A Proximity-based Routing Scheme for Multi-Hop Wireless Networks

Masato Saito

Faculty of Policy Management

Keio University

5322 Endo, Fujisawa Kanagawa 252-8520 JAPAN

Submitted in partial fulfillment of the requirements for the degree of Bachelor

Advisors:

Professor Hideyuki Tokuda
Professor Jun Murai
Associate Professor Hiroyuki Kusumoto
Associate Professor Osamu Nakamura
Instructor Masaki Minami

Abstract of Bachelor's Thesis

A Proximity-based Routing Scheme for Multi-Hop Wireless Networks

This thesis describes the design and implementation of a proximity-based routing protocol, called OR2. OR2 is an adaptive route path tuning technique for wireless mobile ad hoc networks. In OR2, active route paths adapt dynamically to node mobility without any link failures based on the local link quality. Most conventional routing protocols accommodate the change of network topology only when the link fails. Unless the movement of intermediate nodes leads to any link failures, they cannot adapt to the network topology even if other routes with less hop count become available. In contrast, OR2 skips the upstream node in a *proximity* area that indicates the nearness of two communicating nodes and continues to shorten an active route as possible.

We have implemented OR2 as an extension to DSR on FreeBSD. Some simple experiments demonstrate that OR2 is effective in enhancing TCP throughput and reducing end-to-end delay for all relevant flows. Furthermore, we propose the two novel algorithms to generate the realistic node mobility: the *random oriented model* and the *random escape model*. Simulation studies for several scenarios of node mobility and traffic flows reveal that adding dynamic path shortening to DSR and AODV significantly reduces the number of routing packets and the packet latency in most cases.

Masato Saito

Faculty of Policy Management

Keio University

卒業論文要旨　　2001年度(平成13年度)

A Proximity-based Routing Scheme for Multi-Hop Wireless Networks

「モバイルアドホックネットワークにおける動的経路短縮機構の設計と実装」

携帯端末の急速な普及や無線通信技術の発達に伴い、移動通信端末同士を自律的に、かつ動的に接続するモバイルアドホックネットワーク (Mobile Ad hoc NETwork: MANET) が、広く注目を集めている。MANETは、インフラストラクチャ非依存やマルチホップ通信、動的な端末の移動と言った現在のインターネットとは対称的な特徴を持つ次世代無線通信ネットワークを想定しているため、既存のIPルーティングプロトコルとは異なる様々なMANETルーティングプロトコルが提案されている。

しかし、これらの既存のルーティングプロトコルは、通信中のコネクション切断を契機としてのみ通信経路の再構築を行うため、移動端末の動的な移動を考慮した通信経路の効率的な再構築機構を持っていない。本研究では、移動端末同士の通信品質の強さに基づく"proximity"という概念を導入し、移動端末の動的な移動に対応した経路短縮を行う経路制御プロトコルを提案する。

本プロトコルを FreeBSD OS (Operating System) 上に実装し、経路短縮における効果とプロトコル動作の評価のための予備実験を行った。さらに、移動端末数を50台とした大規模アドホックネットワークを想定して、シミュレータ上における本プロトコルの評価結果を示す。同時に、シミュレータ上における移動端末の現実的な移動パターンを生成する新しいアルゴリズムを提案し、生成されたシナリオにおける本プロトコルの評価結果も示す。本プロトコルを既存の DSR (Dynamic Source Routing) や AODV (Ad hoc On-demand Distance Vector routing) といったプロトコルに組み込むことで、ルーティングパケットによるオーバヘッドの減少、通信端末間におけるパケット転送遅延の縮小といった点において大幅な改善が見られた。

慶應義塾大学　総合政策学部

斉藤 匡人

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

The recent growth in the number of wireless alternatives to gain access to the Internet has been remarkable. These technologies enable desktops, laptops and hand-held Personal Digital Assistants (PDAs) (e.g., Palm [22]) to gain access to information sources on the Internet without any tethered connection. Figure 1.1 shows the current tendency of the Internet access by several main devices in Japan [16].

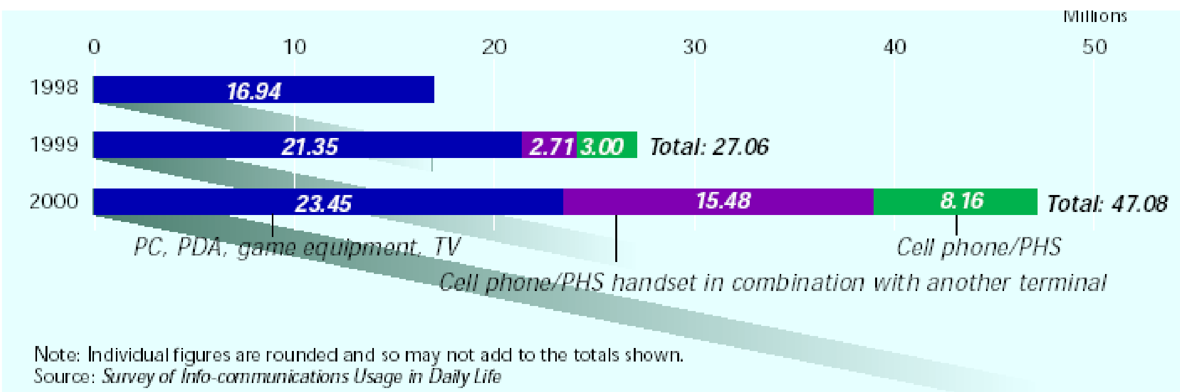
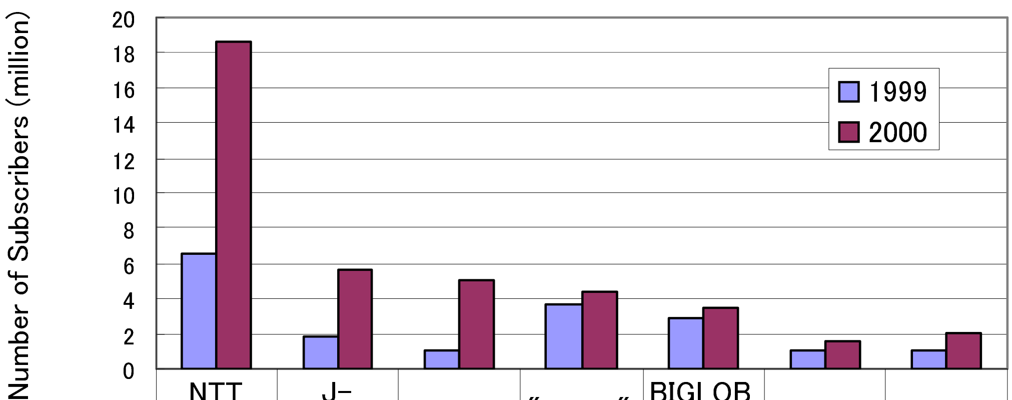


Figure 1.1: How the Internet is accessed.
Sources: Ministry of Public Management, Home Affairs, Posts and Telecommunications, Japan

The access by cell phones and Personal Handy-phone System (PHS [26]) handsets are becoming increasingly popular. Furthermore, as the recent remarkable characteristic of

Japanese telecommunications, Figure 1.2 shows that the subscribers of the Internet Service Providers (ISPs) for cell phones are much more than that of the main leading ISPs in Japan [17]. Surprisingly, NTT DoCoMo already recorded 30 million subscribers on Dec. 25, 2001 [20].



	NTT DoCoMo	J-Phone	au	"@nifty"	BIGLOBE	So-net	OCN
1999	6.51	1.79	1.04	3.66	2.85	1.09	1.02
2000	18.57	5.66	5.01	4.42	3.49	1.6	2.06

Figure 1.2: Subscribers of main ISPs in Japan.
Sources: Ministry of Public Management, Home Affairs, Posts and Telecommunications, Japan

Not only in Japan but in all the world, the number of mobile internet users are rapidly increasing. These rapidly growing trends of the wireless internet access users are shown in wireless LAN areas as well as in cellular phone areas. Currently, in wireless internet access areas, there are a number of wireless access methods enabling the internet access even while moving around. Figure 1.3 shows the examples (e.g., 802.11b [10], Bluetooth [30], IMT-2000 [11]) and each characteristic position in the matrix of the covering mobility and the raw bandwidth. We could expect that the advancement of technologies realize the high data throughput and high node-mobility simultaneously in the near future.

When wireless devices are becoming ubiquitous around our environments, we believe that

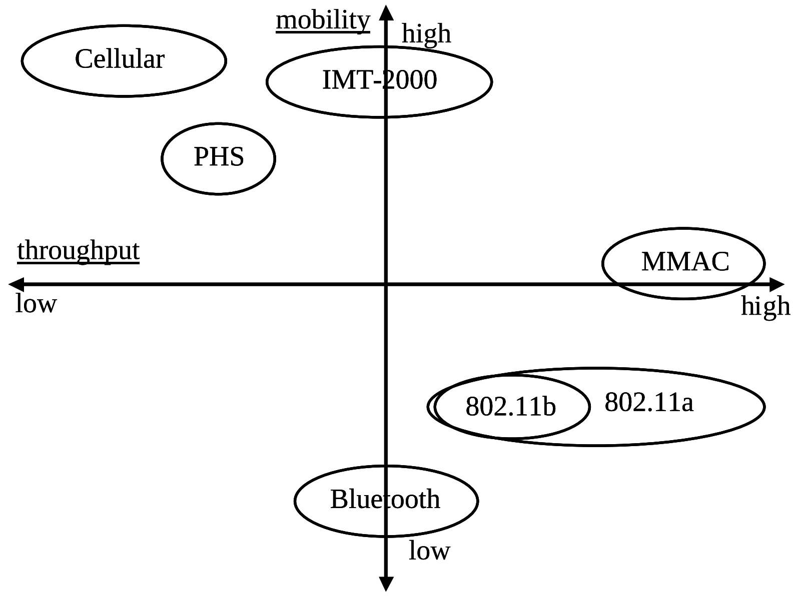


Figure 1.3: Positions of Wireless Access Systems

these wireless devices automatically connect each other and make the distributed novel type of mobile networks called mobile ad hoc networks. Recently there has been a renewed interest in the mobile ad hoc networks due to the common availability of relatively low-cost laptops and palm-tops with wireless interfaces. The interest is also caused by growing enthusiasm in running TCP/IP protocol suites in dynamic wireless environments without specific infrastructures for emergency disaster situations after earthquake or a hurricane, or the battlefield.

There are several scenarios where ad hoc networks are useful. One major application is a military-use communication in a battlefield where a centralized configuration is difficult (e.g., in the enemy jungle). Another application is emergency communication in disaster areas. In addition to these large-sized applications, we can use ad hoc networks when several people have meetings with computers that are equipped with wireless interfaces. Also, it can be interesting research for supporting intelligent transport systems and sensor networks.

However, since the network nodes are mobile, ad hoc networks will typically have dynamic topology which will have serious effects on network characteristics. Network functions such as routing, address allocation, authentication, and authorization must be designed to cope with dynamic and temporary network topology. We should also think the mechanisms to provide the Internet connectivity with mobile ad hoc networks. Additionally, wireless network nodes will often be limited battery powered, which limits the capacity of CPU, memory and bandwidth. This will require network functions that are resource effective. Also, the wireless media will affect the behavior of the network due to fluctuating link bandwidth resulting from relatively high bit error rates.

In this thesis, we identify and solve the fundamental problems of the routing protocols **not adapting node mobility** in mobile ad hoc networks.

1.2 Challenges and Research Goal

As popularity for mobile computing increases, cooperative communications with wireless devices are becoming an attractive technology. A key challenge to succeed in such communications is adapting to node mobility. A mobile ad hoc network is a group of mobile computing devices (nodes) which communicates with each other using multi-hop wireless links. It does not require any stationary infrastructure such as base stations. Each node in the network can act as both a host and a router forwarding data packets to other nodes.

One important issue for achieving efficient network resource utilization is to update route information reactively depending on a change of network topology and connectivity. Since node mobility in an ad hoc network causes frequent, unpredictable and drastic changes to the network topology, it is especially important for communicating nodes to grasp the change of the network topology and find an efficient route between two communicating nodes. A number of research for mobile ad hoc networks has focused on the development of their reactive routing protocols (e.g., DSR [3], AODV [25], LAR [15], SOAR [28]). The key advantage behind reactive (on-demand) protocols is the reduction of routing overheads so that on-demand routing protocols maintain only active paths to those destinations to which data must be sent. Minimizing the routing overhead is effective in such a dynamic environment of ad hoc networks due to limited available bandwidth, unpredictable nodes mobility, battery outages, interference and high bit error rates.

These above on-demand routing protocols accommodate route changes only when an active path is disconnected. They cannot adapt to the change of network topology even if another route with less hop count becomes available by the movement of intermediate nodes unless any link is disconnected. DSR protocol [3] only has the mechanism that shorten an active path which is not driven by link failures but by overhearing packets by operating the network interfaces in promiscuous receive mode. This promiscuous mode, however, requires greater CPU cycles, power consumption and sending delay due to overheard packets. In contrast to the conventional protocols, we propose Optimized Re-Routing (OR2) algorithm that tunes up an active path adaptive to node mobility without any link disconnection based on Smoothed Signal-to-Noise Ratio (SSNR) as a link quality value indicator. OR2's adaptation to node mobility leads to the reduction of a hop count and path delay which significantly improves the performance of Transmission Control Protocol (TCP) flows.

Our research goal is that **OR2 should have the improving effects in enhancing the throughput and reducing end-to-end delay for all relevant flows (e.g.,** ***TCP*****,** ***UDP*****, etc) while the routing overhead incurred with OR2 being sufficiently negligible**.

