.

4.3   Radio Switching Decision Algorithm

Our decision algorithm is part of a larger Autonomous and Intelligent Radio Switching (AIRS) system [27]. The goal of this system is to leverage radio diversity and keep the user connected to available network services using the "best" available interface at any given moment.

As illustrated in Figure 4.1, the AIRS system is composed of four key modules. The *Radio Preference Evaluation* module dynamically maintains an ordered preference list for each of the wireless interfaces, based on user preference, the application's QoS requirement, and the current status of the device's battery [8]. It allows the user to select one of three profiles: "high throughput", "power efficient", and "adaptive". The latter choice optimizes for throughput when battery power is high, then gradually switches to more power efficient interfaces as battery power starts to
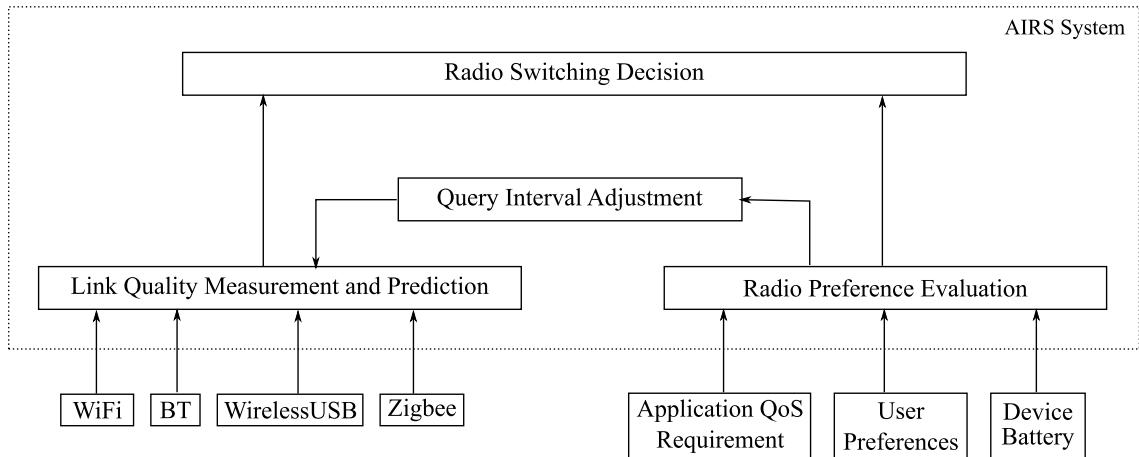
Figure 4.1: AIRS System

decrease. The *Link Quality Measurement and Prediction* module uses periodic measurements of each interface to predict the availability and quality of each radio in the near future [10]. The *Query Interval Adjustment* module adjusts how frequently queries are made, based on the past performance of the interface and its placement in the preference list [9].

In this paper, our focus is on the *Radio Switching Decision* module, which determines which radio should be used and when the handoff should be made to this radio. This module takes into consideration the predicted quality of each interface as well as the ordered preference list from other modules of the system. The prediction, $P_{avail}(R_i)$, is given as a percentage chance that the radio for interface i will meet application QoS requirements in the near future. In AIRS the prediction must be greater than 50% in order for the system to consider that link to be available, and thus eligible to be chosen by the decision module.

The decision module makes a distinction between two types of radio switching. An *upgrade* occurs when a more desirable radio becomes available and the active interface is superseded. A *downgrade* occurs when the active connection becomes unavailable and the connection must switch to a less desirable radio.

4.3.1 Downgrade Switching

Figure 4.2 shows the decision algorithm for a downgrade; this algorithm is executed whenever the AIRS system receives a new periodic link measurement (and hence a new availability prediction) for the active radio, R_a, that is currently being used by a connection.

At the start of this algorithm, a hysteresis parameter, h_a, for the interface is initialized to a positive value, e.g. 15%. The initial value of the hysteresis parameter determines how badly a link may perform before the system will downgrade. By initializing this to, for example, 15%, the system allows a link's predicted availability to reach 35% before a downgrade takes place. The hysteresis decreases when a link is currently unavailable, so that the system can react more quickly when a radio suddenly cannot be used. We later use simulations to determine a good initial value for this parameter.

The first step in the algorithm is to determine whether the current radio is predicted to be available in the near future; this is true if $P_{avail}(R_a)$ is greater than 50%. If the radio will be available, the algorithm next checks whether the current measurement indicates the link is available right now. This is necessary because the predicted availability is based on many previous measurements, whereas the current availability is based on only the most recent measurement. A link may have a long history of availability, then suddenly become unavailable (e.g. due to mobility) or may suffer transient interference, which should be ignored. The challenge is to adapt quickly to changes in link status while remaining stable during periods of transient interference.

To handle this uncertainty, our decision algorithm relies on a combination of predicted availability, plus hysteresis. If the active radio is not currently available, h_a is reduced by 5%, otherwise it is reset to its initial value. If, in the original step, the link is predicted not to be available in the future, then h_a is reduced by 5% and
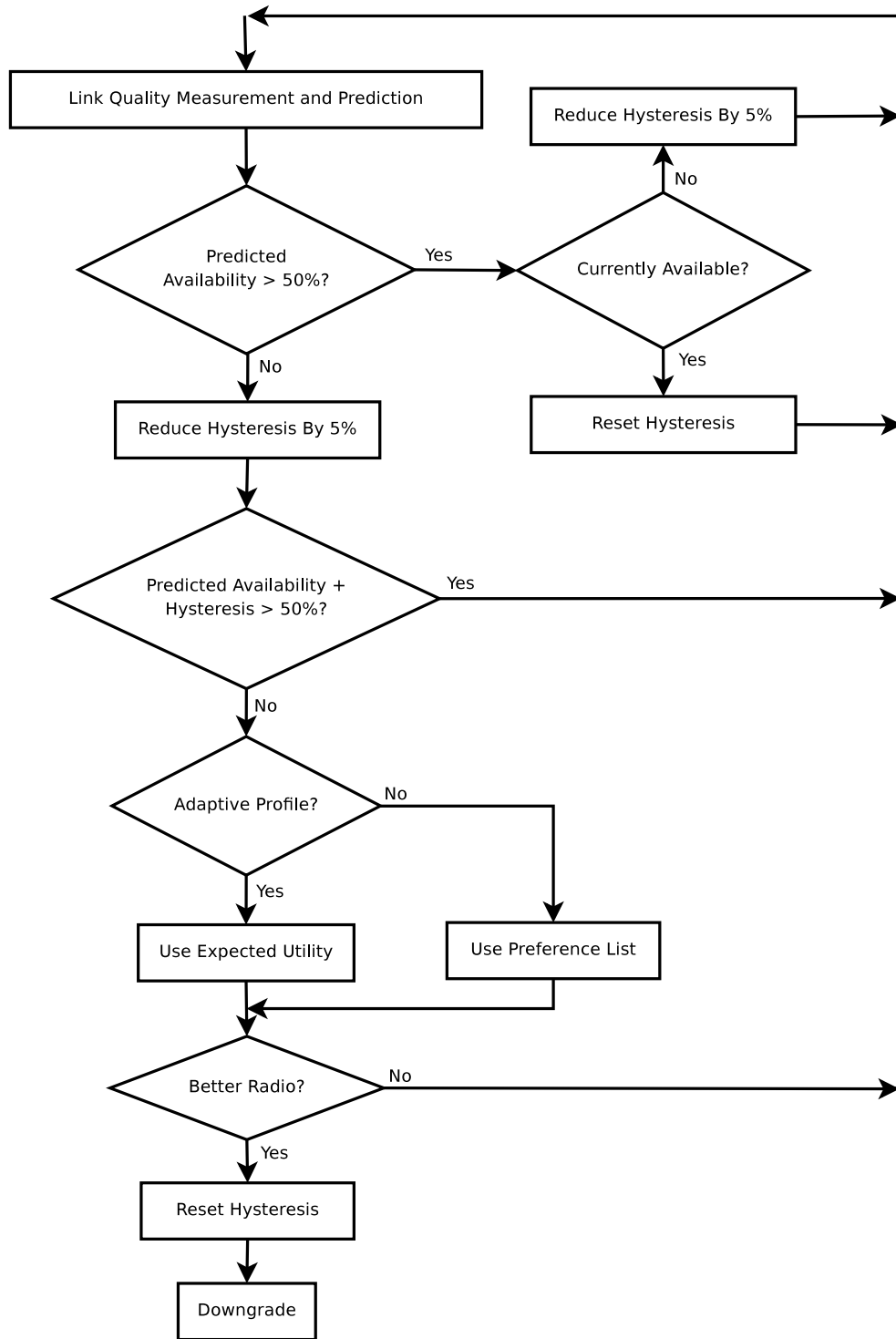
Figure 4.2: Downgrade Decision Algorithm

