Wireless Network Traffic and Quality of Service Support:

Trends and Standards

Thomas Lagkas
University of Western Macedonia, Greece

Pantelis Angelidis
University of Western Macedonia, Greece

Loukas Georgiadis
University of Western Macedonia, Greece

INFORMATION SCIENCE REFERENCE
Hershey · New York

Director of Editorial Content:	Kristin Klinger
Director of Book Publications:	Julia Mosemann
Acquisitions Editor:	Lindsay Johnston
Development Editor:	Joel Gamon
Publishing Assistant:	Keith Glazewski
Typesetter:	Keith Glazewski
Production Editor:	Jamie Snavely
Cover Design:	Lisa Tosheff
Printed at:	Yurchak Printing Inc.

Published in the United States of America by
Information Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue
Hershey PA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com/reference

Library of Congress Cataloging-in-Publication Data

Wireless network traffic and quality of service support : trends and standards / Thomas Lagkas, Pantelis Angelidis, and Loukas Georgiadis, editors. p. cm.
Includes bibliographical references.
Summary: "This book offers cutting edge approaches for the provision of quality of service in wireless local area networks"--Provided by publisher. ISBN 978-1-61520-771-8 (hardcover) -- ISBN 978-1-61520-772-5 (ebook) 1. Wireless LANs--Standards. 2. Wireless LANs--Quality control. I. Lagkas, Thomas, 1980- II. Angelidis, Pantelis, 1967- III. Georgiadis, Loukas, 1977- TK5105.78.W5685 2010
621.384--dc22

2009036914

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

Chapter 10

QoS Support in Multi-hop Ad-hoc Networks

Marek Natkaniec
AGH University of Science and Technology, Poland

Katarzyna Kosek-Szott
AGH University of Science and Technology, Poland

Szymon Szott
AGH University of Science and Technology, Poland

ABSTRACT

The chapter contains an overview of existing QoS solutions for multi-hop ad-hoc networks. Firstly, an introduction and short motivation are presented. The authors present an analysis of the QoS aspects of the physical layer because the wireless communication channel is constantly changing and inherently prone to errors. QoS provisioning at the data link layer is studied next. The authors focus on protocols which enable traffic differentiation, solve the hidden node problem and provide fair medium access. The chapter also deals with QoS issues at the network layer, where the authors mostly discuss QoS routing protocols. Additionally, cross-layer solutions for QoS support in multi-hop ad-hoc networks are analyzed. Finally, the expected direction of future work and a brief summary are presented.

INTRODUCTION

Multi-hop ad-hoc networks are distributed, wireless networks without infrastructure in which every node acts as both terminal and router. They are a rapidly evolving telecommunications technology which will assure connectivity for popular mobile devices (laptops, PDAs, cell phones, etc.). Ad-hoc networks can provide spontaneous communications for users which are out of reach of infrastructure networks. They can also be used as extensions to existing networks. For example, community networks can be used to offer Internet access in a neighborhood. Finally, multi-hop ad-hoc networks can provide communications in emergency situations, in which the infrastructure networks have failed or are unavailable.

Currently existing wireless networks have demonstrated that it is possible to efficiently deal with data services (e.g., Internet connectivity). Therefore, there is a growing expectation that future wireless networks will efficiently deal with multimedia

DOI: 10.4018/978-1-61520-771-8.ch010

services as well. This is caused by the growing popularity of such applications as VoIP, multimedia streaming, peer-to-peer file sharing, etc. However, the nature of ad-hoc networks makes the task of serving delay sensitive or bandwidth consuming traffic with a proper QoS very complex. In comparison to wired networks, ad-hoc networks offer much smaller bandwidth and, therefore, their design requires much more attention. Additionally, such factors as mobility of devices, unpredictable channel conditions, the hidden and exposed node problems, limited battery power, and heterogeneity of devices make QoS provisioning in ad-hoc networks a very complicated challenge.

We begin with background information regarding the challenges of QoS provisioning in multi-hop ad-hoc networks. Then, we describe QoS solutions proposed for the physical, data link and network layers. Additionally, we discuss cross-layer solutions, which combine features of the previously presented protocols. Finally, we sketch future research directions and present the most important conclusions.

BACKGROUND

QoS is a term which has been widely used in modern telecommunications. QoS is the ability to provide different priorities to different applications or flows to guarantee a certain level of performance. QoS guarantees are especially important when the network capacity is insufficient or the network is exposed to congestion. QoS is most commonly measured by the following metrics: bit rate, delay, variation of delay (jitter), packet dropping probability and bit error rate (BER). In multi-hop ad-hoc networks providing QoS is particularly difficult because of the challenges at the following layers:

- **Physical layer.** The use of wireless technologies makes links susceptible to fluctuations in the radio channel. As a result such factors as fading or interferences may lead to low bit rates and high BERs. The physical layer should quickly respond in such situations to prevent high frame error rate (FER) at the data link layer. Furthermore, random movement of mobile nodes introduces unpredictable link failures which lead to network reconfiguration. Additionally, mobile nodes are usually limited by their battery power. Power consumption can be one of the QoS attributes, because it has a strong influence on all QoS metrics.
- **Data link layer.** With the help of adequate MAC protocols, nodes need to support service guarantees for multiple traffic classes and efficiently share a common radio channel with their neighbors. Additionally, traffic scheduling schemes for real-time traffic should be used to avoid starvation of best effort traffic. The protocol should also promptly react to transmission errors and collisions. The automatic repeat request (ARQ) or adaptive error correction methods should also be used when transmission quality degrades on the data link layer.
- **Network layer.** Nodes can move in a random way. Therefore, the network topology changes unpredictably and routing protocols need to quickly adjust. Additionally, there should be a signaling protocol responsible for admission control, resource reservation, reaction to congestion and negotiation of QoS parameters.