1.3 Structure of Thesis

The rest of this thesis is organized as follows.

Chapter 2 describes background material in the area of wireless networks and mobile ad hoc networks, and some related work in dynamically adapting to node mobility in mobile ad hoc networks. In Chapter 3, we discuss the design and the detailed description of our OR2, and Chapter 4 explains the implementation of OR2 in the experimental networks and in the packet-level simulator. In Chapter 5, we present the results and the analysis of several experiments, and the improving effects of adding OR2 to DSR and AODV routing protocols. Finally, in Chapter 7, we present our conclusions and discuss several future work.

Chapter 2

Background and Related Work

This chapter describes a summary and characteristics of wireless LAN technologies. Next, We describe the overview of mobile ad hoc networks and the four main routing protocols in mobile ad hoc networks in detail. Then we discuss some protocols using link-state information and the promiscuous mode as related work.

2.1 Wireless Technology

Foremost, let us review the overview of wireless networks. The wide range of wireless networks includes in-room Infrared networks, building-wide wireless LANs, campus-area packet radio and wireless networks, metropolitan-area cellular wireless networks, regional-area wireless cable networks and broadcast satellite networks. These wireless networks are different from each other in terms of maximum raw bandwidth, channel access mechanisms, link protocols, frequency, power and covering mobility. Among them, we regard the conventional campus-area wireless LAN networks as the basis of our research target.

2.2 Wireless LAN Technology

This section describes wireless LAN technologies throughly. First, we present the overview of wireless LAN technology and then describe several important wireless technologies related to this thesis.

2.2.1 Overview

Wireless LANs are mainly in-room and in-building networks that provide maximum bandwidth between 1 and 54 Mbps over a relatively small range. Examples of these include IBM's infrared, Lucent's WaveLAN [21], IEEE 802.11b, IEEE 802.11a, Bluetooth, HyperLAN and MMAC technologies as shown in Figure 1.3. Most of these operate in the unlicensed Industrial, Scientific and Medical (ISM) bands at 915 MHz, 2.4 GHz and 5 GHz that have been set aside by the national regulations for experimental purposes.

Below, we outline some related wireless technologies briefly.

2.2.2 CSMA/CA

CSMA/CA is the channel access mechanism used by most wireless LANs in the ISM bands. For example, IEEE 802.11 [10] and WaveLAN [33] specifics use this mechanism. A channel access mechanism is one part of the protocol which specifies how the node uses the medium.

The basic principles of CSMA/CA are listen before talk and contention. This is an asynchronous message passing mechanism (connection-less), delivering a best effort service, but no bandwidth and latency guarantee. The main advantages are that it is suited for network protocol s such as TCP/IP, adapts quite well with the variable condition of traffic and is quite robust against interferences.

CSMA/CA is derived from CSMA/CD (Collision Detection), which is the base of Ethernet. The main difference is the collision avoidance : on a wire, the transceiver has the ability to listen while transmitting and so to detect collisions (with a wire all transmissions having approximately the same strength). But, even if a radio node could listen on the channel while transmitting, the strength of its own transmissions would mask all other signals on the air . So, the protocol cannot directly detect collisions like with Ethernet and only tries to avoid them.

2.2.3 WaveLAN

WaveLAN operate in either the 902-928 MHz or the 2.4-2.8 GHz ISM license-free band. WaveLAN employs a CSMA/CA MAC protocol and WaveLAN interface contains a standard Intel 82593 single-chip CSMA/CD LAN controller, custom logic for signal processing and modem control, and a custom radio transceiver. The transmitter applies DQPSK modulation to a 2 megabit/s data stream, yielding a 1 megabaud signal. This signal is further modulated by an 11 chip per bit sequence to produce an 11 MHz wide signal which is transmitted with a power antennas and multiple incoming signal paths to combat multi-path interference.

2.2.4 IEEE 802.11

When 802.11 was eventually released, 1 and 2 Mb/s was no longer considered as decent speed for Wireless LAN and people were already talking of using the 5 GHz band for higher throughput. However, the migration from 2.4 GHz to 5 GHz requires to change all nodes and does not provide back ward compatibility.

Therefore, people producing 2.4 GHz products tried to find way to extend the life of their technology (mostly WaveLAN). They cheated with the Spread Spectrum rules, and got away with it, enabling them to offer 5 and 11 Mb/s systems.

Basically, a Direct Sequence (DS) system generate signal which occupies around 22 MHz of bandwidth. They designed their 11 Mb/s system to generate signal similar to a standard DS system.

Then, they went to the Federal Communications Commission (FCC) and claimed that as their new system was generating the same type of signal as a DS system and its impact on other systems in the band was the same, it should be authorized as well. After a bit of negotiation, the FCC did accept this extension of the rule. Note that some Frequency Hopping (FH) vendors also tried to get 5 MHz FH channels in the 2.4 GHz band but failed to obtain it.

2.2.5 Wireless Network Characteristics

Networks based on wireless technologies have many characteristics in comparison to networks based on more traditional technologies such as optical fiber, coaxial cable or twisted pair wiring [8]. Wireless network characteristics can be summarized in two points:

- Wireless transmission are subject to interference from outside sources, absorption, scattering, fading, and inter-symbol interference.

- Due to mobility or intermittent interference sources the error environment will also change: Location dependent error occurrence.

2.3 Mobile Ad Hoc Networks

This section describes the overview of mobile ad hoc networks and previously proposed chief routing protocols. Then, we discuss several related work profoundly.

2.3.1 Overview

A mobile ad hoc network is a group of mobile computing devices (nodes) which communicates with each other using multi-hop wireless links. It does not require any stationary infrastructure such as base stations. Each node in the network can act as both a host and a router forwarding data packets to other nodes.

One important issue for achieving efficient network resource utilization is to update route information depending on a change of network topology and connectivity. Since node mobility in an ad hoc network causes frequent, unpredictable and drastic changes to the network topology, it is especially important for communicating nodes to grasp the change of the network topology and find an efficient route between two communicating nodes.

A number of research for mobile ad hoc networks has focused on the development of their routing protocols (e.g., DSR [3], AODV [25], TORA [23], ZRP [9], OLSR [12], TBRPF [1], LAR [15], SOAR [28]). These routing protocols can be classified into three types: *pro-active*, *reactive* and *hybrid*. Pro-active protocols attempt to continuously evaluate the routes within the network, so that when a packet needs to be forwarded, the route is already known and can be immediately used. On the other hand, reactive protocols invoke a route determination procedure on an *on-demand* basis. Lastly, hybrid protocols are mixture of reactive/pro-active scheme.

Some comparisons between these different protocols have been published [4], [13]. Both reported results based on simulations show that the reactive protocols perform significantly better than traditional pro-active protocols (DSDV [24] and ZRP) in most situations. The key advantage behind on-demand protocols is the reduction of routing overheads so that on-demand routing protocols maintain only active paths to those destinations to which data must be sent. Minimizing the routing overhead is effective in such a dynamic environment of ad hoc networks due to limited available bandwidth, unpredictable nodes mobility, battery outages, interference and high bit error rates.

2.3.2 Routing Protocols

We present the descriptions of the four remarkable proposed routing protocols for mobile ad hoc networks. These routing protocols have been submitted to the Mobile Ad-hoc Networks (MANET) working group [18] within the Internet Engineering Task Force (IETF). DSR and AODV are on-demand protocols, OLSR and TBRPF are pro-active protocols.

- DSR

  The Dynamic Source Routing protocol (DSR [3]) is a simple and efficient routing protocol designed specifically for use in multi-hop wireless ad hoc networks of mobile nodes. DSR allows the network to be completely self-organizing and self-configuring, without the need for any existing network infrastructure or administration. The protocol is composed of the two mechanisms of "Route Discovery" and "Route Maintenance", which work together to allow nodes to discover and maintain source routes to arbitrary destinations in the ad hoc network.

  The use of source routing allows packet routing to be trivially loop-free, avoids the need for up-to-date routing information in the intermediate nodes through which packets are forwarded, and allows nodes forwarding or overhearing packets to cache the routing information in them for their own future use. All aspects of the protocol operate entirely on-demand, allowing the routing packet overhead of DSR to scale automatically to only that needed to react to changes in the routes currently in use. This document specifies the operation of the DSR protocol for routing unicast IP packets in multi-hop wireless ad hoc networks.

- AODV

  The Ad hoc On-Demand Distance Vector (AODV [25]) routing protocol is intended for use by mobile nodes in an ad hoc network. It offers quick adaptation to dynamic link conditions, low processing and memory overhead, low network utilization, and determines unicast routes to destinations within the ad hoc network. It uses destination sequence numbers to ensure loop freedom at all times (even in the face of anomalous delivery of routing control messages), avoiding problems (such as 'counting to infinity') associated with classical distance vector protocols.

- OLSR

  The Optimized Link State Routing (OLSR [12]) protocol is intended for mobile ad hoc networks. The protocol is an optimization of the pure link state algorithm tailored to the requirements of a mobile wireless LAN. The key concept used in the protocol is that of multi-point relays (MPRs). MPRs are the selected nodes which forward the broadcast packets during the flooding process. This technique substantially reduces the packet overhead as compared to pure flooding mechanism where every node retransmits the packet when it receives the first copy of the packet.

  In OLSR protocol, the information flooded in the network 'through' these multi-point relays is also 'about' the multi-point relays. Hence, a second optimization is achieved here by minimizing the 'contents' of the control packet flooded in the network. Hence, as contrary to the classic link state algorithm, only a small subset of links with the neighbor nodes are declared instead of all the links. This information is then used by OLSR protocol for route calculation, and therefore the routes contain only the MPRs as the intermediate nodes from Source to Destination. It results in providing the optimal routes (in terms of number of hops), and hence another optimization. The protocol is particularly suitable for the large dense networks as the technique of multi-point relays works well in this context.

- TBRPF

  The Topology Broadcast based on Reverse Path Forwarding (TBRPF [1]) is a proactive, link-state routing protocol designed for mobile ad-hoc networks, which provides hop-by-hop routing along minimum-hop paths to each destination. Each node running TBRPF computes a source tree (providing paths to all reachable nodes) based on partial topology information stored in its topology table, using a modification of Dijkstra's algorithm.

2.3.3 Active Adaptation

Here, we discuss related work in active path shortening protocols.

Dynamic Source Routing (DSR) [3, 31] is an on-demand routing protocol which uses aggressive caching and source routing headers to obtain the topology information. A DSR node is able to learn routes by overhearing packets not addressed to it by operating its network interfaces in promiscuous receive mode. This scheme also automatically shortens the active paths as well as our OR2 scheme while sending data packets. The feature can achieve the dynamic multi-hop path shortening, thus it leads the drastic improvement of the packet latency. However, this scheme requires an always-active transceiver mode of the network interfaces and more CPU cycles to process overheard packets, which may be significantly power consuming. This is especially inefficient in environments where battery power is a scarce resource. Also, because DSR does not take the link quality into account, it possibly leads to inefficient and frequent route change and the great degradation of the link quality.

Roy [28] presents the source-tree on-demand adaptive routing protocol (SOAR) based on link-state information. SOAR incurs much less overhead of control routing packets than DSR under various scenarios, ranging from high mobility to low mobility. SOAR also has the mechanism to shorten the active paths, but it achieves that by *periodically* exchanging link-state information in which a wireless router communicates to its neighbors the link states of only those links in its source tree that belong to the path it chooses to advertise for reaching destinations with which it has active flows.

As this partial topology broadcast algorithms exchange control packets of relatively larger size including the minimal source tree, total byte overhead due to control packets has been found to be $2-3$ times more in SOAR compared to the previous ad hoc routing protocols (e.g., even DSR). High control packet overhead is undesirable in low-bandwidth wireless environments. Additionally, periodical exchanging messages could collide with the data streams, thereby may degrading the performance. The mechanism to maintain and synchronize the minimum source tree of its own neighbor nodes with the varying network conditions is fairly complex and computational overhead.

In contrast to the above works, “OR2” does not lead to the weak-connectivity shortened routes or inefficient frequent routes switching since it is based on local link quality, and does not need periodic information advertisements or any overheard packets by making the network interfaces promiscuous receiving mode. OR2 adapts effectively to node mobility using local link quality in wireless ad hoc networks which are scarce bandwidth and battery environment.

Chapter 3

Design of OR2

This chapter describes the detailed design of OR2. First, we explain some scenarios in which OR2 effectively shorten active paths. Second, we introduce the notion of proximity to identify two near nodes by using link quality and discuss deeply the link quality . Finally, we explain OR2 protocol to perform active one-hop shortening in detail and outline the simple multi-hop shortening scheme.

3.1 Path Inefficiency

In a mobile ad hoc network, due to node mobility, we encounter a situation shown in Figure 3.1. In this case, we pay attention to node mobility without link disconnection. For such node mobility, we possibly find the less hop route (i.e., direct hop route shown in Figure 3.1) than the current route in use.