a new check is made by determining whether $P_{avail}(R_a) + h_a$ is greater than 50%. If the interface is not available by this measure, then a downgrade is initiated.

To initiate the downgrade, the decision module first selects the best available interface. If the user has selected the "adaptive" profile, the preferred interface is the one with the highest expected utility, $U_{expected}(R_i)$, calculated as:

$$U_{expected}(R_i) = U_{social}(R_i) \times P_{avail}(R_i) \qquad (4.1)$$

The social utility is derived from user preference on the two communicating devices and the the characteristics of the link, such as delay and bandwidth. If the user instead prefers to optimize throughput or power consumption exclusively, then the best available interface is selected from an ordered preference list. Once a new radio is selected, algorithm resets h_a for the active radio and switches to the new radio.

4.3.2 Upgrade Switching

Figure 4.3 shows the decision algorithm for an upgrade, which is executed whenever the AIRS system receives a periodic link measurement and prediction for an inactive radio, R_i. At the start of this algorithm, a link verification parameter, v_i, for the interface is initialized to a positive value, e.g. 4. This parameter indicates how many additional measurements must be taken before the interface is considered as a candidate for an upgrade switch. Thus a value of 4 would indicate that the link must be available for four consecutive measurement periods before it is used.

The first step in the algorithm is to determine whether this (inactive) radio is available. If it is available, the algorithm next checks whether this radio has a higher expected utility, or higher preference ranking, than the current radio. If this interface is preferred, v_i is decreased by 1. Once v_i reaches zero, this link may be used for an upgrade switch.
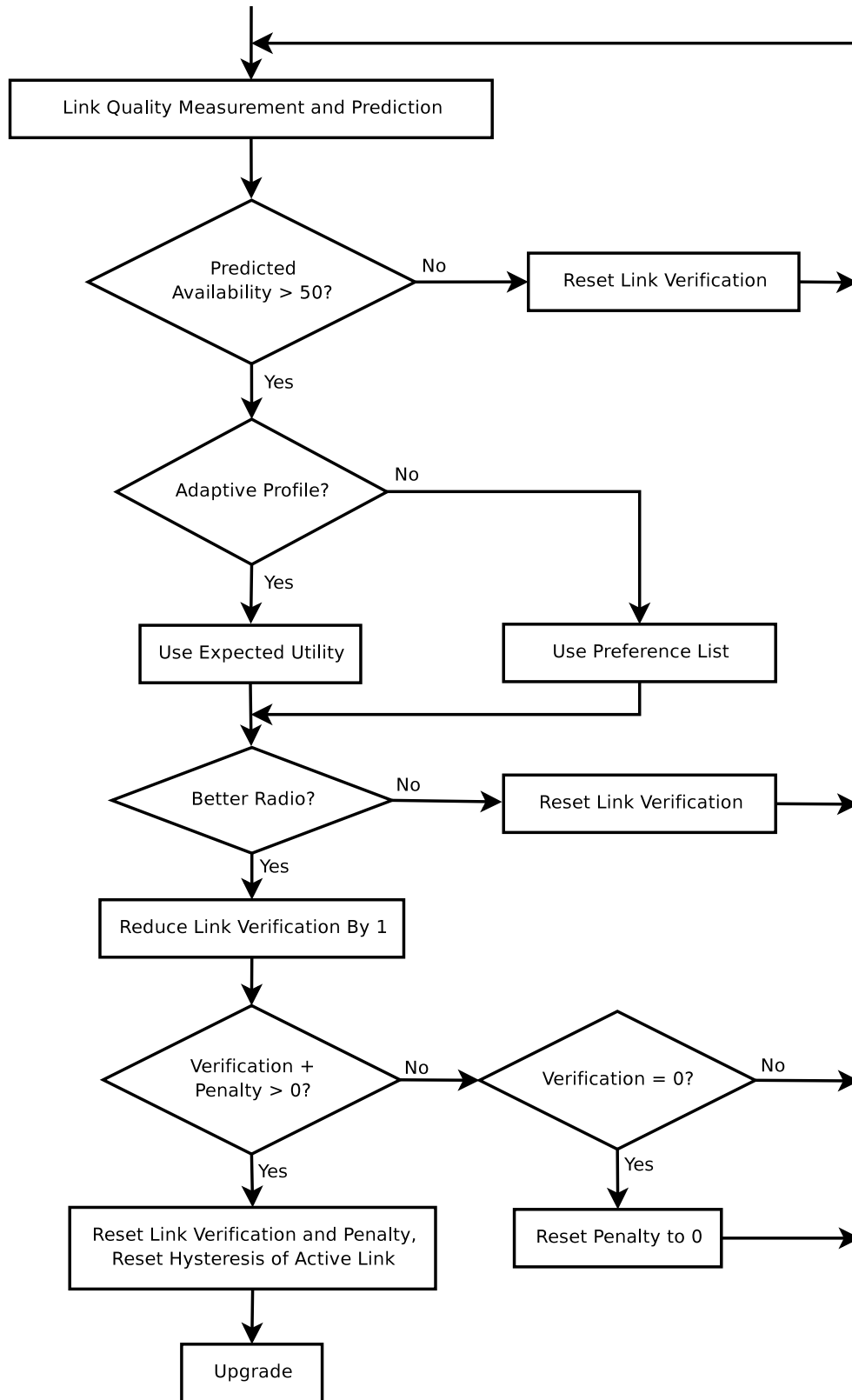
Figure 4.3: Upgrade Decision Algorithm

The decision algorithm also uses a penalty parameter, p_i, to avoid radios that have failed previously. This parameter is set to one if the radio becomes unavailable within 3 seconds after it was used for an upgrade. It is reset back to zero once the radio has been available again for a consecutive number of measurements (equal to the initial value of v_i). If both v_i and p_i reach zero, and this radio is the most preferred available radio, then an upgrade is initiated.

We later use simulations to determine a good initial value for the link verification parameter.

4.4 Performance Evaluation

We perform a simulation study using ns-2.28 to calibrate the radio switching decision algorithm's parameters and to evaluate its effectiveness. Our simulation implements the entire AIRS system, since the decision module depends on input from both the link quality prediction module and the radio preference module, as introduced in Section 4.3.

As mentioned earlier, we decompose the radio switching problem to a negotiation between adjacent devices. Our topology thus consists of two adjacent mobile devices as shown in Figure 4.4, each with WiFi, Bluetooth, WirelessUSB, and ZigBee radios. The two devices use a VoIP application running over UDP, though the choice of application and transport protocol does not affect our results. In addition, our topology includes 10 pairs of Bluetooth devices and 10 pairs of WiFi devices. In our experiments, we use mobility, plus interference from the additional devices, to vary the channel quality for each of the radios.