All the mentioned features make assuring QoS in multi-hop ad-hoc networks both a challenging task and an interesting research problem. Providing a complete QoS solution for the ad-hoc networking environment requires the interaction and cooperation between three OSI/ISO layers, i.e., the physical, data link and network layers. The first two layers allow for QoS support in a single-hop connection, the third layer is responsible for end-to-end QoS. Therefore, in a multi-hop environment a cross-layer approach seems to be mandatory. However, before describing several cross-layer approaches, we first look at the QoS solutions for each layer separately.

PHYSICAL LAYER PROTOCOLS

Wireless communication is very unpredictable, because the physical properties of the wireless channel change continuously. A signal transmitted over the wireless channel is vulnerable to interferences, fading and background noise. As a result, the quality of a wireless link is considerably lower and less stable than of a comparable wired link. In addition, the operation of neighboring nodes in multi-hop ad-hoc networks, where communication usually occurs in a common channel, decreases the available capacity of wireless links. Furthermore, any link changes in a multi-hop path can easily affect the quality of an end-to-end connection. To assure a proper QoS level at the physical layer (PHY), more sophisticated control of wireless links is needed. There are a number of PHY parameters that can directly influence the considered QoS metrics, namely: transmission power, receiver sensitivity, signal-to-noise ratio, and transmission rate. They are presented next.

Transmission Power

A transmitter is an electronic device which generates electromagnetic waves, usually with the aid of an antenna. The transmitter emits these signals with a certain power level, referred to as the transmission power. The strength of these signals decreases with distance. In theory (for short, line-of-sight, LOS, distances), the receiving power is proportional to $1/d^2$, where d is the distance between the transmitter and the receiver. In practice (for long, non line-of-sight, NLOS, distances), the receiving power is proportional to $1/d^{\alpha}$, where α is the path loss exponent and $\alpha \in [4, 6]$. When a receiver moves from a LOS to a NLOS condition the received power drops (typically by 15-25 dB).

National regulation agencies set an upper limit on the transmission power to protect human health and to avoid interferences. These limits usually depend on the type of devices and frequency bands used.

Ad-hoc transmitters are usually equipped with a limited power source such as batteries or accumulators. Their transmission power should be small enough to extend their operation time and high enough to achieve acceptable signal quality at the receiver.

Receiver Sensitivity

Receiver sensitivity determines its ability to discern low-level signals. It is a measurement of the weakest signal that can be received and correctly recognized by the receiver. Therefore, this parameter is one of the key specifications of any radio device. The larger the absolute value of the negative number, the better the sensitivity. For example, a sensitivity of -95 dBm is better than a sensitivity of -92 dBm by 3 dB, or a factor of two. This also means that at a specified data rate, a receiver with a -95 dBm sensitivity can hear signals that are half the strength of those heard by a receiver with a -92 dBm sensitivity. The impact of this parameter on network performance in multi-hop ad-hoc wireless networks is studied by Ferrari, Tonguz, and Bhatt (2004).

Signal-to-Noise Ratio

The signal-to-noise ratio (SNR) is defined as the power ratio between a signal and the background noise. The higher the ratio, the less obtrusive the noise is. Due to the nature of signals, mostly characterized by a very wide dynamic range, SNR is usually expressed in the logarithmic decibel scale. The noise is calculated as the sum of the background noise and the level of interferences at the receiver. The SNR at the receiver can be improved by reducing noise at the receiver or increasing the transmission power.

Transmission Rate

Transmission rate or bit rate is defined as the number of bits that are transmitted over a wireless link within a unit of time. It is usually quantified in bits per second. There are a number of bit rate definitions depending on the protocol layer. They are listed next. The physical layer gross bit rate (also known as the raw bit rate or data signaling rate) is the total number of physically transferred bits per second over a wireless link, including both useful data as well as protocol overhead. The net bit rate (also known as the useful bit rate or information rate) is the link capacity excluding the PHY layer protocol overhead, for example framing bits, equalizer training symbols, forward error correction (FEC) codes and other channel coding. The relationship between the gross bit rate and net bit rate can be expressed using the following formula: gross bit rate × FEC code ≥ rate. For example, in one of the PHY layers defined by IEEE 802.11, for a net bit rate of 6 Mbps the gross bit rate is 12 Mbps.