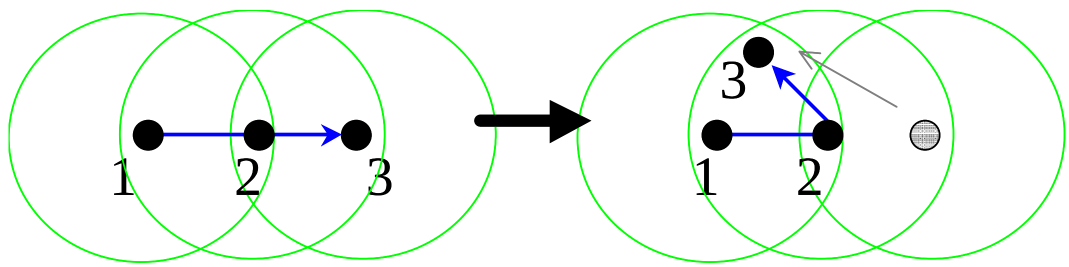


Figure 3.1: Node 1 sends packets to Node 3 through Node 2. At the next step Node 3 moves into the cell of Node 1 without link failures. Although Node 1 can directly send packets to Node 3, Node 1 still sends packets to Node 3 through Node 2.

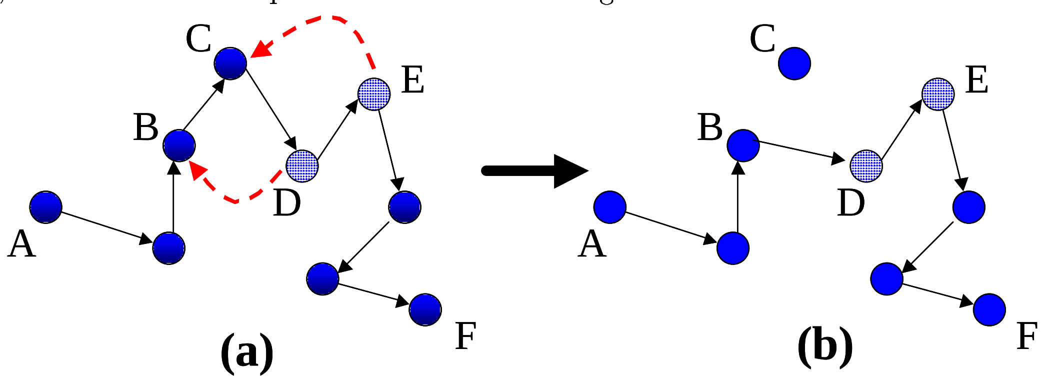


Figure 3.2: Node A sends packets to Node F in a multi-hop network. By using OR2's algorithm, Node D and E can shorten the path currently in use preserving the consistency of the active path.

This scenario in particular is likely to occur frequently in the realistic environment with pedestrian or slow vehicle speed in our daily life. However, the most of the previous routing protocols cannot accommodate the change of network topology without any link failures.

Thus, there exists the path inefficiency in respect to the hop count, network capacity and power consumption while communicating with other nodes in an ad hoc network. We eliminate the inefficiency by using local link information and the concept of *proximity* in the next section.

On the other hand, OR2 also finely accommodates large-scale and dense networks since it is decentralized algorithm using local link quality information. Figure 3.2 shows a more complicated scenario in which OR2 is tuning up the active path adapting to node mobility. In the figure, some less hop routes are available in the active path from source to destination. If each neighbor node simultaneously shortens the active path (in Figure 3.2 $D \rightarrow B$ and $E \rightarrow C$), it leads to the isolated routes and deadlocking. As a result, the active path from source to destination is failed and the sender node must re-initiate a new route discovery. We describe how to overcome this problem later.

3.2 Link Quality

It is desirable for a node on a path to determine whether or not it can shorten the path based on some indicators of the quality of the link between the node and its neighbors on the path. For such an indicator, we use the Signal-to-Noise Ratio (SNR) of the link associated with receiving packets. By definition, SNR represents a channel condition and is expressed as the ratio of signal to noise in electrical power. When the value of SNR becomes higher, the link communication quality will also be relatively higher. However, it should be noted that the SNR could change dynamically with a high frequency due to electro-magnetic effects.

From the point of view of measuring the link quality, OR2 uses a smoothed value of SNR in a time domain. This value, Smoothed SNR (SSNR), can be computed using a weighted moving average technique as follows: $ssnr = (1 - \alpha) * old\_ssnr + \alpha * cur\_snr$, where cur_snr and old_ssnr represent the value of SNR on receipt of a packet and the previously computed

SSNR, respectively. The constant value of α is a filtering factor and is set to *1/8* in this paper. It is because we could adapt to the large fluctuation of SNR and use a shift operation in our experimental implementation. In OR2, the filter calculates SSNR whenever a node receives the frames.

Below, we explain our assuming SNR model used in this work. In simulation experiments, we use the theoretical analysis of SNR, on the other hand, we use our empirical analysis of SNR in real experimental implementation.

3.2.1 Theoretical SNR Analysis

The signal power at any point is the sum of the main signal transmitted by the antenna in addition to components of the signal that reflect off-of the surrounding features (multi-path effect) [27] plus the thermal (receiver) noise. This resulting power is used as the basis of SNR, which determines the probability of successful signal reception for a given frame packet.

To theoretically predict the received signal power of each packet, several radio propagation models are proposed. Mainly, the three propagation models are generally used [27]: the free space model, two-ray ground reflection model and the shadowing model. We present a summary of them shortly. The free space model assumes the ideal propagation condition that there is only one clear line-of-sight path between the transmitter and receiver, hence, it basically represents the communication range as a circle around the transmitter.

$$P_r(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L} \tag{3.1}$$

where P_t is the transmitted signal power. G_t and G_r are the antenna gains of the transmitter and the receiver respectively. $L(L \geq 1)$ is the system loss, and λ is the wavelength. It is common to select $G_t = G_r = 1$ and $L = 1$ in ns2 simulator.

The two-ray ground reflection model considers both the direct path and a ground reflection path. It is shown [27] that this model gives more accurate prediction at a long distance than the free space model. In contrast, it does not give a good result for a short distance due to the oscillation caused by the constructive and destructive combination of the two rays. At the last,

The free space model and the two-ray model predict the received power as a deterministic function of distance. They both represent the communication range as an ideal circle. In reality, the received power at certain distance is a random variable due to multi-path propagation effects, which is also known as fading effects. In fact, the above two models predicts the mean received power at distance d. A more general and widely-used model is called the shadowing model.

The shadowing model consists of two parts. The first one is known as path loss model, which also predicts the mean received power at distance d, denoted by $\overline{P_r(d)}$. It uses a close-in distance d_0 as a reference. $\overline{P_r(d)}$ is computed relative to $P_r(d_0)$ as follows.

$$\frac{P_r(d_0)}{\overline{P_r(d)}} = \left(\frac{d}{d_0}\right)^{\beta} \tag{3.2}$$

β is called the path loss exponent, and is usually empirically determined by field measurement. From Equation. (3.1) we know that $\beta = 2$ for free space propagation. Table 3.1 gives some typical values of β. Larger values correspond to more obstructions and hence faster decrease in average received power as distance becomes larger. $P_r(d_0)$ can be computed from Equation. (3.1).

The path loss is usually measured in dB. So from Equation. (3.2) we have

Table 3.1: Some typical values of path loss exponent β

Environment		β
Outdoor	Free space	2
	Shadowed urban area	2.7 to 5
In building	Line-of-sight	1.6 to 1.8
	Obstructed	4 to 6

Table 3.2: Some typical values of shadowing deviation σ_{dB}

Environment	σ_{dB} (dB)
Outdoor	4 to 12
Office, hard partition	7
Office, soft partition	9.6
Factory, line-of-sight	3 to 6
Factory, obstructed	6.8

$$\left[\frac{\overline{P_r(d)}}{P_r(d_0)}\right]_{dB} = -10\beta \log\left(\frac{d}{d_0}\right) \tag{3.3}$$

The second part of the shadowing model reflects the variation of the received power at certain distance. It is a log-normal random variable, that is, it is of Gaussian distribution if measured in dB. The overall shadowing model is represented by

$$\left[\frac{P_r(d)}{P_r(d_0)}\right]_{dB} = -10\beta \log\left(\frac{d}{d_0}\right) + X_{dB} \tag{3.4}$$

where X_{dB} is a Gaussian random variable with zero mean and standard deviation σ_{dB}. σ_{dB} is called the shadowing deviation, and is also obtained by measurement. Table 3.2 shows some typical values of σ_{dB}. Equation. (3.4) is also known as a log-normal shadowing model.

The shadowing model extends the ideal circle model to a richer statistic model: nodes can only probabilistically communicate when near the edge of the communication range. In our simulation, we use the shadowing propagation model.

3.2.2 Empirical SNR Analysis

Let us consider the situation which two nodes approach each other. If the distance between two nodes is associated with the SSNR of the link between the two nodes, one of the pair can determine whether the other one is near the own position. In order to investigate this assumption (i.e., the relationship between distance and SSNR), we transmitted a constant rate UDP stream at 1 Mbps between two wireless nodes and changed the location of the receiver. Fig.3.3 and 3.4 show the results. We found that the values of SSNR larger than 10 dB were stable enough for the threshold value (S_{max}). In these experiments we used the WaveLAN NICs.

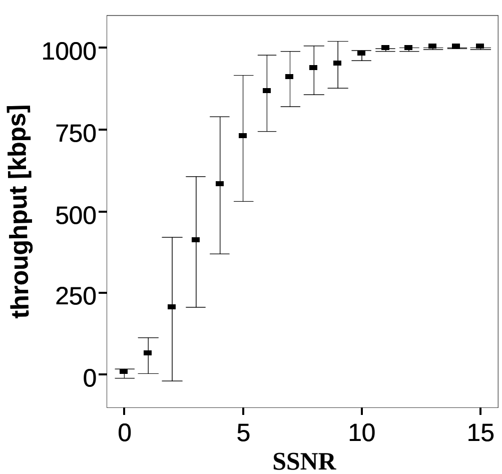


Figure 3.3: SSNR vs. throughput

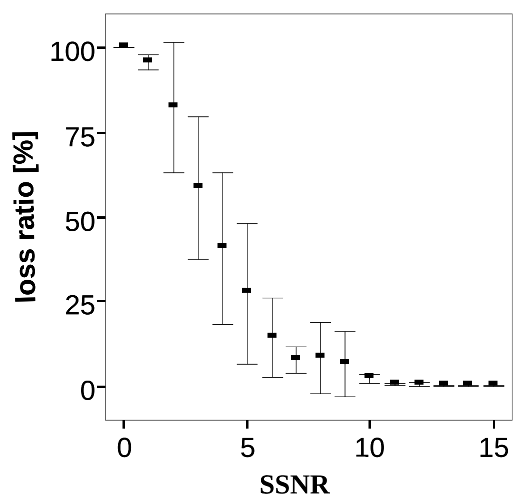


Figure 3.4: SSNR vs. loss ratio

Next, we traced the relationship between SSNR and the communication distance in a wireless outdoor environment. This experiment was performed in our flat rectangular ground (300m x 300m) with no obstacles or walls. Fig.3.5 shows the transition of SSNR as a function of the distance. At small distance (< 10m), we have obtained the significantly increasing values of SSNR. As we can see from the figure, the values of SSNR increase considerably when the distance is smaller than 10m. In addition, even when the distance is farther than 10m, the values of SSNR monotonically decrease with respect to the distance. In this case, we used the IEEE 802.11b NICs.

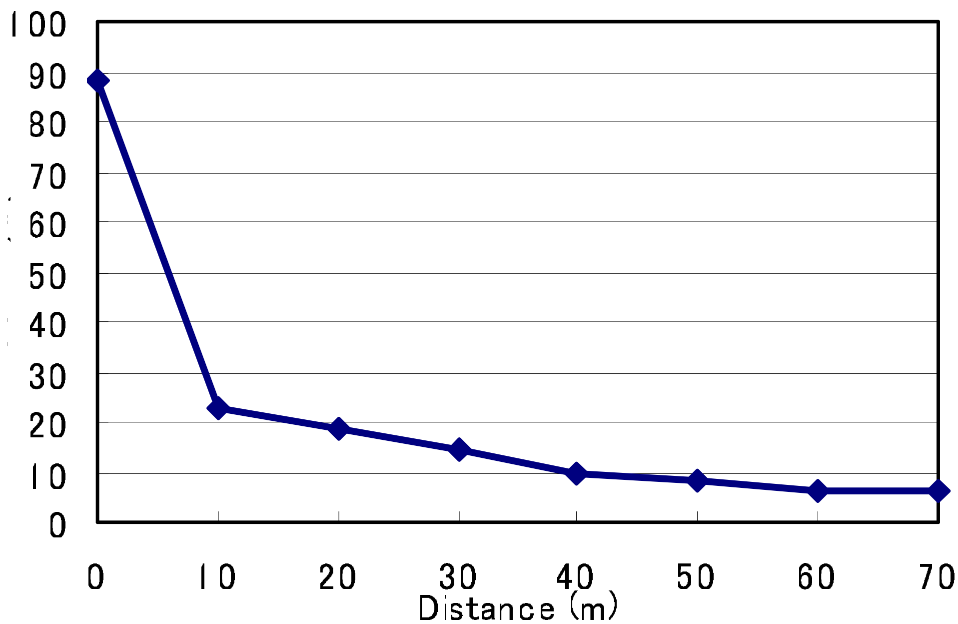


Figure 3.5: SSNR vs. Distance

By deciding some optimal threshold value of the receiving SSNR, we can possibly estimate the nearness of the other node. Thus we assume that some highly receiving SSNR value indicate the nearness of the two nodes or the good condition of the link. In OR2, we assumes that transmission power can not be varied and all nodes in an ad hoc network have the same network interfaces.

3.3 Proximity

To argue the "nearness" of two nodes more formally, we introduce the notion of proximity based on the observation of the relationship between the distance and the SSNR between two nodes. Let us define the following symbols.