4.4.1 Evaluation Metrics

To evaluate the effectiveness of our decision algorithm, we measure the *average switch latency.* For a downgrade, this is the difference between the time when the active radio
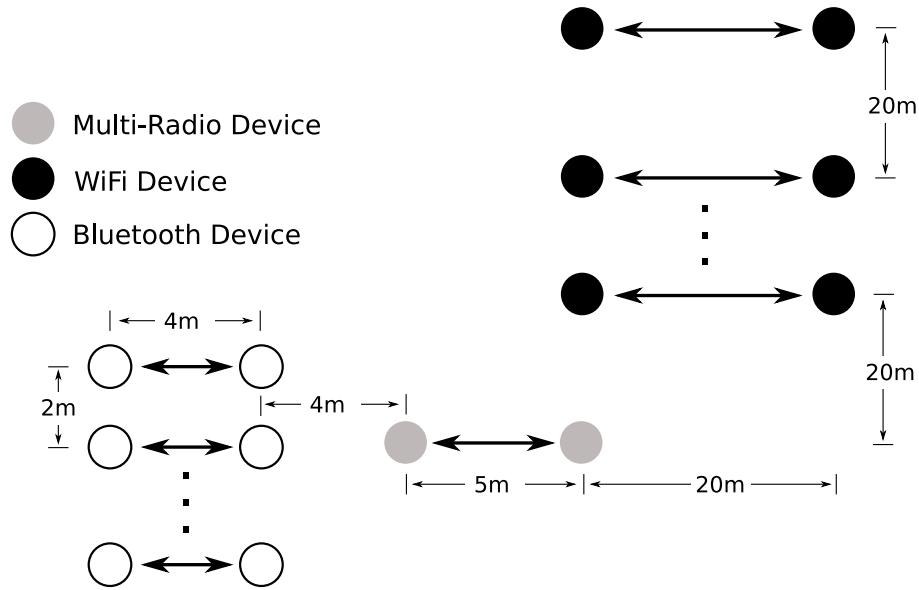
Figure 4.4: Simulation Topology

becomes unavailable and when the downgrade switch occurs. For an upgrade, this is the difference between the time when the switch occurs and the time when the new radio becomes available. A naive decision algorithm could immediately switch to a different radio whenever the current one becomes unavailable or a better one becomes available. Some of the switches however are incorrect and should be avoided. Thus latency must be balanced by the need to eliminate frequent switches.

We would like to minimize such frequent switches, since each radio switch incurs some latency and overhead. The primary device sends out a switch request, and the secondary device responds with either a switch accept or a switch reject. The primary device waits for this response with a certain timeout, and retries the request several times if needed. We consider a frequent switch to be one that occurs within 3 seconds of the last switch, and report the *frequent switches* as a percentage of the total switches.

We also measure *goodput*, which is the application level throughput averaged throughout the entire simulation, and *battery power* to determine the effect of radio switching on application performance.

4.4.2 Decision Algorithm Parameters

We perform a variety of simulations with different scenarios to determine the proper settings for the hysteresis and link verification parameters. In selecting scenarios for these experiments, our goal is to have enough variation in radio availability so that the switching model parameters we choose will work across a wide range of possible situations. Accordingly, we use scenarios that include times when the radio is continuously available, times of periodic unavailability, and times of high volatility. We also use both the high throughput and power efficient user profiles, with preference ranking as the selection criteria:

1. **High throughput Scenario**: One of the multi-radio devices stays stationary, and the other moves in and out of Bluetooth coverage at a speed of $4m/s$. There are 10 pairs of WiFi nodes and 5 pairs of Bluetooth nodes generating interfering traffic. During the simulation, the devices may need to downgrade from WiFi to Bluetooth, or from Bluetooth to WirelessUSB, and upgrade from WirelessUSB to Bluetooth, or from Bluetooth to WiFi.

2. **Power efficient scenario**: One of the multi-radio devices stays stationary, and the other moves in and out of WirelessUSB coverage at a speed of $11m/s$. There are 10 pairs of Bluetooth nodes generating interfering traffic. During the simulation, the devices may need to downgrade from WirelessUSB to Bluetooth, or from Bluetooth to WiFi, and upgrade from WiFi to Bluetooth, or from Bluetooth to WirelessUSB.

In both profiles, the simulation scenarios are generated randomly. Mobility and interference each occur 3 - 5 times during the simulation, with a random starting time. The exact numbers of mobility and interference occurrences are randomly determined in simulation setup. The duration of each mobility/interference lasts from 30 - 50 seconds, which is also generated randomly in the setup stage.

We run each simulation for 300 seconds, and average our results over 50 replications. We use a typical battery life for PDAs, 10 watt-hours. We compare the AIRS decision algorithm to a naive radio switching algorithm that uses the same prediction inputs and radio preference rankings as introduced in Section 4.3, but switches as soon as possible whenever a better radio is available.

For downgrades, there is a clear tradeoff between the average switch latency and frequent switches, as shown in Figure 4.5. Each symbol on the graph represents a different combination of the hysteresis parameter (ranging from 0.05 to 0.25) and the link verification parameter (ranging from 1 to 4). The points that represent the same hysteresis setting cluster together, since the link verification parameter does not affect downgrade switching. With just 15% hysteresis, the percentage of frequent switches decreases to less than 5%, while the average switch latency increases from about a half a second to 2 seconds. More hysteresis can nearly eliminate frequent switches, but at the cost of another second and a half of latency. This tradeoff is clearly better than the naive algorithm, which switches quickly but frequently. Based on this evaluation, we use 15% for this parameter in the remaining simulations.

A similar tradeoff exists for upgrades and the link verification parameter, shown in Figure 4.6. The number of frequent switches decreases and the latency increases as the verification parameter increases, and a setting of at least 4 reduces the percentage of frequent switches to below 5% again. The naive algorithm is again limited to frequent but fast switches. In the case of upgrade switching, an active radio is already being used, so it is less critical to have low latency in this situation than for a downgrade. We thus use a setting of 4 for the link verification parameter in the remaining simulations.