The PHY layer of modern wireless standards (e.g. IEEE 802.11, IEEE 802.16) can support multiple transmission rates. In order for a wireless device to utilize high transmission rates, the received signal needs to be greater than a given threshold, which is highly dependent on receiver sensitivity. It is up to the transmission rate selection algorithm to decide which rate to choose given the current channel conditions. This has a direct influence on the QoS metrics of the channel, especially throughput. Unfortunately, modern wireless standards do not specify the method of automatic rate selection in the presence of multi-rate capable devices. As a consequence, there are several existing methods of choosing the appropriate transmission rate and vendors of wireless devices are free to choose one of them or design their own. Several of the most popular approaches are presented next.

Statistics-Based Rate Selection Algorithms

Auto Rate Fallback (ARF) is an example of a rate selection protocol based on channel statistics. According to Awerbuch, Holmer, and Rubens (2003), it is one of the most common rate selection protocols. It was developed for Lucent's WaveLAN II devices (Kamerman & Monteban, 1997) and it uses the FER to determine the quality of the channel. After successful reception of a given number of consecutive ACKs from a neighboring node, the transmission rate is increased. Similarly, after a consecutive number of ACKs have been lost, the rate is decreased. This protocol requires no changes in the IEEE 802.11 standard because the sender imposes the transmission rate. However, ARF is not the optimal strategy because it is very slow to adapt to the channel conditions. Additionally, even if the channel conditions are stable, it will unnecessarily try to change the rate. Furthermore, it can mistake collisions for channel losses. A slight improvement over ARF is a retry-based approach (Van der Vegt, 2002). In comparison to ARF, it differs in that down-scaling is performed after a number of unsuccessful retransmissions. This results in a very short response time to deteriorating links. However, the protocol behavior is pessimistic. The rate will increase only after a FER threshold has been reached. This takes longer than the down-scaling procedure. There are also other well known statistics-based algorithms such as Onoe (http://madwifi-project.org/), Adaptive Multi Rate Retry (AMRR) (Lacage, Manshaei, & Turletti, 2004), and SampleRate (Bicket, 2005). Onoe is similar to ARF but not as sensitive to individual packet loss. It looks for the highest bit rate that has a loss rate less than 50%. AMRR uses binary exponential backoff and works well for high latency systems. SampleRate uses aggressive probe packets to estimate the optimum transmission rate.

SNR-based Rate Selection Algorithms

Numerous SNR-based alternatives to the statistics-based approach have been proposed. One of them is Receiver Based Auto Rate (RBAR) proposed by Holland, Vaidya, and Bahl (2001). In this protocol, the receiver can determine the transmission rate on the basis of the SNR of each received RTS frame. Then it informs the sender about the desired rate in the CTS frame. This estimation is precise because it is done just before the transmission of a data frame. RBAR requires both changes to the IEEE 802.11 standard and the use of RTS/CTS even when there are no hidden nodes. On the other hand, it allows faster adaptability then ARF.

The Opportunistic Auto Rate (OAR) protocol (Sadeghi et al., 2002) uses a different, more efficient approach. It utilizes the coherence times of good channel conditions to send high-rate multi-frame bursts. This is similar to the TXOP feature of IEEE 802.11 (IEEE, 2007). The overhead in OAR is low because there is no contention period or sending of RTS/CTS frames in these bursts. Changing the burst size can also increase fairness (in terms of bandwidth allocation time) within the network. However, the downside to these advantages is that OAR requires modifications to the IEEE 802.11 standard. Additionally, both RBAR and OAR suffer from using pre-selected SNR thresholds, therefore they may not perform well under different channel conditions.

DATA LINK LAYER PROTOCOLS

The data link layer is responsible for establishing the physical and logical communication between network nodes. This layer consists of two sub-layers: the Medium Access Control (MAC) sub-layer and the Logical Link Control (LLC) sub-layer. Most of the issues related to QoS provisioning occur at the MAC sub-layer. This includes the aspects of efficient and fair channel access, the problem of hidden and exposed nodes, traffic differentiation, resource reservation, and traffic scheduling. The perfect MAC protocol should provide suitable mechanisms to efficiently share the available bandwidth among nodes, achieve high system throughput, support different traffic classes with the required QoS metrics, and perform well in a multi-hop environment affected by hidden and exposed nodes. The QoS solutions at the upper layers (discussed in the next subchapters) usually assume the existence of a QoS-aware MAC protocol which supports reliable unicast transmission and scheduling of real-time traffic.

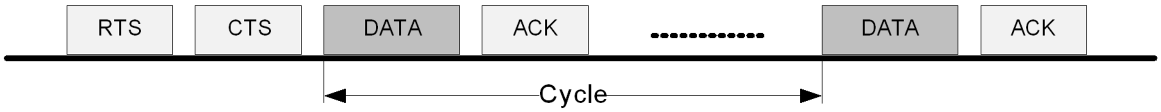
The problem of QoS support at the MAC layer has been a broad topic of research in recent years. The IEEE 802.11 EDCA (IEEE, 2007) protocol has been widely studied in the literature and is the only commercially available QoS MAC protocol. It is described elsewhere in this book. Many other MAC protocols supporting QoS have also been proposed and the most interesting solutions are shortly presented next.