- $S_{(AB)}$: The SSNR value observed at Node B for received data packets from Node A.
- S_{max}: A threshold value of SSNR.
- $P_{(A)}$: The proximity of Node A.
- $R_{uf}(A)$: The upstream adjacent node of Node A for flow f.
- $R_{df}(A)$: The downstream adjacent node of Node A for flow f.

We hypothesize that $S_{(AB)} = S_{(BA)}$. This is not impractical since homogeneous nodes are assumed in many mobile ad hoc networks. We will discuss a case in which this assumption does not hold in the future. If $S_{(AB)} \geq S_{max}$, Node B is said to be in the proximity of Node A, or $B \in P_{(A)}$. Based on the above hypothesis, if $B \in P_{(A)}$, then $A \in P_{(B)}$.

Let us assume that a flow traverses Node A, B, and C in this order. This can be written as $A = R_{uf}(B) = R_{uf}(R_{uf}(C)) = R^2_{uf}(C)$. If $C \in P_{(B)}$, there is a possibility that the path of the flow can be changed: $A = R_{uf}(C)$. As shown in Figure 3.6, each node is associated with its own proximity. When Node C moves to the proximity of Node B, Node A can directly send data packets to C. This motivates us to design our scheme described in the next section. In practice, we need a hysteresis mechanism around the threshold value to avoid oscillation.

3.4 Design of OR2: One-Hop Shortening

We set two design goals to OR2: reducing the hop count of a path, and minimizing the number of additional control packets. The first goal is obvious in the context of the problem

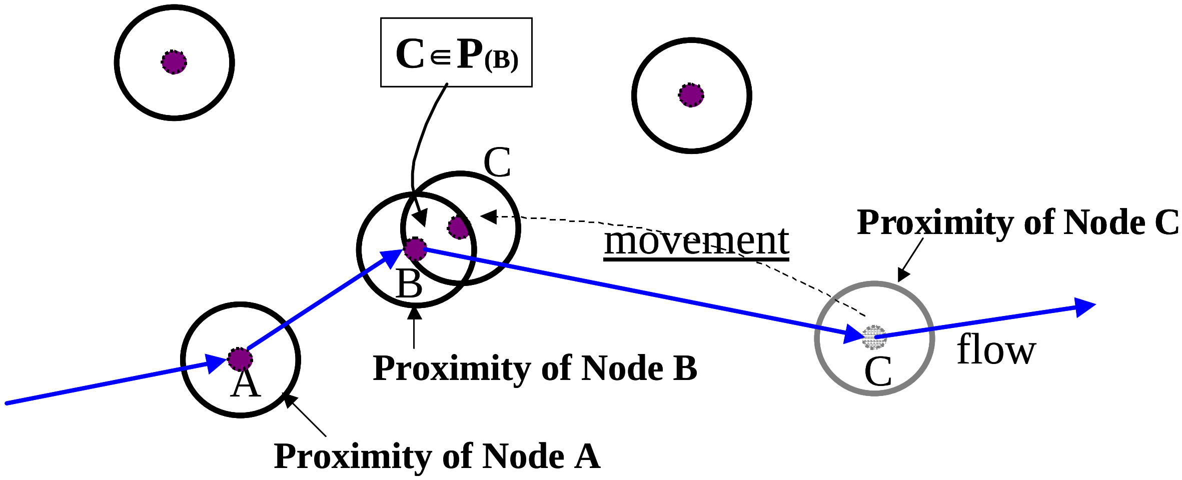
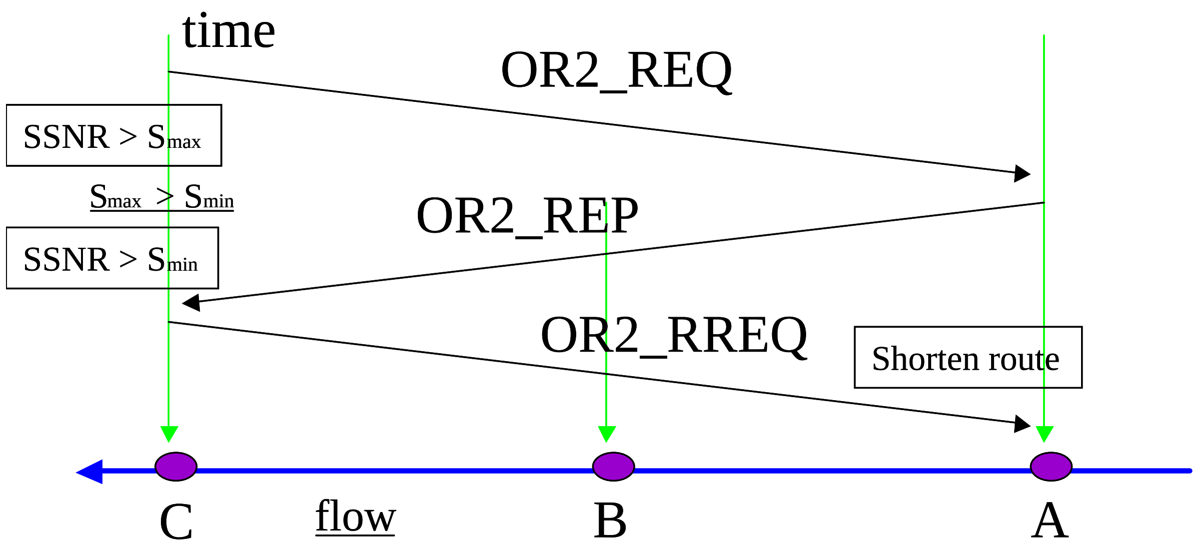


Figure 3.6: Proximity of node

aforementioned. In addition to the first goal, we aim at a scheme that does not produce many control packets. In particular, we do not allow transmission of control packets when active flows do not exist. This is an important consideration for an ad hoc network since nodes in the network need to reduce their power consumption.

We design a scheme in which control packets are transmitted only when a node determines that a path should be changed based on the proximity. We call this scheme OR2. In designing OR2, we made an assumption that each node in ad hoc networks has the original routing information concerning upstream two-hop-away nodes. Since a node attempts to transmit the control packet to the upstream two-hop-away node, the node needs to retain the route information of its upstream two-hop-away nodes of the active flows as well as its neighbors.

Let us now explain the fundamental messages passed among three nodes. OR2 uses three kinds of messages: OR2_REQ, OR2_REP, and OR2_RREQ; they are shown in Figure 3.7. OR2_REQ and OR2_RREQ are newly generated control packets, while OR2_REP can be piggybacked on a data packet. Let us assume that $A = R_{uf}(B)$ and $B = R_{uf}(C)$ for flow f as shown in Figure 3.7.



OR2_REQ	Request to $R^2_{uf}(k)$ as a packet to probe direct route
OR2_REP	Reply to $R^2_{df}(k)$ via $R_{df}(k)$ for OR2_REQ (piggybacked)
OR2_RREQ	Redirect Request to $R^2_{uf}(k)$ via $R_{uf}(k)$ to shorten route

Figure 3.7: Three OR2 control packets

When Node C determines that it has moved into the proximity of Node B, it sends OR2_REQ to Node A. The intent is to observe whether or not a packet can be directly exchanged between Node A and C. Upon receipt of OR2_REQ, Node A sends OR2_REP to Node C. Unlike OR2_REQ, OR2_REP is not sent as a single control packet. Rather, Node A inserts it as a DSR option header into the data packet of flow f. Therefore OR2_REP reaches Node A via Node B. By receiving OR2_REP, Node C knows that Node A can send packets directly to Node C; Node C sends OR2_RREQ to Node A to initiate a change of route. The extra packets of OR2_REQ and OR2_RREQ may temporarily interfere with data packets. However, the overhead incurred with the packets is still negligibly small compared with an alternative scheme using HELLO messages.

There is concern about a race condition; simultaneous attempts by each adjacent nodes to shorten the same path may occur as shown in Figure 3.2. We solve this problem in a way similar to TCP's three-way handshake but in a more delicate way to handle mutual exclusion. Considering the above problem, let us describe the protocol of OR2. The state transition at node K of flow f is shown in Figure 3.8 and the handling of the race condition is shown in the following Figure 3.9.

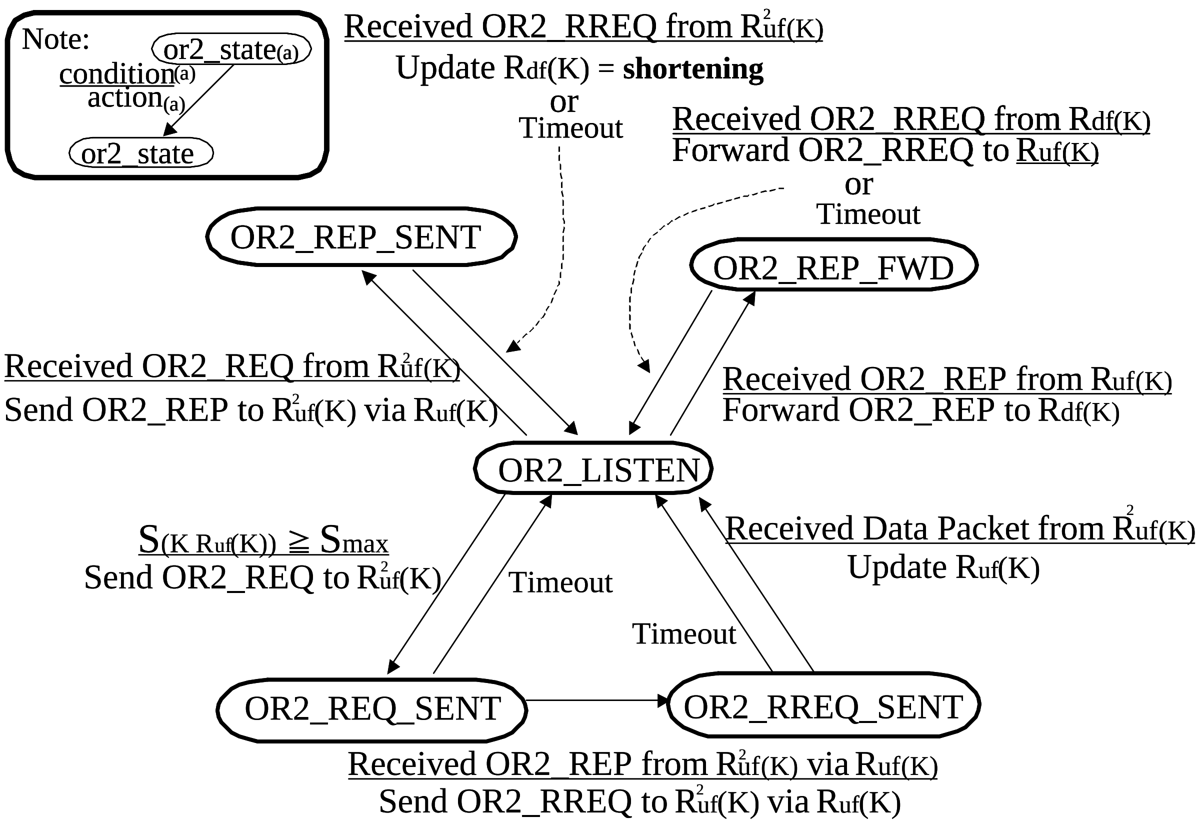


Figure 3.8: OR2 state transition diagram at node K

Let us assume that $A = R_{uf}(B)$, $B = R_{uf}(C)$, and $C = R_{uf}(D)$ for flow f. When $S_{(BC)} \geq S_{max}$, Node C sends OR2_REQ to Node $R^2_{uf}(C)$ (i.e., Node A) to locate the direct hop route. As long as $S_{(BC)} \geq S_{max}$, Node C continues to send OR2_REQ every time OR2's timer expires until Node C receives OR2_REP. Upon successful receipt of OR2_REP, Node C sends

```
node K :

Received OR2_REQ, OR2_REP or OR2_RREQ

if (timeout)
        goto OR2_LISTEN
else if (or2_state(a) & condition(a))
        exec action(a)
else drop the message
```

Figure 3.9: OR2 solution of race condition

OR2_RREQ to Node A to ask for the redirection of the path of flow f. Upon success in the above process, Node A can directly send data packets to Node C.

If we consider a case in which Node D is also attempting to make a short cut between Nodes B and D, Node D sends OR2_REQ to Node B. When Node B receives OR2_REQ, the state of flow f at Node B moves to OR2_REP_SENT. If there is an OR2_REP message from Node A to C, it traverses Node B. When this OR2_REP message reaches Node B and the state is OR2_REP_SENT, the message is discarded since the short cut between Nodes B and C is on-going. Thus the short cut from Node B to C is prioritized. In contrast, if an OR2_REP message from A to C reaches Node B ahead of an OR2_REQ message from B to D, the state of B changes to OR2_REP_FWD and suppresses the short cut from Node B to C.

3.5 Multi-Hop Shortening

In the previous section, we described the mechanism of one-hop shortening scheme. This one-hop shortening algorithm is simply workable and can also continue to shorten an active route as possible, not only one-hop route. However, multi-hop shortening scheme is ideally the best way to optimize the active path than repeating one-hop shortening in such a case as Figure 3.10.