We next compare AIRS using expected utility to AIRS using preference ranking with the adaptive profile. In this profile, the preferences of the radios are dynamically adjusted based on the battery usage, changing from high throughput preference
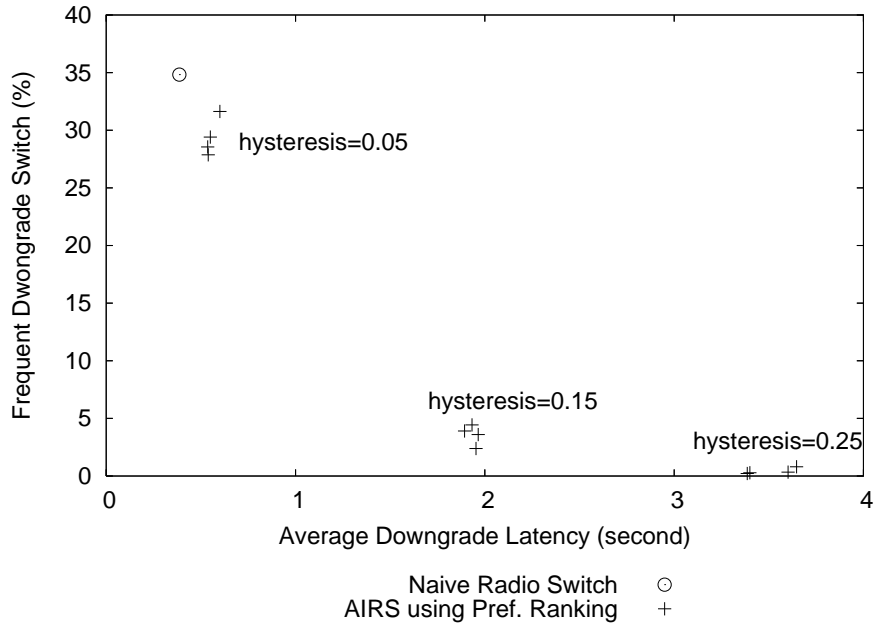
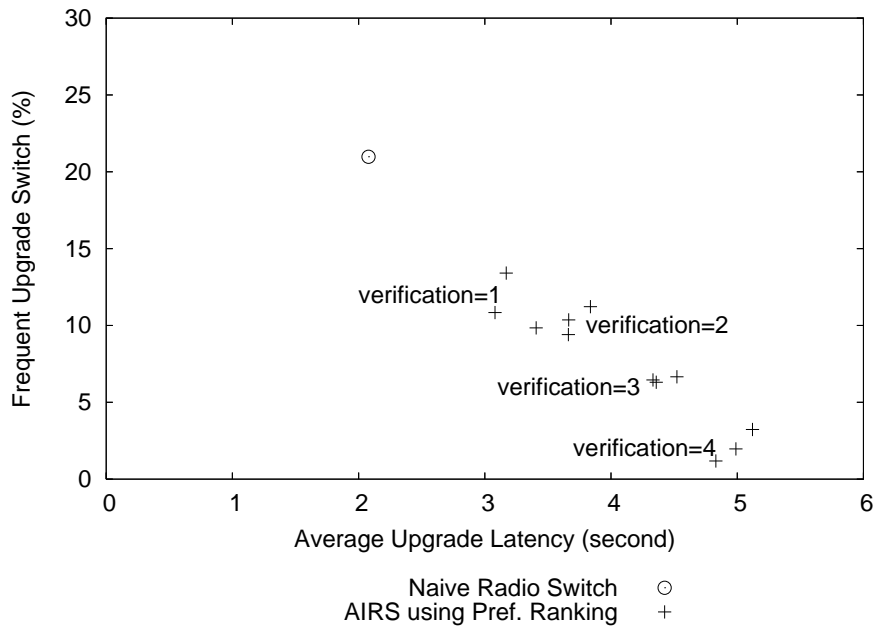Figure 4.5: Downgrade Switching Hysteresis Tradeoff

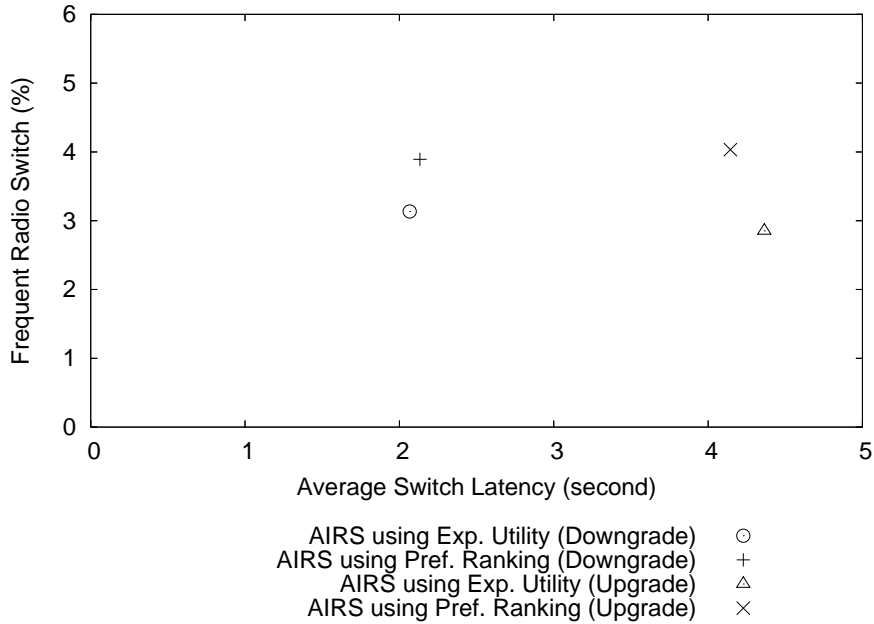Figure 4.6: Upgrade Switching Link Verification Tradeoff

Figure 4.7: Effect of Applying Expected Utility

to power efficient preference as the battery level goes down. We make our selection based on the expected utilities of support radios as discussed in Section 4.3, and we use a hysteresis parameter of 15% and a link verification parameter of 4. Simulation scenarios are generated randomly, and the results are averaged over 50 replications.

Figure 4.7 shows the effect of using expected utility for both downgrade and upgrade scenarios. The expected utility algorithm further decreases the frequent radio switch ratio, without impacting the average switch latency.

To illustrate how effective the decision algorithm can be in avoiding frequent switches, we run an additional experiment that causes frequent disruptions in the availability of one of the radios. Figure 4.8(a) shows the measured availability for each radio on the two devices. The radios are shown from bottom to top in order of highest power consumption to lowest power consumption. The WiFi radio is always available; the Bluetooth radio is available at first, but then drops off; the WirelessUSB radio is volatile (device moves in and out of the WirelessUSB service range frequently); and the ZigBee radio is always unavailable.

The important part of this scenario is that, for the power efficient user profile, the WirelessUSB radio is the most preferred radio, since the ZigBee radio is always unavailable. There is one period where WirelessUSB is mostly available, with spikes where it is ineffective, and another period where it is mostly available, with spikes of activity. These periods are caused by the radio moving in and out of range, or perhaps by interference.

In this scenario, as shown in Figure 4.8(b), the naive algorithm switches very frequently, which can cause interruptions in the conversation and additional overhead. The AIRS decision algorithm, however, allows for much more stable selection of radios, using lower powered options when they are mostly available, and switching to WiFi only when necessary. The AIRS algorithm effectively reduces frequent switches in both downgrade and upgrade.

4.4.3 Performance Comparison

We evaluate the AIRS decision algorithm by comparing it to several alternative algorithms. The naive switching algorithm, discussed previously, switches whenever there is a more preferred radio available, and uses the AIRS prediction module to determine availability. The packet loss algorithm switches to the next best radio whenever a single link-layer frame is lost using the current radio. The timeout algorithm switches to the next best radio whenever the transport layer times out (about 10 seconds). Each of these algorithms uses preference ranking lists, so we test them against the AIRS algorithm using both the high throughput and power efficient user profiles. In addition, we compare these alternatives to the AIRS decision making model that uses expected utility with adaptive user profile.