MACA/PR

Multiple Access Collision Avoidance with Piggyback Reservation (MACA/PR) (Lin & Gerla, 1997) is a MAC protocol which provides guaranteed bandwidth to real-time traffic in a single hop network. Additionally, with the cooperation of QoS routing and fast connection setup mechanisms it can be used to support end-to-end multimedia delivery in a multi-hop network.

To transmit real-time frames, the sender initiates an RTS/CTS handshake and then, after receiving the CTS frame, proceeds with the DATA/ACK frames. The RTS/CTS frame transmission is used only to set up reservation for the first data exchange. If the ACK frame is not received, e.g. due to a collision, the DATA frame is not retransmitted. Moreover, if the sender fails to receive a number of consecutive ACKs, which

Figure 1. The MACA/PR protocol operation (Adapted from (Lin & Gerla, 1997))



is a configurable protocol parameter, it restarts the connection with the RTS/CTS exchange. The reservation scheme requires that each node has a Reservation Table (RT) which keeps track of transmit and receive reserved windows (for any node within transmission range). The real-time scheduling information is carried in the headers of DATA and ACK frames. A node recognizes the next transmit time from the DATA or ACK headers and records it in its RT. This allows to avoid conflicts with ongoing reservations. The sender should piggyback the reservation information for the next DATA frame transmission on the current DATA frame. The receiver reads this information, puts it in its RT and confirms it with the ACK frame. The information transmitted in the ACK frame also prevents other nodes from transmitting at the time when the receiver is scheduled to receive the next DATA frame from the sender. The propagation and maintenance of RTs among neighbors overcomes the hidden node problem for real-time traffic. A typical frame transmission cycle is presented in Figure 1.

For the transmission of best effort traffic the operation of MACA/PR is similar to IEEE 802.11 DCF. The sender must first wait for a free window in the RT. Additionally, it waits a random time in the order of a single-hop round-trip delay. Then it starts sensing the channel. If the channel is free, it initiates the transmission of RTS/CTS/DATA/ACK frames. If the channel is busy, the whole procedure is delayed until the channel becomes idle.

To summarize, in MACA/PR best effort and real-time frame transmissions can be mixed at each node, with priority given to real-time traffic. For this traffic, the protocol behaves like a Time Division Multiplexing (TDM) system. Best effort frames can easily fill all empty windows in the cycle to achieve high overall protocol efficiency.

IEEE 802.11 DCF with a Multi-Priority Scheme

A variation of the IEEE 802.11 DCF protocol to support different traffic classes is proposed by Deng and Chang (1999). There are four traffic classes differentiated by their inter-frame space (IFS) and backoff periods. For higher priority traffic a node waits for the channel to be idle for PIFS, for lower priority traffic it waits for DIFS. Even if a node is waiting for PIFS it can still lose the contention if it chooses a backoff larger than other nodes (in particular, nodes which waited for DIFS).

The proposed scheme is simple and can be easily implemented in IEEE 802.11 devices. Simulation results (Deng & Chang, 1999) show that the proposed protocol has better performance than DCF in terms of throughput, access delay, and loss probability for higher priority traffic. Unfortunately, the considered scheme cannot provide deterministic delay bounds for higher priority traffic. Moreover, the lowest priority traffic suffers from much higher delay compared to DCF because longer backoff periods are selected even when no higher priority traffic is being transmitted.

Black Burst Contention Scheme

The Black Burst protocol (Sobrinho & Krishnakumar, 1999) provides a bounded time delay for real-time traffic in ad-hoc networks. This protocol is distributed and based on the CSMA access method. It ensures collision-free transmission of real-time frames. Nodes sending real-time traffic use pulses of energy, which are called Black Bursts (BB), to contend for medium access. The length of these pulses is proportional to the time the nodes had to wait for the channel to become idle. This delay is measured from the first attempt to access the channel by a node until its transmission starts. After transmitting its BB, the node waits for a specified time interval to see if any other node is transmitting a longer BB. If the channel is perceived idle after this interval, then the node can immediately transmit its real-time frame. Otherwise, it waits for the next channel access cycle and repeats the algorithm. A round-robin discipline among nodes transmitting real-time frames is enforced, which results in bounded access delays. The BB protocol can also support asynchronous data transmission. Nodes transmitting asynchronous data frames use a longer IFS than nodes sending real-time traffic. The BB contention scheme guarantees that real-time frames are always favored over asynchronous data frames. The BB protocol can be easily combined with DCF and implemented, with minor modifications, in WLAN cards. Unfortunately, the protocol does not consider the exposed node problem.