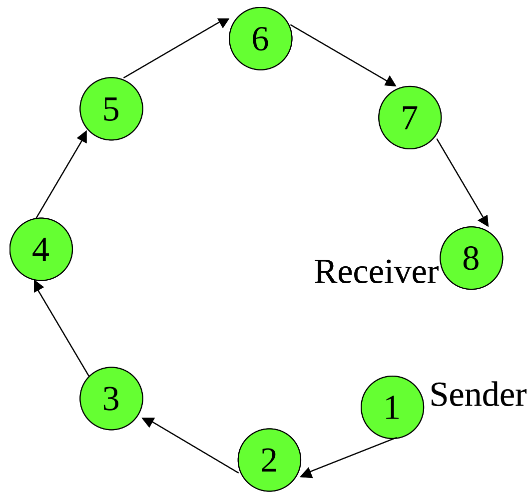


Figure 3.10: Typical Multi-Hop Shortening Scenario

Currently, we think that dynamic multi-hop shortening schemes may be realized only by promiscuous listening mode. Thus, to perform multi-hop shortening, we cooperate OR2 with DSR feature which overhears packets not addressed to it by operating its network interfaces in promiscuous receive mode. This collaboration enables the routing protocols enhanced by OR2 to conduct both the active (by OR2) and reactive (by the promiscuous mode) multi-hop path shortcut. Presently, though this multi-hop shortening mechanism is only provided to DSR protocol which has the feature operating by the promiscuous mode, other protocols can also perform the multi-hop shortening if they can do the promiscuous listening.

However, this feature requires an active receiver in the nodes, which may be rather power consuming. In networks were nodes have limited power the aim is to shut down the transceiver as often as possible to conserve power. Thus, in this thesis we mainly pay attention to the one-hop shortening scheme and show the remarkable benefit of our one-hop shortening scheme, though we will perform the preliminary experiments and evaluation of the multi-hop shortening scheme.

3.6 Protocol Parameters

In OR2, the configurable protocol parameters are S_{max}, the threshold value of SSNR, which determines whether OR2 performs the path shortening, and the time-out value of the state for shortening active paths which affects the spacing between sending OR2_REQs. As to S_{max}, we typically set the value assuming wireless devices in the ISM bands (such as Lucent WaveLANs) and to be conservative for the protocol performance. Also, determining this value is dependent on what simulator we use.

On the other hand, we set the time-out value to three seconds considering some scenarios in relatively large mobile ad hoc networks. This three seconds was determined by using a number of heuristics from the real experiments and our simulation.

Chapter 4

Implementation of OR2

This chapter describes the implementation of a proximity-based routing scheme (OR2). We explain the management implementation of received signal power to envision a proximity area. Next, we describe our implementation of OR2 as an extension to DSR on FreeBSD OS. Last, we present the description of OR2 simulator implementation and describe the implementation of our proposed two node mobility generation models.

4.1 Management of Link Quality State

We describe the implementation of local proximity checking mechanisms retrieving the SNR value form wireless network interfaces.

We extended "wi" driver of IEEE 802.11b [10] wireless LAN cards to export the pairs of the source IP address and the SSNR, which is supplied by the register on the interface cards when a frame packet arrives, to the OR2 protocol layer (in the */sys/i386/isa/if_wi.c* file).

Since the SNR value is retrieved every time a frame arrives, the SNR value accurately reflects up-to-date link quality. Additionally, to control the oscillation of SNR values, we computed Smoothed SNR (SSNR) using a weighted moving average technique as described in the previous chapter. We compare the SSNR with S_{max} n times to initiate path shortening. This n parameter is currently set to 10. Interestingly, the n parameter has the relatively correlation to the frame data receiving rate. Figure 4.1 shows the or2_sigcache structure which associates the IP address of $R_{uf}(k)$ with the SSNR value.

```
struct or2_sigcache {
    char    macsrc[6]; /* unique MAC address */
    int     ipsrc;     /* IP source address */
    int     ssnr;      /* Smoothed Signal-to-Noise Ratio */
}
```

Figure 4.1: or2_sigcache structure

Note that we need to modify the link layer frame routines of DSR to make use of the standard link mechanisms, because Monarch DSR implementation uses the link broadcast scheme to perform promiscuously listening.

4.2 One-hop shortening

OR2 scheme was built on off-the-shelf wireless LAN technology. We have implemented OR2 as an extension to DSR developed by the Monarch project [31]. Since the Monarch DSR was implemented on FreeBSD 3.3-RELEASE OS platform with WaveLAN [21] driver particularly, we have performed porting DSR to current FreeBSD 4.4-RELEASE OS. We implemented DSR extended by OR2 on FreeBSD OS running on laptops as one of the kernel protocol stacks.

More specifically, as control packets to initiate the shortening of active paths, we added OR2_REQ, OR2_REP, and OR2_RREQ header options as one of DSR header option types: */sys/dsr/ip6_opts.[h,c]*. OR2_REP is always piggybacked on a data packet. Additionally, we added some routines which send and receive OR2_REQ, OR2_REP, and OR2_RREQ in */sys/dsr/dsr_output.c* and */sys/dsr/dsr_input.c*. Specifically, the DSR option header is inserted following DSR header after IP header, followed by headers such as a transport layer header. OR2 needs to hold the two state information of exchanging packets to shorten active paths: the skipped node's IP address and the shortening initiate node's IP address (the first OR2_REQ sender). Figure 4.2 shows the or2_reqTbl structure which keeps the two state.

```
struct or2_seqTbl {
    int32_t     skipped;  /* skipped node ip address   */
    int32_t     xmit_at;  /* OR2_REQ sender ip address */
}
```

Figure 4.2: or2_seqTbl structure

Also, OR2_REQ sender obtains IP address of the two-hop upstream neighbor from the source routing header of receiving data packets. Since DSR is the source routing protocol, we should report the changes of the active route to send GRATUITOUS_REPLY [3] to the source node when OR2 completes shortening of active paths. Furthermore, we implemented the timer routine to reset the above states to compensate such cases as OR2 shortening failures or incompletion due to dropping one of the three OR2 control packets, etc.

The pseudo-code summarizing the salient feature of our algorithm is shown in Figure 4.3. This figure mainly depicts the action of the initiator which starts OR2's procedures.

OR2 Notations:

```
is_reply()    : Is this OR2_REP?
probe         : Flag indicating probing now.
Smax          : SSNR max threshold of Proximity.
Smin          : SSNR min threshold for hysteresis.
conform()     : Send OR2_RREQ for shorting route.
probe_route() : Send OR2_REQ.
```

Shortening path:

```
Each packet arrives
reply_flag = is_reply();
if (!reply_flag && probe)
   return;
SSNR = calc_SSNR();
if (reply_flag && SSNR > Smin)
   conform();
else if (SSNR > Smax)
   probe_route();
   probe++;
```

Figure 4.3: Pseudo-code of OR2

4.3 Multi-hop shortening

As discussed in the previous chapter, to perform multi-hop shortening, we only cooperate OR2 with DSR feature which overhears packets not addressed to it by operating its network interfaces in promiscuous receive mode.

4.4 Simulator Implementation

This section describes our implementation of OR2 adding to DSR and AODV, and the realistic node mobility models we used in our simulations. We ran OR2 implementation in the *ns-2* network simulator environment.

4.4.1 DSR Implementation Decisions

In contrast to the implementation on FreeBSD OS, we implemented the two protocols in simulation: the fast shortening mode and robust shortening mode. The latter one is the same as the scheme in real environments. In the former fast shortening mode, OR2 realizes faster shortening at the cost of the completion probability of the shortening operation.

This scheme operates as follows. When a node enters the proximity area of his upstream neighbor node, it sends OR2_REQ to the two-hop upstream neighbor node as the same way in real implementation. However, the node which received the OR2_REQ promptly switch the active route to directly forward data packets to the node which sent the OR2_REQ. Since this scheme operates based on the optimistic policy, it cannot assure the consistency of the shortening for the race condition problems as described in the previous chapter. Although the performance of the fast shortening scheme were better than that of the robust mode in preliminary some simulation scenarios, in the end, we chose the robust shortening scheme policy as a conservative solution.

4.4.2 AODV Implementation Decisions

To illustrate that OR2 is not specific to DSR protocol, we incorporated the dynamic path shortening mechanism into AODV. The modifications we had to make for AODV were somewhat different than those incorporated for DSR. Specifically, data packets do not carry the full source route in their header since AODV is the distance vector routing algorithm. Thus, the two-hop upstream neighbor is not available from their header. However, we can easily cope with this problem to be forwarded OR2_REQ packets by the one-hop upstream neighbor node. In other words, OR2_REQ packets travel two-hop journey via the upstream neighbor node as the intermediate node, instead of on-hop direct communication. In the intermediate node, to distinguish own particular route, we needed the matching scheme based on the final destination and the next hop node.

Also, AODV in *ns-2* was implemented as a user-land application daemon, the final destination node cannot initiate the shorten route, unlike the cases in DSR. For comparison with DSR based OR2, we chose to implement OR2 on AODV-LL (Link Layer) [4] using only link layer feedback from 802.11 as in DSR, completely eliminating the standard AODV Hello mechanism.

4.4.3 Realistic Node Mobility Models

To investigate how OR2 scheme perform in the *realistic* node mobility pattern, we proposed the two node mobility models: the "random oriented model" and "random escape model". These models are based on the the *random way-point model* [14] used in most of the previous simulation research.

In the *random way-point model*, each node begins the simulation by remaining stationary for *pause time* seconds. It then selects a random destination in the specified field space and moves to the destination at a speed distributed uniformly between *0* and some maximum speed. On reaching the destination, the node pauses again for *pause time* seconds, selects another destination, and proceeds there as previously described, repeating this behavior for the duration of the simulation.

In contrast, our two node mobility model generate more realistic movement patterns. The random oriented node mobility is assuming people pursuing something (e.g., *peace*, *money*, *hope*, etc) or attracted something(e.g., *gravity*, *power*). On the other hand, the random escape model is literally assuming people are escaping from something (e.g., *disaster*, *ghost*, etc).

More specifically, we explain these models. In our proposed models, mobile nodes are classified into two types: CORE NODES and ORIENTED or ESCAPE NODES. CORE NODES move around the simulation field based on the random way-point models accurately. The other side, ORIENTED NODE select one destination from the destinations of CORE NODES instead of a random destination and pursue one CORE NODE at a speed distributed uniformly between *0* and some maximum speed (*not all people require money*). If one ORIENTED NODE reaches the selecting destination, then it selects another destination among that of CORE NODES. Figure 4.4 shows the snapshot of the random oriented model.

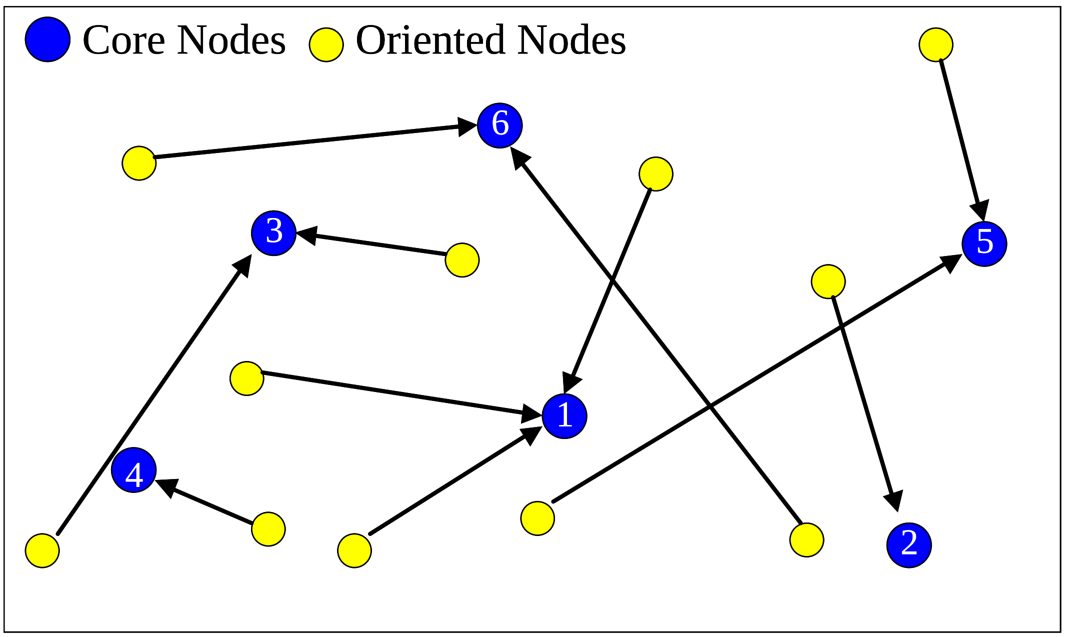


Figure 4.4: Snapshot of Random Oriented Model

In the random escape model, ESCAPE NODES desperately leave from one particular CORE NODE. ESCAPE NODES select the exact opposite side destination to the particular destination of one Core Node. By the random escape model, we consider human mobility in the situations as disaster to where ad hoc networks expect to apply. Figure 4.5 presents the snapshot of the random escape model. Note that, when the node mobility file is generate, we specify the ratio of Oriented Nodes or Escape Nodes to Core Nodes as one argument. If the stated ratio is 0.0, the generated node mobility pattern is accurately based on the random way-point model.