We again generate simulation scenarios randomly, including periods of interference and availability to affect the different radios. To achieve a high likelihood that there is at least one radio available at all times, we reduce the mobility and
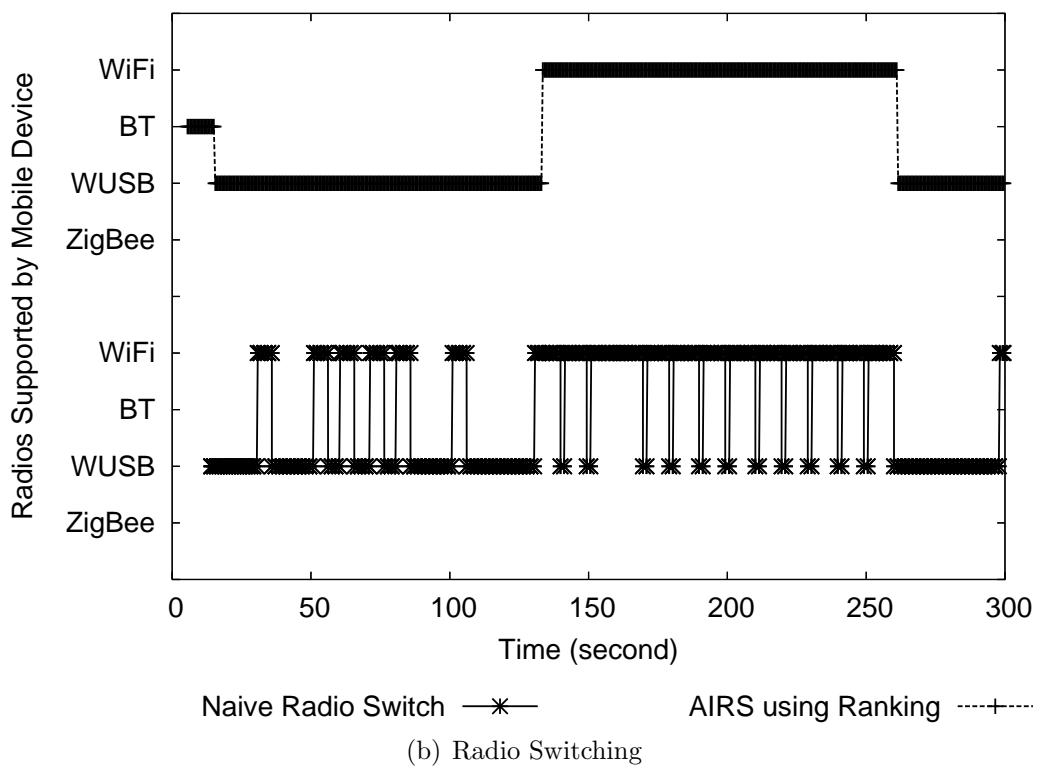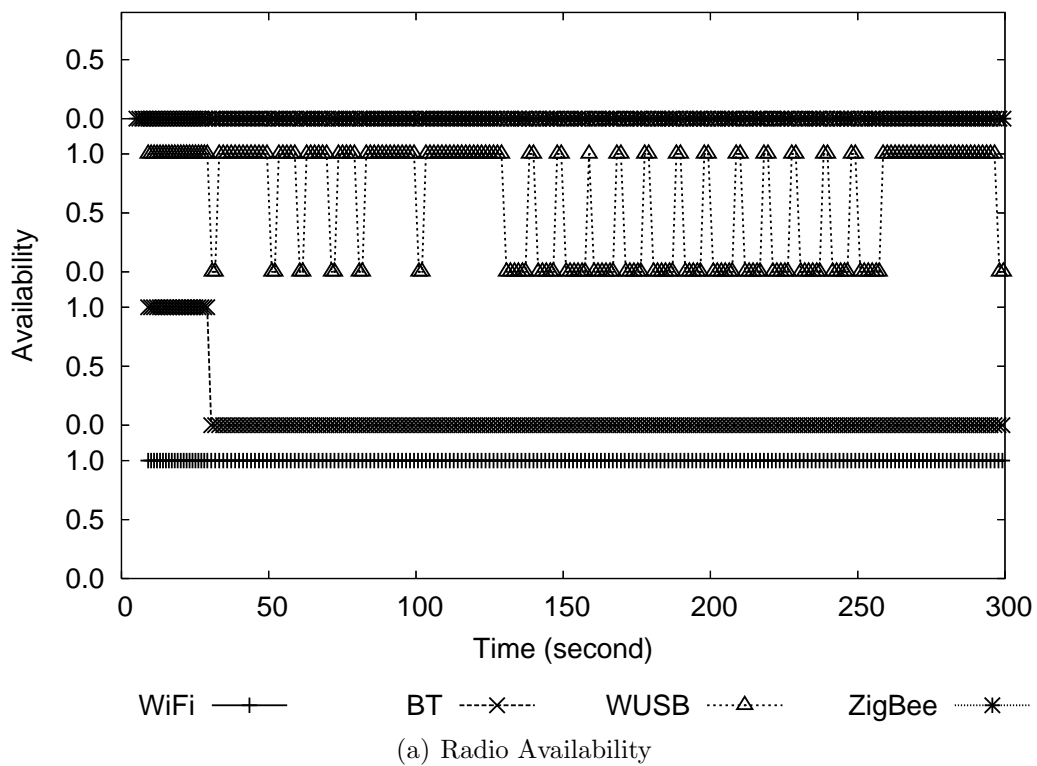
(a) Radio Availability

(b) Radio Switching

Figure 4.8: Handling Volatility

91

interference occurrence to 2 - 3 times, and reduce the mobility duration to 10 - 20 seconds. We set the device battery randomly in the range 35 - 65 watt-hours; at the low end of this range the battery is not sufficient to use the highest powered radio for the duration of the simulation. Each simulation runs for 300 seconds, and we average the results over 50 replications.

As shown in Figure 4.9, the AIRS system using expected utility provides the best tradeoff between battery power and goodput. The scenarios for the high throughput profile are clustered on the bottom of the graph. For each algorithm, the power of the device is nearly depleted, eventually decreasing the averaged goodput. The AIRS algorithm gets the most goodput because many frequent radio switches can be avoided using hysteresis and link verification. Likewise, the scenarios for the power efficient profile are clustered near the top of the graph. Most of these actually get higher goodput, plus longer battery life, because there are radios that provide good enough throughput while consuming less power. The AIRS algorithm again does the best of these. Finally, the AIRS system using the adaptive profile gets the most goodput, while still preserving much of the battery. This is because radios preferences are dynamically adjusted according to battery power. The result shows that the adaptive profile, along with expected utility in the decision algorithm, is a good choice for balancing these two objectives.

The results shown in Figure 4.9 are affected by the time when switches actually happen. The AIRS decision making model is able to switch when necessary based on accurate link quality prediction, while avoiding frequent switches. Hence, connections are carried over different radios during the entire simulation. The naive algorithm often makes incorrect switches, and sacrifices goodput and power due to such intensive switches. Switching based on packet loss causes even more incorrect switches. Switching based on timeout is another extreme case. Switch actions are severely delayed, making the connection break for a period of time, which affects the

AIRS using Exp. Utility ⚹
AIRS using Pref. Ranking (High Throughput Mode) ◆
AIRS using Pref. Ranking (Power Efficient Mode) ◇
Naive Radio Switch (High Throughput Mode) ■
Naive Radio Switch (Power Efficient Mode) □
Packet Loss Radio Switch (High Throughput Mode) ●
Packet Loss Radio Switch (Power Efficient Mode) ○
Timeout Radio Switch (High Throughput Mode) ▲
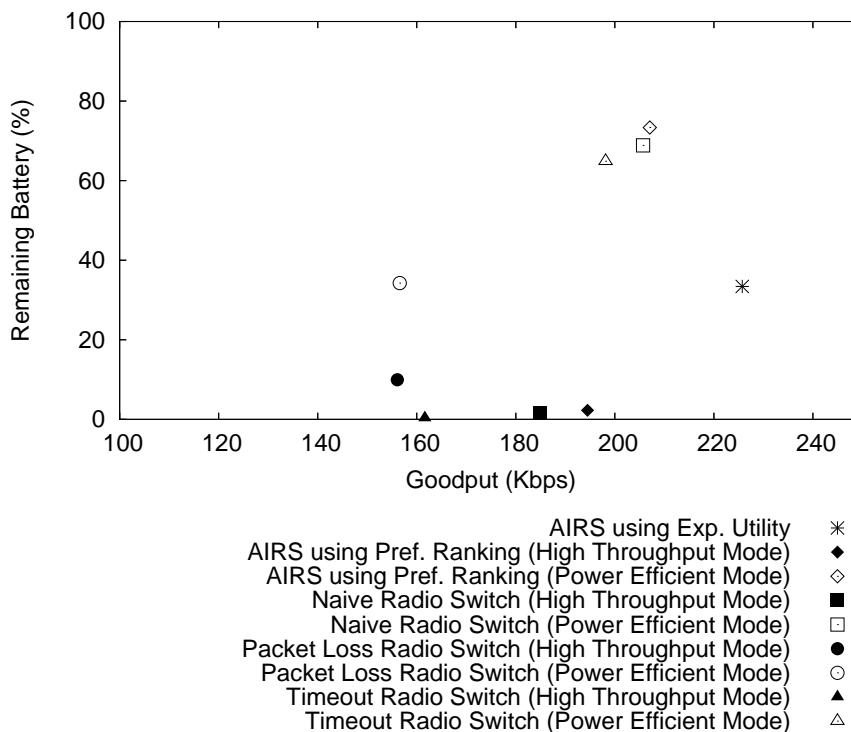Timeout Radio Switch (Power Efficient Mode) △