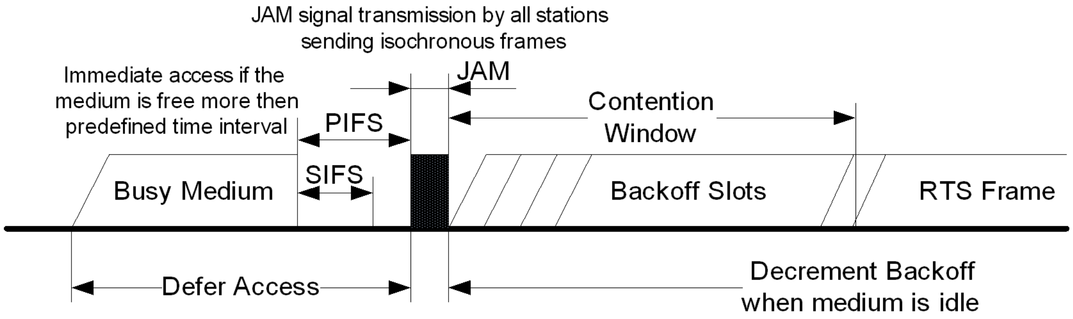
PUMA

The Priority Unavoidable Multiple Access (PUMA) protocol (Natkaniec & Pach, 2002) enhances DCF to support strict priority isochronous traffic transmission in ad-hoc mode. Three different time intervals are defined: SIFS, PIFS, and DIFS, where DIFS>PIFS>SIFS. This is similar to the IEEE 802.11 standard. Every active station measures these intervals after the end of each frame to determine the moment it can start its own transmission. The station proceeds with its isochronous transmission if the medium is determined to be idle for an interval that exceeds PIFS. All stations sending isochronous frames should start its transmission simultaneously and send the JAM signal. The JAM signal consists of pulses of energy (similarly to the BB protocol) and has the length of one slot. This signal informs all other stations (especially stations sending asynchronous frames) that in their neighborhood an isochronous transmission will begin. This means that all other stations have to defer their transmission until the reception of an RTS or CTS frame to update their network allocation vector (NAV). A random backoff interval is then selected and used to initialize the backoff timer. The backoff timer is decremented only when the medium is idle. It is frozen when the medium is busy until the next PIFS period. A station initiates an RTS frame transmission when the backoff timer reaches zero. To increase the efficiency of PUMA in a scenario with high load and a large number of contending stations, a backoff scheme called DIDD was used as the default backoff mechanism (Natkaniec & Pach, 2000). A typical isochronous frame transmission scenario is presented in Figure 2.

On reception of an RTS frame the receiver responds with a CTS frame, which can be transmitted after the channel has been idle for a time interval exceeding SIFS. After a successful exchange of RTS and CTS frames the transmitter sends its DATA frame in a collision free manner. In the case when a CTS frame is not received within the predetermined time interval, the RTS is retransmitted according to the backoff rules. Additionally, a multiple frame transmission mechanism is implemented in PUMA to increase the protocol performance measures. Data frames are transmitted in sequence without the risk of collision after a successful medium reservation through the RTS/CTS exchange. The number of data frames transmitted in sequence is configurable.

PUMA has the following additional features. The life-time of each isochronous frame is measured. If it reaches its limit and the frame cannot be sent to its destination it is treated as useless and removed from the station buffer. Furthermore, PUMA allows controlling the minimal amount

Figure 2. The operation of PUMA for isochronous traffic (Adapted from (Natkaniec & Pach, 2002))



of asynchronous traffic by introducing an additional timer. It is used to measure the life-time of asynchronous frames located in the source station buffer. An asynchronous frame located in the head of the queue gets a higher priority if its life-time is reached. Its priority becomes equal to the priority of isochronous frames.

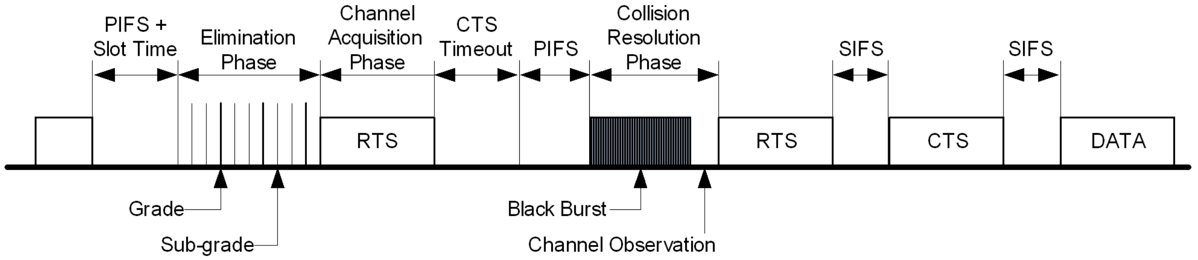
ES-DCF and DB-DCF

Two variants of DCF that incorporate explicit support of real-time traffic are proposed by Pal, Dogan, and Ozguner (2002). Both protocols use deterministic collision resolution algorithms in order to provide QoS guarantees for different traffic classes. The interesting fact is that both schemes do not apply any backoff mechanism.