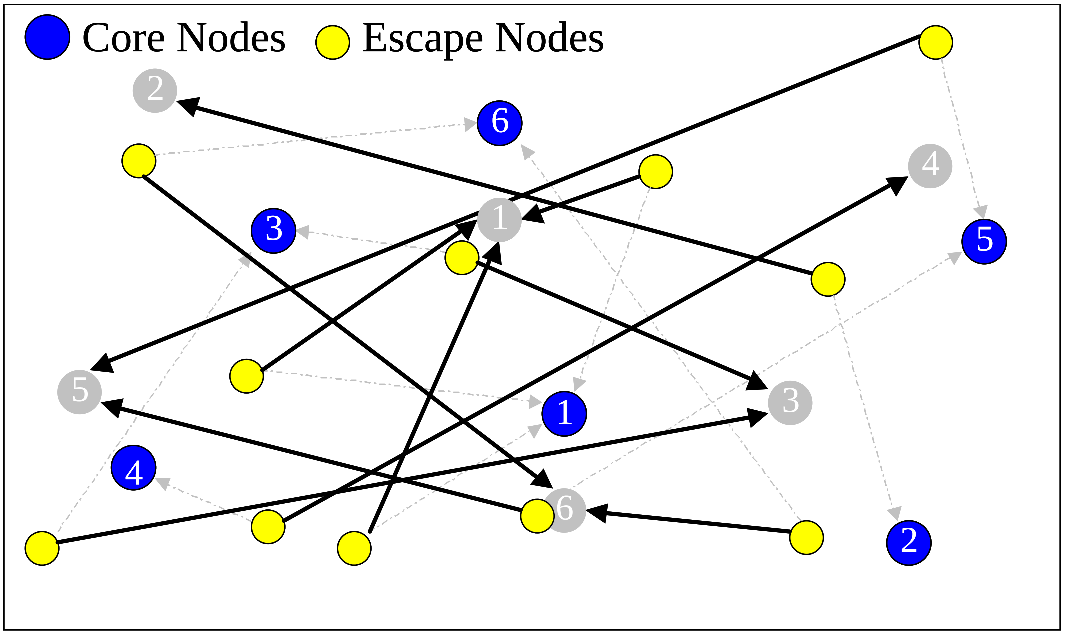


Figure 4.5: Snapshot of Random Escape Model

Chapter 5

Performance Evaluation

In this chapter, we show the experimental results of OR2 and detailed simulation results. In the first evaluation, we build a small wireless ad hoc network and perform several preliminary experiments. Next, we simulate OR2 on several large mobile topologies to qualify the scaling behavior of OR2 in Network Simulator (ns2) [32]. In addition, to study how OR2 scheme perform in *realistic* node mobility patterns, we measure the effectiveness of OR2 using our two practical mobility models.

5.1 Experiments in Real Environments

In our experiments, mobile nodes are Pentium-based laptop computers running FreeBSD 4.4 and equipped with a MELCO IEEE 802.11b wireless network card. We installed DSR and OR2 in these nodes and conducted two preliminary experiments: measurement of the latency in re-routing paths, and quantification of improving TCP throughput by reducing the number of hops.

5.1.1 Latency

To observe the overhead associated with path shortening, we conducted five trials of path shortening among three nodes, Nodes A, B, and C. Node A sends UDP packets continuously to Node C via Node B. For comparison, we measured the round-trip time (RTT) from C to A. The result of measuring the RTT is shown as "Ping" in Figure 5.1.

To create the situation of path shortening, we moved Node C close to Node B. We measured the duration from the time at which Node C sent the first OR2_REQ message to the time at which Node C received data packets directly from Node A. As observed in Figure 5.1, the overhead incurred with the exchange of OR2's messages is sufficiently small; it is less than 5 ms.

5.1.2 TCP Throughput vs. Number of Hops

We also examined the relationship between TCP throughput and the number of hops. We used netperf [19] to send TCP flows. Figure 5.2 shows the obtained results. As seen in the figure, the TCP throughput dramatically decreases as the number of hops increases from 1 to 2. It is also observed that TCP throughput decreases monotonically with the number of hops. Therefore reducing the number of hops performed by OR2 will leads to significant improvement in TCP throughput.

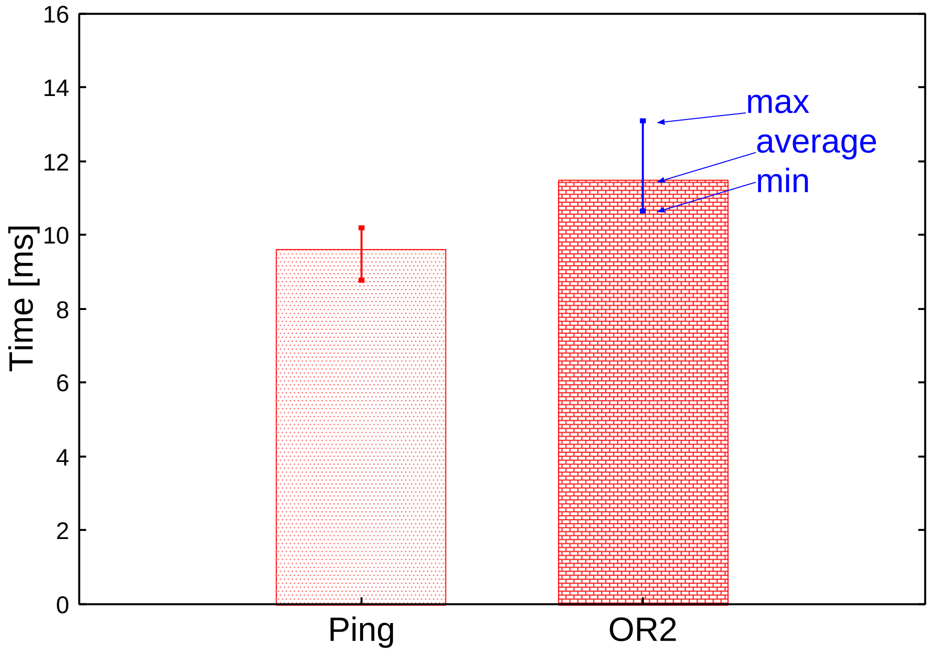


Figure 5.1: Network delay (ping) and latency time of OR2 to shorten an active path over two-hop route

5.2 Detailed Simulation

In this section, we present the results of our simulation comparing the performance of DSR, AODV, DSR and AODV extended by OR2. We verify the effectiveness of OR2.

5.2.1 Simulation Model

We use a detailed simulation model based on *ns-2* in our evaluation. On *ns-2*, the Monarch research group in CMU developed support for simulating multi-hop wireless networks complete with physical, data link and MAC layer models [4]. The distributed coordination function (DCF) of the IEEE standard 802.11 for wireless LANs is used as the MAC layer. The 802.11 DCF uses Request-to-Send (RTS) and Clear-to-Send (CTS) control packets [2] for "unicast" data transmission to a neighboring node. The radio model uses characteris-

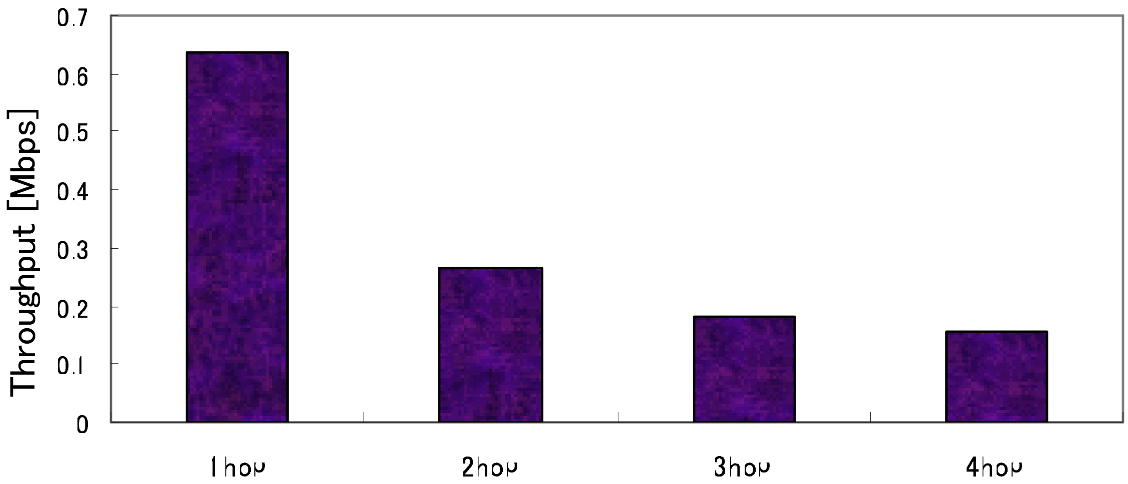


Figure 5.2: TCP throughput vs. number of hops

tics similar to a commercial radio interface, Lucent's WaveLAN [21, 33]. WaveLAN is a shared-media radio with a nominal bit-rate of 2 Mb/sec and a nominal radio range of 250 meters.

The routing protocol model handles all data packets transmitted or forwarded, and responds by invoking routing activities as appropriate. The ROUTE REQUEST (RREQ) packets are treated as broadcast packets in the MAC. ROUTE REPLY (RREP), ROUTE ERROR (RERR) and data packets are all unicast packets with a specified neighbor as the MAC destination. DSR and AODV protocols detect link breakage using feedback from the MAC layer. A signal is sent to the routing layer when the MAC layer fails to deliver a unicast packet to the next hop. In this evaluation, no additional network layer mechanism such as *HELLO Messages* [25] is used.

Table 5.1 and 5.2 provide all the simulation parameters of both protocol extended by OR2. These parameters are remained default parameters of *ns-2* current distribution except OR2 parameters.

Table 5.1: DSR Simulation Parameters

Time between retransmitted Route Requests (exponentially backed off)	500 ms
Size of source route header carrying n addresses	$4n + 4$ bytes
Timeout for non-propagating search	30 ms
Time to hold packets awaiting routes	30 s
Max rate for sending gratuitous Replys for a route	1/s
Max rate for sending OR2 Request for a route	3/s

Table 5.2: AODV-LL Simulation Parameters

Time for which a route is considered active	50 sec
Lifetime on a Route Reply send by destination node	1 sec
Number of times a Route Request is retried	3
Time before a Route Request is retried	10 s
Time for which the broadcast id for a forwarded Route Request is kept	6 sec
Time for which reverse route information for a Route Reply is kept	10 sec
Time before broken link is deleted from routing table	3 sec
MAC layer link breakage detection (Hello Packets OFF)	yes
Max rate for sending OR2 Request for a route	3/s

Traffic and mobility models

Traffic and mobility models use similar to previous published results using *ns-2* ([4], [13], [6]) for appropriate performance comparisons. Traffic sources are CBR (constant bit rate). The source and destination pairs are spread randomly over the network. Only 512 byte date packets are used. The number of source-destination pairs and the packet sending rate in each pair is varied to change the offered load in the network.

The mobility model uses the *random way-point model*, our proposed random oriented model and random escape model in a rectangular area. One field configurations are used - (i) $1500m \times 300m$ field with 50 nodes. Thus, each node starts its travel from a random location with a randomly chosen speed (uniformly distributed between $0-20$ m/sec except in the random escape model). We vary the pause time, which affects the relative speeds of the mobile nodes; in this thesis, we used the following pause times (0, 30, 60, 120, 300, 500 [sec]). Simulation are run for 500 simulated seconds for 50 nodes. Each data point represents an average of five runs with identical traffic models, but different randomly generated mobility scenarios. For fairness, identical mobility and traffic scenarios are used across protocols.

In all the below experiments, we assumed that the useful range of proximity should be restricted below $S_{max} = 0.000008$ obtained from our preliminary analysis and experiments of SNR.

5.2.2 Light Traffic Loads

First, we perform experiments using the light traffic loads to study the behavior of OR2 added DSR and AODV. For the 50 node experiments we used 10 traffic sources and a packet rate 4 packet/sec. We found that OR2 improves the end-to-end delay as expected and reduces the routing control packet overhead effectively (see Figure 5.3, 5.4 and 5.5). However, in the packet delivery ration (PDR), OR2 loses about 5 - 10% packets. While we are currently working the accurate reason of lost packets, we think that the reason is by the failures of the path shortening. In Figure 5.5, note that the packet delivery fractions for DSR are more than 100 %, we think that it may be caused by the retransmissions of data packets performed by DSR protocol.

5.2.3 Heavy Traffic Loads

Here, to stress the traffic loads to OR2, we used 30 traffic sources. The other configuration parameters are the same as the above the light traffic load experiments. In Figure 5.6, 5.7 and 5.8, we can see that DSR with OR2 achieves the significantly reduction of the packet delay and routing overhead. Additionally, in contrast to the first simulation experiment, DSR with OR2 has the high performance of PDR. However, AODV with OR2 does not show the improved performance as salient as DSR with OR2. We think that one of the reasons is the potential feature of AODV protocol; AODV node holds many state information and uses the timer-based routine frequently. Hence, under high node mobility, AODV may not operate normally. In fact, such the simulation results were pointed out in the previous research work [4, 6].

5.2.4 Realistic Mobility Model

To study the performance of OR2 under the high node mobility, we proposed two novel realistic node mobility. These node mobility generate some network congestion points and network partitioning areas reasonably.

Random Oriented Mobility Model

This model typically makes several network and node congestion points. Thus, we can assume the effectiveness of the active shortening in such a area. In Figure 5.9, 5.10 and 5.11, we can see that the improved delay reduction is significant. To generate heavy mobility loads, we have set the ratio of oriented nodes to core nodes to *0.8* (i.e., in 50 mobile nodes case, the number of oriented nodes is *40*) .

Random Escape Mobility Model

In turn, this model makes some network partition areas intentionally. Thus, mobile ad hoc nodes suffer from frequently link failure and relatively speedy node mobility. As Figure 5.12, 5.13 and 5.14 show, AODV protocol degrade its performance in the case. However, OR2 improved the performance of such a protocol as AODV. In this case, we used the ratio of escape nodes to core nodes to *0.8*.