Figure 4.9: Goodput and Battery Life Tradeoff

goodput and is unacceptable from users' perspective. In each scenario, devices move randomly within the coverage of WiFi. Timeout rarely occurs on WiFi with only the impact of interference. Once the device switches to WiFi, it may stick to this radio even in power efficient user profile. Hence, battery power is easily depleted when using the timeout algorithm.

To illustrate how AIRS works when using the adaptive profile, we randomly selected one simulation and show how the system dynamically chooses a radio over time to balance power and throughput. Figure 4.10(a) shows the measured availability for each radio on the two devices, with WiFi, Bluetooth, WirelessUSB, and ZigBee from top to bottom.

Figure 4.10(b) shows how the AIRS system changes the active radio over time, using the adaptive profile. Initially the system uses WiFi, since it offers the highest throughput and the battery power is high. As the battery becomes depleted, it switches to Bluetooth, then WirelessUSB when Bluetooth is unavailable for a short
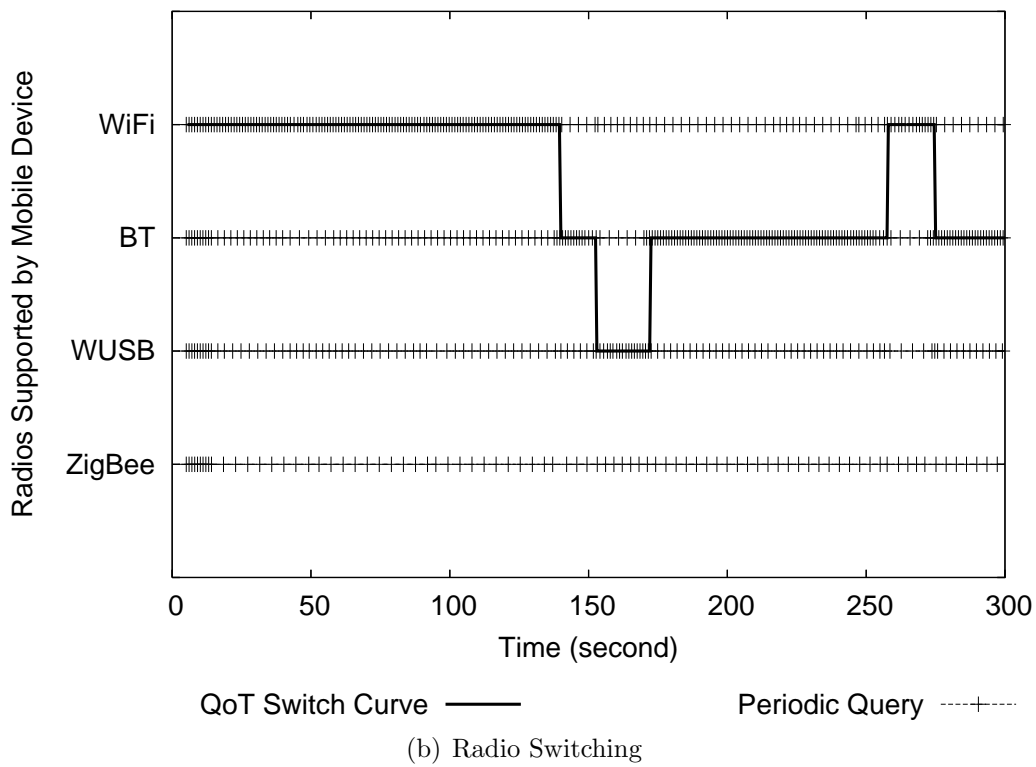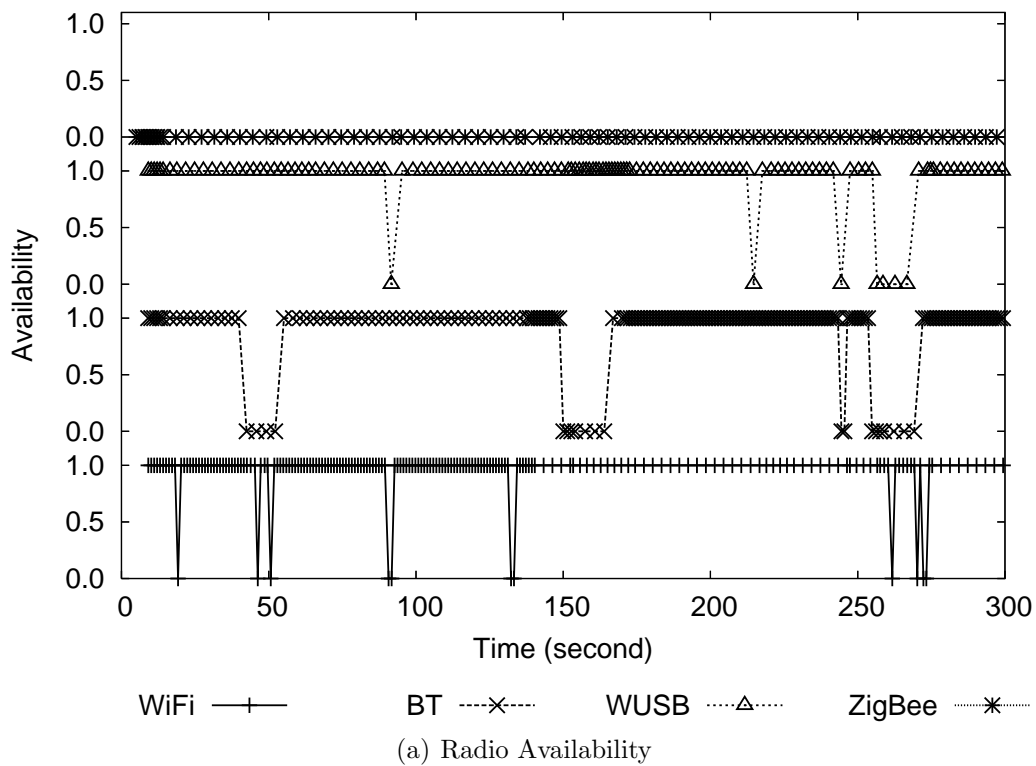
(a) Radio Availability

(b) Radio Switching

Figure 4.10: AIRS Radio Switching Scenario

period of time. It then continues to use Bluetooth to preserve battery power, except for a short period near the end where it must switch to WiFi to maintain connectivity. Radio switching using AIRS is smart, timely, and stable.