The Elimination by Sieving DCF (ES-DCF) protocol defines three phases of operation: elimination, channel acquisition and collision resolution. In the elimination phase, every node receives a grade which depends on the deadlines and priorities of its real-time frames. A lower numerical grade means that a frame has a closer deadline, which also means that it has waited in the queue for a longer time, and its *channel-free-wait-time* parameter is smaller. This parameter also depends on real-time frame priority because ES-DCF defines two classes of real-time traffic. Additionally, finer sub-grades are assumed by choosing random numbers from a specified interval to avoid the existence of two or more real-time frames from different nodes with the same grade (similar frame deadlines usually mean the same grade). After a node has waited for the *channel-free-wait-time* the channel acquisition phase begins. If the channel is idle, the node transmits an RTS frame. After receiving the RTS frame all other nodes defer their accesses until the next channel acquisition phase. If the node receives a CTS frame, it can begin the transmission phase, in which its real-time frame can be sent. Otherwise, the collision resolution phase is initiated by transmitting BBs of lengths equal to the unique node ID numbers. The node that sends the longest BB wins the contention and accesses the channel at the subsequent attempt. It should be pointed out that the collision resolution phase introduces the blocked-access feature, where all nodes that have experienced a collision in the channel acquisition phase use the smallest *channel-free-wait-times*. This means that they can pre-empt all other nodes during the collision resolution phase. The operation of ES-DCF is presented in Figure 3.

The Deadline Bursting DCF (DB-DCF) protocol is similar in operation to ES-DCF. In the first phase, called the BB contention phase, a real-time node starts the transmission of a BB proportional to the urgency of its real-time frames (which corresponds to their deadlines). The BB lengths are defined as multiples of a BB slot time. After sending its BB transmission, the node checks the

Figure 3. Phases of ES-DCF operation (Adapted from (Pal, Dogan, & Ozguner, 2002))



channel to determine any longer BB duration. If it senses any other BB transmission, it defers its channel access until the next channel access cycle, where a new BB length is calculated and transmitted. The channel acquisition and collision resolution phases of DB-DCF are exactly the same as ES-DCF. The operation of the DB-DCF protocol is presented in Figure 4.

Both protocols assume asynchronous frame transmission after the DIFS period. However, the ES-DCF protocol cannot be directly combined with an IEEE 802.11 DCF implementation because the *channel-free-wait-time* intervals for asynchronous data frames are longer than DIFS. Moreover, a high volume of real-time traffic can completely suppress asynchronous data transmission. Simulation results show that ES-DCF is more efficient for hard real-time traffic (i.e., real-time frames are dropped when expired), while DB-DCF behaves better for soft real-time traffic (Dogan & Ozguner, 2002).

QoS Enabled MAC for Multi-Hop Ad-Hoc

The MAC protocol proposed by Ying, Anand, and Jacob (2003) provides service differentiation for real-time constant bit rate traffic, real-time variable bit rate traffic and asynchronous non-real-time traffic. For real-time traffic it uses a distributed mechanism for scheduling and reserving the radio channel. According to the proposed scheme every non-real-time frame and the first frame from a real-time session (or burst) begins a typical RTS/CTS/DATA/ACK sequence. The subsequent frames in a real-time burst are transmitted using the DATA/ACK exchange. The protocol differentiates between ACK and DATA frames of real-time (R-ACK, R-DATA) and non-real-time traffic (D--ACK, D-DATA). R-ACK performs a reservation for the next R-DATA frame because RTS/CTS frames are transmitted only at the beginning of the transmission of real-time traffic.

Figure 4. Phases of DB-DCF operation (Adapted from (Pal, Dogan, & Ozguner, 2002))

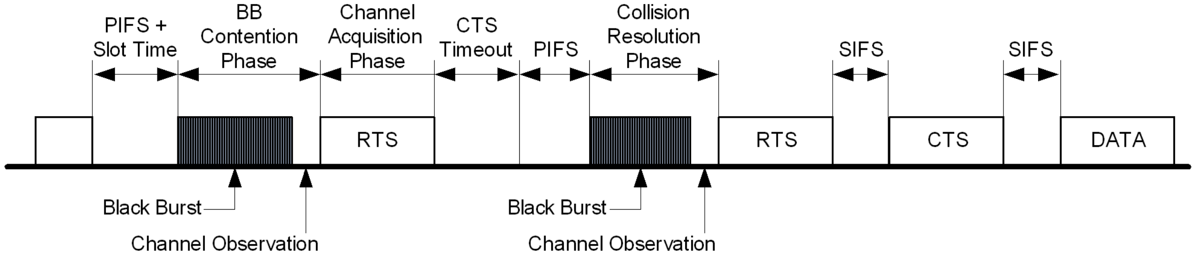
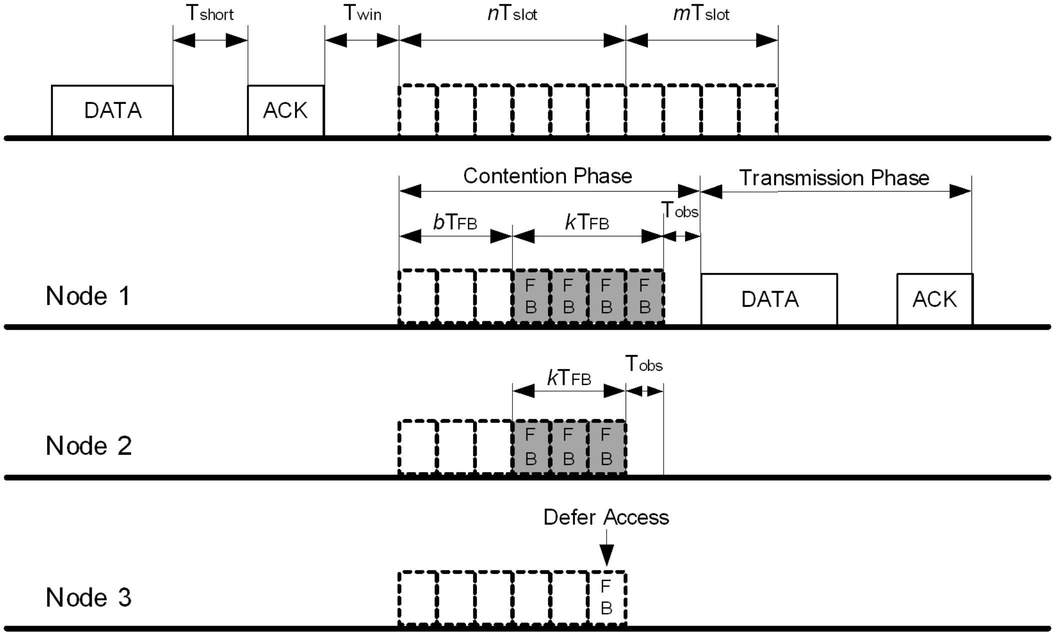