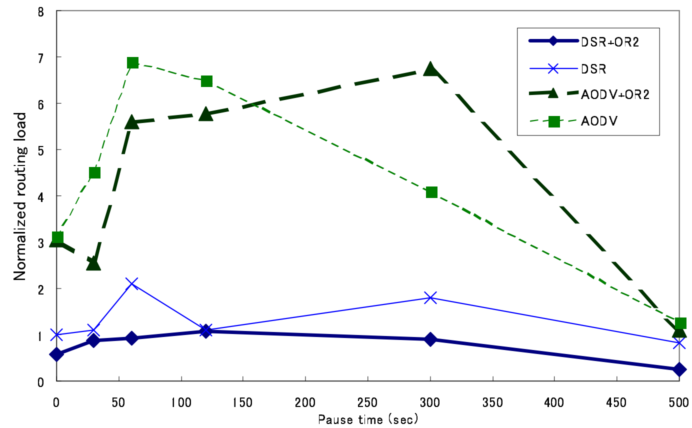


Figure 5.3: Normalized routing load for 10 sources model

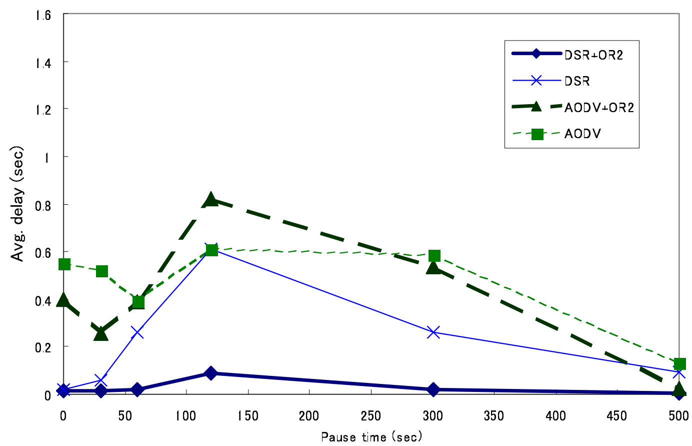


Figure 5.4: Average data packet delay for 10 sources model

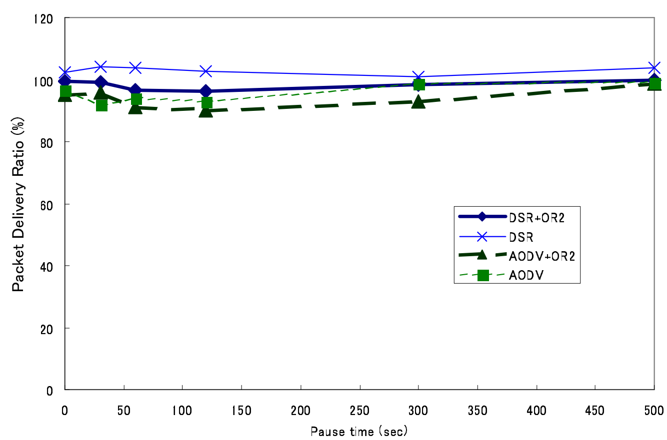


Figure 5.5: Packet delivery fraction for 10 sources model

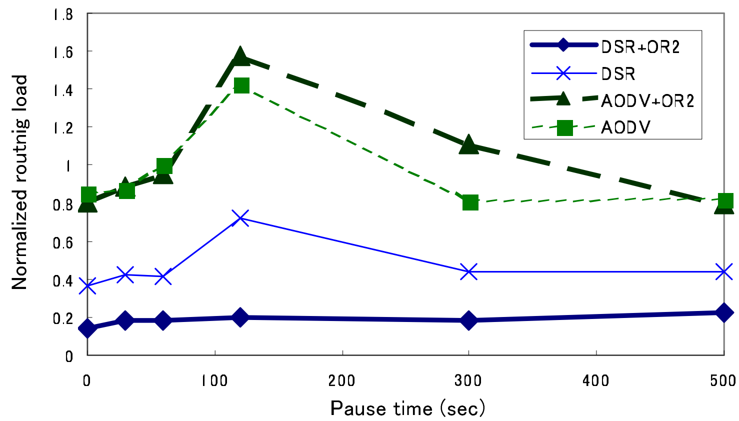


Figure 5.6: Normalized routing load for 30 sources model

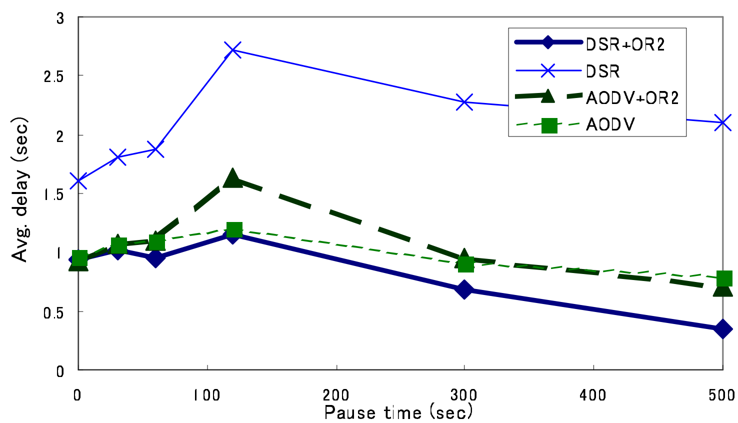


Figure 5.7: Average data packet delay for 30 sources model

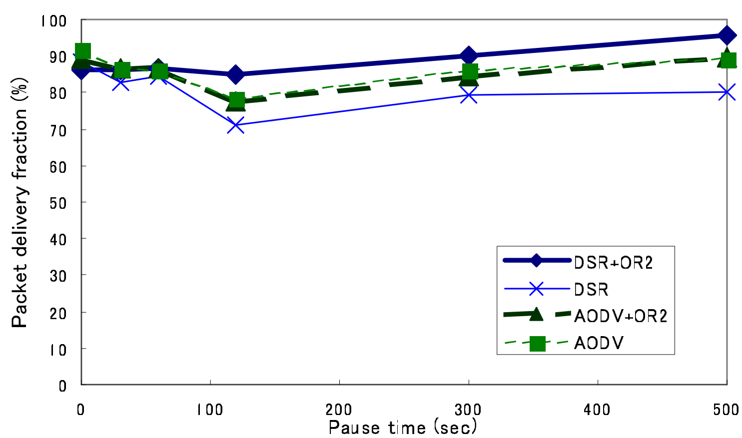


Figure 5.8: Packet delivery fraction for 30 sources model

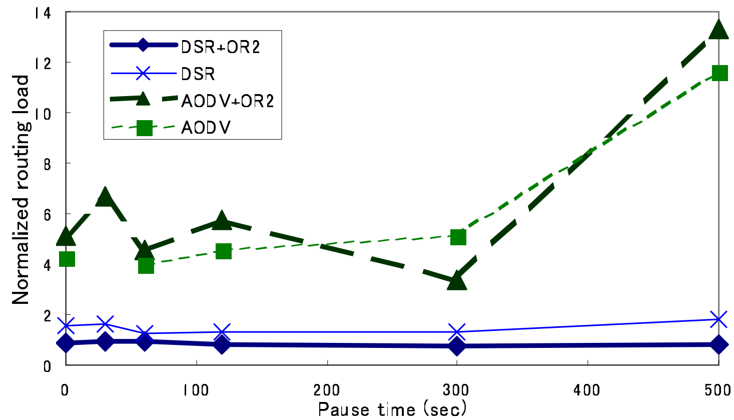


Figure 5.9: Normalized routing load in our oriented mobility model

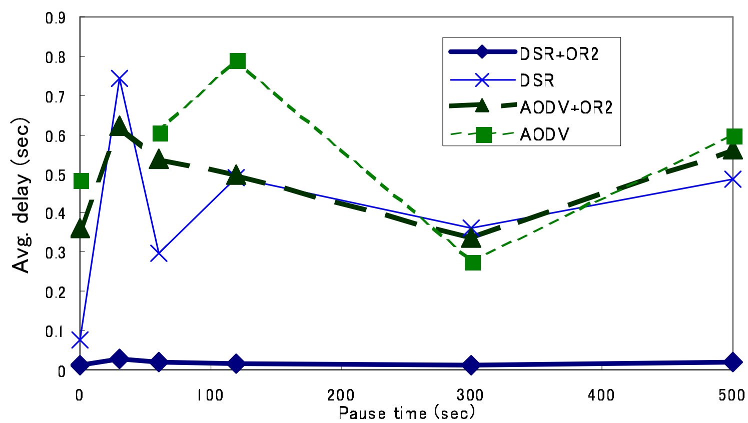


Figure 5.10: Average data packet delay in our oriented mobility model

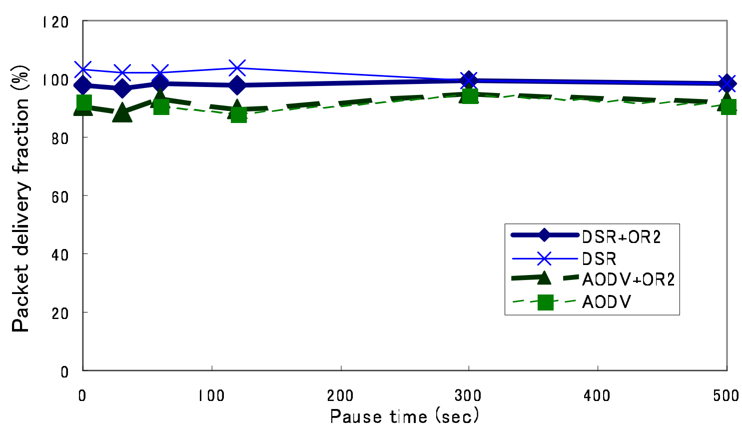


Figure 5.11: Packet delivery fraction in our oriented mobility model

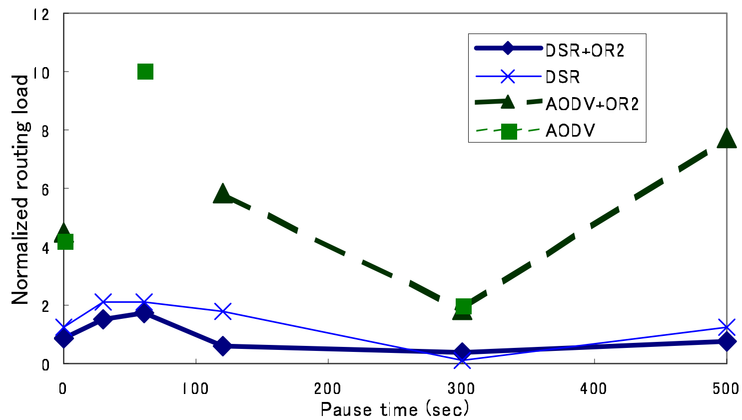


Figure 5.12: Normalized routing load in our escape mobility model

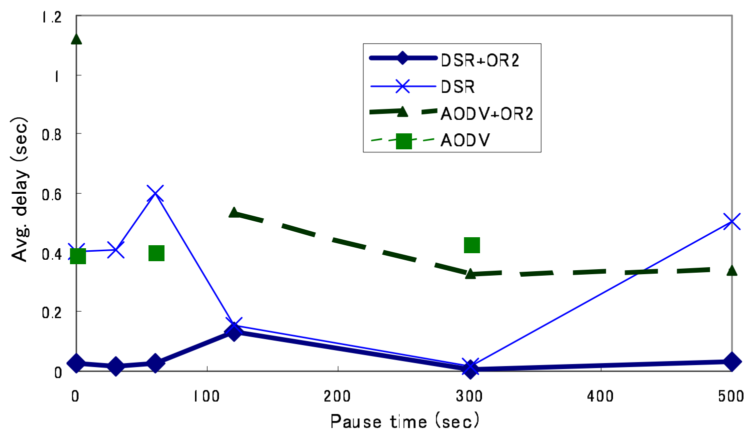


Figure 5.13: Average data packet delay in our escape mobility model

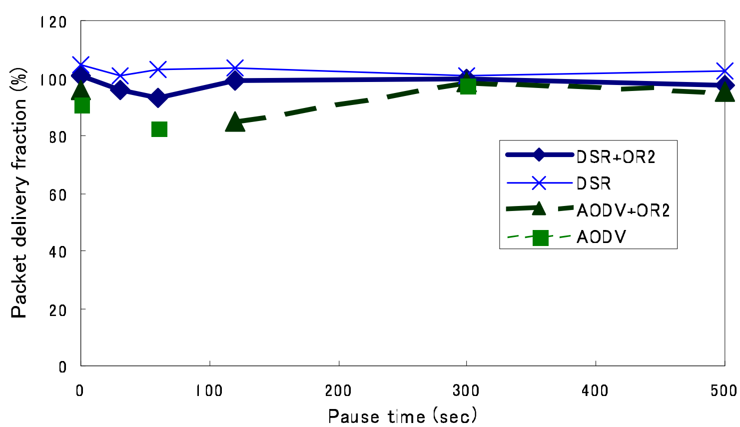


Figure 5.14: Packet delivery fraction in our escape mobility model

Chapter 6

Conclusions and Future Work

We conclude this thesis with a summary of our contributions and directions for future work.

We have proposed OR2, an adaptive path tuning algorithm for mobile ad hoc networks. While most other proposed routing protocols have no dynamic path routing mechanisms, OR2 shortens an active path adaptive to node mobility by using the notion of *proximity*. OR2 has shown its impressive effectiveness in reducing the routing overhead and the end-to-end packet delay. We can also conclude that our dynamic path shortening scheme is general and can be used with other routing algorithms and optimizations to them.