4.5 Conclusion and Future Work

The AIRS system is a key component of a heterogeneous wireless network. For any pair of communicating nodes, the system is able to dynamically choose the best available radio, while balancing throughput and power. The system uses several mechanisms to avoid frequent switching, and offers the user the choice of three different performance profiles.

A number of areas remain for future work. In a wireless network with many systems using AIRS simultaneously, additional mechanisms may be needed to provide stability and ensure that the network-wide utility is optimized. In addition, an implementation of AIRS would provide valuable insight into its feasibility and performance.

Chapter 5

Summary

5.1 Contribution

The AIRS System is able to intelligently and transparently improve connectivity by exploiting the inherent heterogeneity of multi-radio devices, and make the supported radios to serve best where needed. The AIRS system fulfills the functionality of QoTBrain, and makes QoT more intelligent, adaptive, and efficient. For any pair of communicating devices, the system is able to select the most preferred radio considering both connecting users' preference in a mutually beneficial manner, accurately predict link quality with efficient measurement intervals, and dynamically choose the best available radio, while balancing throughput and power. Users can achieve high goodput and long battery life as the AIRS system is applied for handoff management.

5.2 Delivered Artifact

The follows artifacts were generated in the ns-2.28 simulator [14]:

- Implemented energy consumption control at device scope

- Implemented an interference detection module for radio switching

- Extended WirelessUSB module to support packet exchange in both directions and queuing functionality at network layer

- Extended ZigBee to support communicating within multi-radio context

- Implemented a radio preference evaluation and negotiation scheme for intelligent radio selection

- Implemented an efficient link quality prediction model for dynamic radio selection

- Implemented a seamless and adaptive decision making system

- Automatic testing using Python and OTcl

5.3 Future Work

Three issues should be discussed in the future work to further enhance the AIRS system.

1. Extend the AIRS system with efficient radio switching algorithms for multi-hop connections. Our current work focuses on intelligent radio switching between adjacent devices. In a heterogeneous wireless network, a flow may span several hops. The radio switching algorithm should be able to improve the performance of the entire flow. Future research may focus on coordination between hops, packet routing, and the overall efficiency.

2. Extend the AIRS system to support multiplexing of multiple network stacks, both wireless and wired interfaces. Our current work only considers integrating the wireless radios supported by the device via dynamic radio switching, assuming mobility support is always desired. Sometimes when user is not in motion, wired interface is available and can provide even better link quality. An advanced AIRS system should be able to switch between all kinds of interfaces to fully leverage the inherent heterogeneity and to further improve the communication experience. Future research may focus on evaluating the preference of wired interface and predicting its link quality.

3. Extend the AIRS system to provide improved and stable connections for all connecting pairs within the entire network. Many systems may use AIRS simultaneously in a wireless network. Additional mechanisms may be needed to provide stability and to ensure that the network-wide utility is optimized. Future research may focus on radio switching stability and fast convergence, so that multiple pairs will not switch to the same radio simultaneously and repetitively.

Bibliography

[1] Paramvir Bahl, Atul Adya, Jitendra Padhye, and Alec Walman. Reconsidering Wireless Systems with Multiple Radios. *ACM SIGCOMM Computer Communication Review*, 34(5), 2004.

[2] M. Baker, X. Zhao, and J. Stone. Supporting Mobility in MosquitoNet. In *USENIX Technical Conference, San Diego, California*, 1996.

[3] Shannon B. Barnes, Ryan W. Woodings, and Charles D. Knutson. Transport Discovery in Wireless Multi-Transport Environments. In *IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2003.

[4] G. A. Di Caro, S. Giordano, M. Kulig, D. Lenzarini, A. Puiatti, and F. Schwitter. A Cross-Layering and Autonomic Approach to Optimized Seamless Handover. In *International Conference on Wireless on Demand Network Systems and Service (WONS)*. IEEE, 2006.

[5] P.M.L. Chan, Y.F. Hu, and R.E. Sheriff. Implementation of Fuzzy Multiple Objective Decision Making Algorithm in a Heterogeneous Mobile Environment. In *Proceedings of the IEEE Wireless Communications and Networking Conference, Orlando, FL, USA*, March, 2002.

[6] Ling-Jyh Chen, Tony Sun, Benny Chen, Venkatesh Rajendran, and Mario Gerla. A Smart Decision Model for Vertical Handoff. In *The 4th ANWIRE International Workshop on Wireless Internet and Reconfigurability (ANWIRE)*, 2004.

[7] R. Chellappa Doss, A. Jennings, and N. Shenoy. A Review of Current Mobility Prediction Techniques for Ad Hoc Networks. In *Proceedings of the 4th IASTED International Multi-Conference Wireless and Optical Communications*, 2004.

[8] Qiuyi Duan, Lei Wang, Charles D. Knutson, and Michael A. Goodrich. Axiomatic Multi-Transport Bargaining: A Quantitative Method for Dynamic Transport Selection in Heterogeneous Multi-Transport Wireless Environments. In *IEEE WCNC*, 2006.

[9] Qiuyi Duan, Lei wang, Charles D. Knutson, and Daniel Zappala. Efficient Link Quality Prediction for Wireless Devices with Multiple Radios. Technical report, Mobile Computing Lab., Brigham Young University, 2008.

[10] Qiuyi Duan, Lei wang, Charles D. Knutson, and Daniel Zappala. Link Quality Prediction for Wireless Devices with Multiple Radios. In *Proceedings of IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks*. IEEE, June, 2008.

[11] Heidi R. Duffin, Charles D. Knutson, and Michael A. Goodrich. Negotiating Transport Selection Between QoT-Enabled Devices. Technical report, Mobile Computing Lab., Brigham Young University, 2003. Tech. Rep. TR-04-01.

[12] Heidi R. Duffin, Charles D. Knutson, and Michael A. Goodrich. Prioritized Soft Constraint Satisfaction: A Qualitative Method for Dynamic Transport Selection in Heterogeneous Wireless Environments. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, 2004.

[13] Bain / Engelhardt. *Introduction to Probability and Mathematical Statistics*. Thomson Learning, January, 2000.

[14] Kevin Fall and Kannan Varadhan. *The ns Manual (formerly ns Notes and Documentation)*. The VINT Project - A Collaboration between researchers at UC Berkeley, LBL, USC/ISI, and Xerox PARC, September 6, 2005.

[15] Kroly Farkas, Theus Hossmann, Lukas Ruf, and Bernhard Plattner. Pattern Matching Based Link Quality Prediction in Wireless Mobile Ad Hoc Networks. In *ACM MSWIM*, 2006.

[16] Gregor Gaertner, Eamonn ONuallain, Andrew Butterly, Kulpreet Singh, and Vinny Cahill. 802.11 Link Quality and Its Prediction – An Experimental Study. In *Personal Wireless Communications*, 2004.

[17] Michael Gerharz, Christian de Waal, Matthias Frank, and Peter Martini. Link Stability in Mobile Wireless Ad Hoc Networks. In *Proceedings of the 27th Annual IEEE Conference on Local Computer Networks (LCN)*. IEEE, 2002.

[18] S. Giordano, M. Kulig, D. Lenzarini, A. Puiatti, F. Schwitter, and S. Vanini. WiOptiMo: Optimised Seamless Handover. In *Wireless Personal Multimedia Communications (WPMC)*. IWS, 2005.

[19] S. Giordano, D. Lenzarini, A. Puiatti, and S. Vanini. WiSwitch: Seamless Handover between Multi-Provider Networks. In *International Conference on Wireless on Demand Network Systems and Service (WONS)*. IEEE, Jan. 2005.

[20] Silvia Giordano, Davide Lenzarini, and Alessandro Puiatti. Demonstrating Seamless Handover of MultiHop Networks. In *Realman in conjunction with ACM MobiHoc*. ACM, 2006.