Figure 5. QMA protocol operation (Adapted from (Wang & Liu, 2007))



Each node maintains a receive and transmit reservation table (Rx RT and Tx RT, respectively). They contain reservation windows in which the neighboring nodes have scheduled to receive and transmit real-time frames. When a node receives R-DATA, it estimates the next Tx time and writes it into its Tx RT. After receiving R-ACK, it writes the reservation into its Rx RT. If a node wants to initiate a new non-real-time or real-time session, it has to check its Rx RT and Tx RT to find a free slot in which no neighbor is scheduled to receive or transmit during the time needed for the RTS/CTS/DATA/ACK transmission. In the case of receiving an RTS frame, it checks its Rx RT and Tx RT tables before responding with a CTS frame. After a successful DATA transmission, an ACK frame is expected. If no ACK is received, the sender node assumes that a collision has occurred and enters the backoff stage. The protocol uses the same binary exponential backoff mechanism as defined in the IEEE 802.11 standard.

The proposed protocol guarantees bounded delays for real-time traffic, however, its effectiveness highly depends on overhearing R-DATA and R-ACK frames. The authors have proven that their protocol achieves lower maximum and average delays for real-time traffic than EDCA, BB, and MACA/PR. The reservation tables help avoid collisions in hidden node scenarios, which results in small packet loss rates.

QMA

Wang and Liu (2007) have proposed a QoS-based Multiple Access (QMA) protocol for ad-hoc networks. This protocol supports two types of traffic: real-time and best effort. In QMA, the channel access cycle is divided into a contention and transmission phase (Figure 5). Each node is obliged to sense the medium for a time interval T_{win} before accessing it. If the medium is idle, the node can start the contention phase. The contention phase is composed of $n + m$ slots which are assigned to real-time (n slots) and best effort traffic (m slots). The nodes sending best effort traffic are allowed to broadcast a forecast burst (FB) in m slots only when all n slots are idle. This assures priority of real-time over best effort traffic.

A node chooses a number b, which is a random variable with a truncated geometric distribution. If a node senses the first b slots idle, it immediately starts transmission of $k \cdot$ FB slots. Otherwise, it stops its backoff. The k parameter depends on the frame lifetime. The frame with the earliest deadline has the largest k value. After transmission of FBs the node senses the medium. If the channel is busy, it means that there must be at least one contending node with a higher priority frame, and the node with a lower priority frame has to backoff. Otherwise, if the channel is idle, the node can start its DATA transmission. Successful reception of DATA is confirmed with an ACK frame. An example of the operation of QMA is presented in Figure 5.

The QMA protocol guarantees that only the nodes that start transmission of FBs in the same slot can successfully survive the contention phase. The node which sends the largest number of FBs wins the overall contention. The simulation results show that the QMA protocol with its well-organized collision resolution mechanism obtains a higher efficiency than IEEE 802.11 EDCA (Wang & Liu, 2007). Unfortunately, the protocol supports only two types of traffic. Furthermore, high real-time traffic can completely starve best effort traffic.

NETWORK LAYER PROTOCOLS

The network layer is mostly responsible for ensuring QoS routing, admission control and signaling. In this subchapter we mostly discuss QoS routing protocols, because they are the main focus of research in this layer. The main goals of such protocols are the following:

- Estimate the available network capacity. This information is often used to perform admission control.
- Find loop free routes which satisfy QoS requirements of flows. QoS constraints typically taken into account are jitter, delay, bandwidth and power consumption.
- Reserve the required resources.
- Maintain routes by utilizing redundant routes, predicting route breaks, and using a route recovery mechanism.