We are currently working on providing the multi-hop path shortening and also plan to make the experimental routing protocols based on signal quality metric.

6.1 Summary and Contributions

We have argued the several points as our contributions in this thesis.

- **Most conventional routing protocols do not adapt to node mobility.**

  Most previously proposed routing protocols in mobile ad hoc networks identify node mobility as link failures. In other words, they adapt to topology changes only when an active path is disconnected. Also, they cannot detect the nearness of communicating nodes. It does not say that these protocols are truly suitable for node mobility.

- **OR2 dynamically shortens the active path using the proximity.**

  In OR2, the active path adapts dynamically to node mobility based on local link quality not triggered only by link failures. OR2 skips the upstream neighbor node in a *proximity* area that indicates the nearness of two communicating nodes, and continues to shorten an active route as possible. There is no need to exchange periodic control information such as HELLO messages.

- **Previously proposed routing protocols were effectively extended by OR2.**

  As a case study, we extended conventional DSR and AODV to accommodate the dynamic path shortening mechanism. Since these two protocols are on-demand routing approaches, OR2 efficiently enhances the performance of them due to negligible routing overhead of OR2.

- **OR2 significantly reduced the routing packet overhead and packet latency.**

  The experimental results have shown that OR2 is effective in enhancing TCP throughput and reducing end-to-end delay for all relevant flows. This scheme achieved a significant reduction of a path delay while the links are still active. Similarly, simulation studies for several scenarios of high node mobility and traffic flows have revealed that adding dynamic path shortening to DSR and AODV significantly reduces the number of routing packets and the packet latency in most cases.

- **OR2 enabled more effective multi-hop shortening using the promiscuous mode.**

  DSR has the feature which realizes multi-hop shortening by overhearing packets not addressed to oneself by operating network interfaces in promiscuous mode. Cooperating with this feature as a supplement mechanism, OR2 greatly enhanced the performance of multi-hop shortening in most scenarios of various node mobility.

- **We proposed the two realistic node mobility models.**

  In most previous simulation research of ad hoc networks, simulated nodes move according to the random way-point model [14]. This model generates the random movement of nodes based on a random destination and a speed distributed uniformly between 0 and some maximum speed. In order to investigate how OR2 scheme perform in more *realistic* movement pattern, we proposed the two node mobility models: the "random oriented model" and "random escape model". These model generate the movement patterns assuming people pursuing something (e.g., *dreams*, *love*, *power*, etc) and escaping from something (e.g., *a fire*, *disaster*, *due date*, etc).

- **We experimented in *real world* environments and by simulation .**

  Most of ad hoc routing protocol proposals are evaluated through simulation, although a few of them through real environment. We have implemented OR2 as a extension to DSR on FreeBSD OS and experimented in some simple scenarios. We believe that evaluations of ad hoc routing protocols are incomplete without real-world experiments.

6.2 Future Directions

We are currently working on examining much more effects of our multi-hop shortening schemes with tapping network interfaces while taking into account its influences to power consumption. Ideally, we believe that dynamic multi-hop shortening schemes without promiscuous listening is the better approach for power saving.

We also need to analyze the decision of the SSNR threshold S_{max} value and the comparing frequency because these factors have important impact on its effectiveness of OR2. Although we consider devices operating in the ISM bands (such as Lucent WaveLANs or IEEE 802.11b wireless LAN cards) in this thesis, we hope to choose a flexible signal threshold value since signal power of received packets is dependent on what wireless devices are used.

In addition, we are now working to extend the ns2 network simulator to accurately model the physical layer behavior of the IEEE 802.11b and 802.11a wireless LAN standard [29], so that we can simulate environments of the wireless raw bandwidth from 11 Mbps (802.11b) to 54 Mbps (802.11a).

We also plan to add OR2 to Optimized Link State Routing (OLSR [12]) and Topology Broadcast Reverse Path Forwarding (TBRPF [1]) and evaluate its effectiveness. Compared with DSR and AODV, these two protocols assume larger scale ad hoc networks by using multi-point relays (OLSR) or pro-active link-state source tree computing (TBRPF).

We are presently building a blending testbed network of ad hoc networks and sensor networks. We use some operational PDAs (iPAQs [5] running Linux) in the testbed and will make such working testbed network around our office and campus as experimental multi-hop wireless networks. We also plan to make the novel experimental routing protocol based on signal power for the blending network. While signal stability based adaptive routing (SSA) protocol already was proposed [7], our planning routing protocol is based on signal strength and supplemental location information using the Global Positioning System (GPS) in the real world.

Acknowledgments

First and foremost, I would like to thank my advisor, Professor Hideyuki Tokuda, for his technical and professional advice, guidance, and encouragement.

I would like to thank Professors Jun Murai, Hiroyuki Kusumoto, Osamu Nakamura, and Masaki Minami for their valuable and technical comments on this thesis.

I would like to be extremely thankful to Dr. Yoshito Tobe, Dr. Yosuke Tamura, and Hiroto Aida for their daily supports, and valuable advice and discussions. I also thank Kaori Ishii and Mika Minematsu for helping to improve the quality of this thesis.

I am grateful to Dr. Nobuhiko Nishio, Professor Ragunathan Rajkumar, and the members of MKng and RT-HDI projects for their constructive comments.

Finally, I thank the numerous members of Tokuda, Murai, Kusumoto, Nakamura, and Minami Laboratories, especially the members of ECN, MCng, HORN and KMSF research groups for their valuable discussions and supports.

February 21, 2002

Masato Saito

References

Published Papers Related to this Thesis

- Masato Saito, Hiroto Aida, Yoshito Tobe, Yosuke Tamura, and Hideyuki Tokuda
  "OR2: A Path Tuning Algorithm for Routing in Ad Hoc Networks, "
  IEEE LCN Workshop on Wireless Local Networks (WLN '01), pp. 560-567,
  Nov. 2001.
- Masato Saito, Hiroto Aida, Yoshito Tobe, Yosuke Tamura, and Hideyuki Tokuda
  "A Dynamic Path Shortening Scheme in Ad Hoc Networks, "
  情報処理学会 マルチメディア通信と分散処理 (DPS) ワークショップ, pp. 169-174,
  Oct. 2001.
  Winner of the Best Presentation Award and the Young Researcher's Award
- 斉藤匡人, 間博人, 田村陽介, 徳田英幸
  "モバイルアドホックネットワークにおける TCP の性能解析, "
  情報処理学会 第 62 回 全国大会, Vol. 3, pp. 469-470,
  Oct. 2000.

Other Published Papers

- 青木基衣, 斉藤匡人, 間博人, 徳田英幸
  "アドホックネットワークに適した名前解決機構: RANR (ランダム割り当て名前解決機構), " 情報処理学会 モバイルコンピューティングとワイヤレス通信研究会 (MBL),

Mar. 2002.

- 高橋ひとみ, 斉藤匡人, 間博人, 徳田英幸
  “モバイルアドホックネットワークにおける TCP 転送性能を考慮した経路制御機構, ”
  情報処理学会 コンピュータシステムシンポジウム, pp. 9-16,
  Nov. 2001.
- 高橋ひとみ, 斉藤匡人, 間博人, 徳田英幸
  “モバイルアドホックネットワークにおけるパケットロス率と遅延を考慮した経路制御機構, ” 情報処理学会 第63回 全国大会, Vol. 3, pp. 297-298,
  Sep. 2001.
- 峰松美佳, 斉藤匡人, 間博人, 徳田英幸
  “モバイルアドホックネットワークにおけるリミテッドブロードキャストの測定と評価, ” 情報処理学会 第63回 全国大会, Vol. 3, pp. 299-300,
  Sep. 2001.
- Hiroto Aida, Yoshito Tobe, Masato Saito, and Hideyuki Tokuda
  “A Software Approach to Channel-State Dependent Scheduling for Wireless LANs, ” The 4th ACM International Workshop on Wireless Mobile Multimedia (WOWMOM 2001) , pp. 34-43,
  Jul. 2001.

Bibliography

[1] Bellur, B., Ogier, R., Templin, F. and Lewis, M. Topology Broadcast based on Reverse Path Forwarding (TBRPF), IETF Internet-Draft [Work in Progress] (Jan. 2002).

[2] Bharghavan, V., Demers, A., Schenker, S. and Zhang, L. MACAW: A media access protocol for wireless LAN's, Proceedings of ACM SIGCOMM'94 (Aug. 1994).

[3] Broch, J., Johnson, D. and Maltz, D. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks, IETF Internet-Draft [Work in Progress] (Mar. 2001).

[4] Broch, J., Maltz, D., Johnson, D., et al. A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols, Proceedings of ACM/IEEE International Conference on Mobile Computing and Networking (Oct. 1998).

[5] Compaq, Inc., iPAQ Handhelds, http://www.compaq.com/products/iPAQ/ (2002).

[6] Das, S., Perkins, C. and Royer, E. Performance Comparison of Two On-demand Routing Protocols for Ad Hoc Networks, Proceedings of IEEE INFOCOM'00 (1) (Mar. 2000).

[7] Dube, R., Rais, C., Wang, K. and Tripathi, S. Signal Stability-Based Adaptive Routing (SSA) for Ad Hoc Mobile Networks, *IEEE Personal Communications Magazine*, **4**, 1 (Feb. 1997), 36–45.

[8] ECKHARDT, D. A. and STEENKISTE, P. Measurement and Analysis of the Error Characteristics of an In-Building Wireless Networks, Proceedings of ACM SIGCOMM'96 (Aug. 1996).

[9] HASS, Z. and PEARLMAN, M. The Performance of Query Control Schemes for the Zone Routing Protocol, Proceedings of ACM SIGCOMM'98 Conference (Sept. 1998).

[10] IEEE 802.11 STANDARD (IEEE COMPUTER SOCIETY LAN MAN STANDARDS COMMITTEE), *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Standard 802.11 (June 1999).

[11] INTERNATIONAL TELECOMMUNICATION UNION (ITU), IMT-2000, http://www.itu.int/imt (2002).

[12] JACQUET, P., MUHLETHALER, P., QAYYUM, A., et al. Optimized Link State Routing Protocol (OLSR), IETF Internet-Draft [Work in Progress] (Mar. 2001).

[13] JOHANSSON, P., LARSSON, T., HEDMAN, N. and MIELCZAREK, B. Scenario-based Performance Analysis of Routing Protocols for Mobile Ad-hoc Networks, Proceedings of ACM MOBICOM'99 (Aug. 1999).

[14] JOHNSON, D. and MALTZ, D. *Dynamic Source Routing in Ad Hoc Wireless Networks*, In Mobile Computing, edited by Tomasz Imeilinski and Hank Korth, chapter 5, pages 153-181, Kluwer Academic Publishers (1996).

[15] KO, Y. and VAIDYA, N. Location-Aided Routing (LAR) in Mobile Ad Hoc Networks, Proceedings of ACM MOBICOM'98 (June 1998).

[16] MINISTRY OF PUBLIC MANAGEMENT, HOME AFFAIRS, POSTS AND TELECOMMUNICATIONS, 2001 WHITE PAPER Information and Communications in Japan, *http://www.soumu.go.jp/joho_tsusin/eng/Resources/WhitePaper/WP2001/2001-index.html* (2001).

[17] Ministry of Public Management, Home Affairs, Posts and Telecommunications, 2001 WHITE PAPER Information and Communications in Japan [Japanese], *http://www.soumu.go.jp/hakusyo/tsushin/h13/pdf/index.html* (2001).

[18] Mobile Ad-hoc Networks Working Group (MANET), Charter [Work in Progress], http://www.ietf.org/html.charters/manet-charter.html (1998-11-29).

[19] netperf, The Public Netperf Home page, http://www.netperf.org/ (2000).

[20] NTT DoCoMo, Inc., What's New., *http://www.nttdocomo.co.jp/info/new/* (2001).

[21] Orinoco, Inc., The WaveLAN Home Page, http://www.wavelan.com (1998).

[22] Palm, Inc., Palm Hand-helds Home Page, http://www.palm.com/products/handhelds/ (2002).

[23] Park, V. and Corson, M. A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks, Proceedings of IEEE INFOCOM'97 (Apr. 1997).

[24] Perkins, C. and Bhagwat, P. Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers, Proceedings of ACM SIGCOMM'94 (Aug. 1994).

[25] Perkins, C., Royer, E. and Das, S. Ad Hoc On Demand Distance Vector (AODV) Routing, IETF Internet-Draft [Work in Progress] (Jan. 2002).

[26] PHS MoU Group, PHS, http://www.phsmou.or.jp/ (2002).

[27] Rappaport, T. S. *Wireless Communications: Principles & Practice*, Prentice Hall (1996).

[28] Roy, S. and Garcia-Luna-Aceves, J. Using Minimal Source Trees for On-Demand Routing in Ad Hoc Networks, Proceedings of IEEE INFOCOM'01 (Aug. 2001).

[29] Takai, M., Martin, J. and Bagrodia, R. Effects of Wireless Physical Layer Modeling in Mobile Ad Hoc Networks, Proceedings of ACM Mobihoc 2001 (Oct. 2001).

[30] The Bluetooth SIG, Bluetooth, http://www.bluetooth.com (2002).

[31] The Rice University Monarch Project, The MONARCH Project at Rice University, http://www.monarch.cs.rice.edu/ (2002).

[32] The VINT Project, Network Simulator - ns2, http://www.isi.edu/nsnam/ns (2001).

[33] Tuch, B. Development of WaveLAN, an ISM band wireless LAN, *AT&T Technical Journal*, **72**, 4 (1993), 27–37.