[21] Bing Hongyan, He Chen, and Jiang Lingge. Intelligent Signal Processing of Mobility Management for Heterogeneous Networks. In *International Conference on Neural Networks and Signal Processing*, volume 2, pages 1578 – 1581. IEEE, Dec. 2003.

[22] Raj Jain. *Art of Computer Systems Performance Analysis Techniques for Experimental Design, Measurement, Simulation, and Modeling.* John Wiley and Sons Inc, April, 1991.

[23] Shengming Jiang, Dajiang He, and Jianqiang Rao. A Prediction-Based Link Availability Estimation for Mobile Ad Hoc Networks. In *IEEE INFOCOM*, 2001.

[24] Steven D. Kaehler. Fuzzy Logic Tutorial. The Newsletter of the Seattle Robotics Society, March, 1998.

[25] E. Brent Kelly. Quality of Service In Internet Protocol (IP) Networks. Technical report, Wainhouse Research, 2002. Prepared for the International Communications Industries Association To support Infocomm 2002.

[26] Abtin Keshavarzian, Elif Uysal-Biyikoglu, Falk Herrrnann, and Arati Manjeshwar. Energy-efficient Link Assessment in Wireless Sensor Networks. In *IEEE Infocom*, 2004.

[27] Charles D. Knutson, Heidi R. Duffin, Jeffrey M. Brown, Shannon B. Barnes, and Ryan W. Woodings. Dynamic Autonomous Transport Selection in Heterogeneous Wireless Environments. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, 2004.

[28] Dhananjay Lal, Arati Manjeshwar, Falk Herrmann, Elif Uysal-Biyikoglu, and Abtin Keshavarzian. Measurement and Characterization of Link Quality Metrics in Energy Constrained Wireless Sensor Networks. In *Global Telecommunications Conference (Globecom)*. IEEE, 2003.

[29] C. C. Lee. Fuzzy Logic in Control Systems: Fuzzy Logic Controller. *IEEE Transactions on Systems, Man and Cybernetics*, 20(2):404–435, April. 1990.

[30] W. C. Y. Lee. Estimate of Channel Capacity in Rayleigh Fading Environment. *IEEE Transactions on Vehicular Technology*, 39(3):187–189, Aug. 1990.

[31] Zhong Li. *Fuzzy Chaotic Systems: Modeling, Control, And Applications.* Springer-Verlag New York Inc, June, 2006.

[32] A. Bruce McDonald and Taieb Znati. A Path Availability Model for Wireless Ad-Hoc Networks. In *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 1999.

[33] Jim Metzler. A LAN Perspective on Wireless WAN Requirements. Technical report, Leverage Technology and Talent for Success, 2003.

[34] Mobile Computing Lab, Brigham Young University, Provo, Utah. *Quality of Transport (QoT) Protocol Specification*, 1.0 edition, 2002.

[35] Abhinay Muthoo. *Bargaining Theory with Applications.* Cambridge University Press, 1999.

[36] Jr. J. F. Nash. The Bargaining Problem. *Econometrica*, 18:155-162, 1950. Reprinted in Classics in Game Theory, H. W. Kuhn, ed.

[37] Transition Networks. Quality of Service (QoS) in High-Priority Applications. Technical report, Transition Networks, 2003.

[38] D. Panigrahi, C. Chiasserini, S. Dey, R. Rao, A. Raghunathan, and K. Lahiri. Battery Life Estimation of Mobile Embedded Systems. In *The 14th IEEE International Conference on VLSI Design*, 2001.

[39] Vijay T. Raisinghani and Sridhar Iyer. Cross Layer Design Optimization in Wireless Protocol Stacks. *Submitted to Elsevier Science*, Mar. 2, 2003.

[40] D. Rakhmatov, S. Vrudhula, and D. A. Wallach. Battery Lifetime Prediction for Energy-Aware Computing. In *In Proceedings of the International Symposium on Low Power Electronics and Design*, Augest 2002.

[41] P. Rodriguez, R. Chakravorty, I. Pratt, and S. Banerjee. MAR: A Commuter Router Infrastructure for the Mobile Internet. In *ACM MOBISYS*, June 2004.

[42] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach.* Prentice Hall, 1995.

[43] Sanjay Shakkottai, Theodore S. Rappaport, and Peter C. Karlsson. Cross-layer Design for Wireless Networks. In *Communications Magazine, IEEE*, volume 41, pages 74 – 80, Oct. 2003.

[44] Jatinder Pal Singh, Tansu Alpcan, Piyush Agrawal, and Varun Sharma. An Optimal Flow Assignment Framework for Heterogeneous Network Access. In *the proceedings of IEEE Int. Symp. on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, June 2007.

[45] M. Stemm and R. H. Katz. Vertical Handoffs in Wireless Overlay Networks. *Mobile Networks and Applications*, 4, 1999.

[46] William Su, Sung-Ju Lee, and Mario Gerla. Mobility Prediction and Routing in Ad Hoc Wireless Networks. *Int. J. Netw. Manag.*, 11(1):3–30, 2001.

[47] Tim Szigeti and Christina Hattingh. Quality of Service Design Overview. Technical report, Cisco Press, 2004.

[48] Jean Tourrilhes and Casey Carter. P-Handoff: A Protocol for Fine Grained Peer-to-Peer Vertical Handoff. In *The 13th IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications.* IEEE, 2002.

[49] Nishith D. Tripathi, Jeffrey H. Reed, and Hugh F. Van Landingham. Pattern Classification Based Handoff using Fuzzy Logic and Neural Nets. In *International Conference on Communications (ICC)*, pages 1733–1737. IEEE, June 1998.

[50] Narayanan Venkitaraman, Jay Almaula, Anwar Haneef, and Jayanth Mysore. Session Aware Network Controlled Interface Selection for Multi-homed Hosts. In *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC).* IEEE, 2004.

[51] H. J. Wang, R. H. Katz, and J. Giese. Policy-Enabled Handoffs across Heterogeneous Wireless Networks. In *Second IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '99).* IEEE, 1999.

[52] Gerhard Weiss, editor. *Multiagent Systems : A Modern Approach to Distributed Artificial Intelligence.* Cambridge, Mass. MIT Press, 1962, C1999.

[53] Ye Wen, Rich Wolski, and Chandra Krintz. Online Prediction of Battery Lifetime for Embedded and Mobile Devices. Technical report, Computer Science Department, University of California, Santa Barbara, June 2003.

[54] Ryan Winfield Woodings. Quality of Transport (QOT): Dynamic Transport Switching in Wireless Personal Area Networks. Master's thesis, Computer Science Department, Brigham Young University, December 2002.

[55] M. Ylianttila, R. Pichna, J. Vallstrom, J. Makela, A. Zuhedi', P. Krishnamurthy, and K. Pahlavan. Handoff Procedure for Heterogeneous Wireless Networks. In *Global Telecommunications Conference (Globecom)*. IEEE, 1999.

[56] Jukka Ylitalo, Tony Jokikyyny, Tero Kauppinen, Antti J. Tuominen, and Jaakko Laine. Dynamic Network Interface Selection in Multihomed Mobile Hosts. In *36th Hawaii International Conference on System Sciences (HICSS)*. IEEE, 2002.

[57] Wenhui Zhang. Handover Decision Using Fuzzy MADM in Heterogeneous Networks. In *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2004.

[58] Fang Zhu and J. McNair. Optimizations for Vertical Handoff Decision Algorithms. In *IEEE Wireless Communications and Networking Conference (WCNC)*, volume 2, pages 867–872, Mar. 2004.