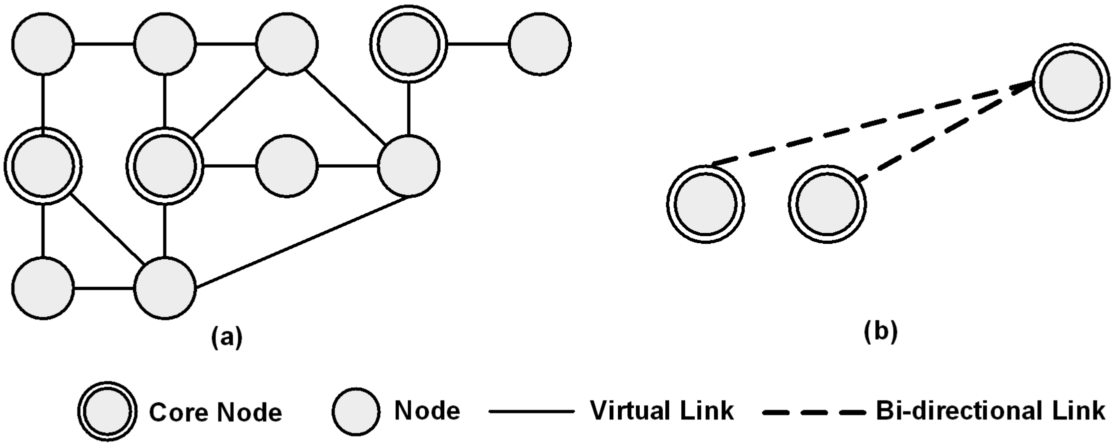
In multi-hop ad-hoc networks routing is very challenging mostly because of two reasons. Firstly, bandwidth is very limited and, therefore, a QoS routing protocol must have small overhead. Secondly, the topology is constantly changing and, therefore, the reserved resources cannot be hard guaranteed. The most important QoS routing protocols are described next in chronological order.

CEDAR

Core-Extraction Distributed Ad-hoc Routing (CEDAR) (Sinha, Sivakumar, & Bharghavan, 1999) is a hierarchical QoS routing algorithm for MANETs. It consists of three key components: core extraction, link state propagation, and route computation.

The main goals of CEDAR are, firstly, to compute routes quickly and, secondly, to react to the network changes without large amounts of state propagation. Therefore, the protocol focuses on rapid reaction to network changes rather than on the optimality of routes. Furthermore, there are several basic assumptions made in CEDAR. Firstly, nodes communicate on the same channel. Secondly, transmitters have a fixed transmission range. Thirdly, networks are small or of a medium size (tens to hundreds of nodes). Finally, the MAC-link layer can be used to estimate available link bandwidth.

Figure 6. Network with core nodes (a) and corresponding core graph (b). (Adapted from (Sinha, Sivakumar, & Bharghavan, 1999))



Route Discovery and Maintenance

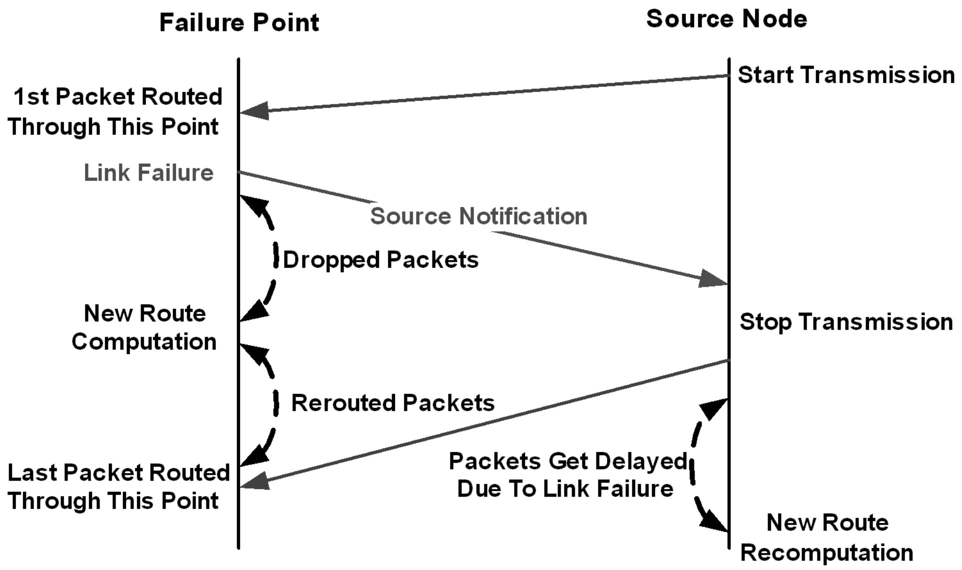
CEDAR uses a greedy algorithm to create an approximate minimum dominating set (DS) of the core nodes (Figure 6). This set is chosen in a distributed manner. Each core node has enough local topology information to reach the domain of its nearby core nodes and set up paths (virtual links) to them. Other MANET members need to choose a *dominator* from the DS because only the core nodes maintain local topology information, participate in the exchange of network state information, discover and maintain routes, and perform admission control. When a host loses connectivity with its *dominator* (due to mobility) it either finds a new *dominator* from the neighboring core nodes, nominates one of its neighbors to join the core, or itself joins the core.

CEDAR assumes that each core node not only has up-to-date information about its local topology but also about stable high bandwidth distant links. To achieve this goal, it adopts *increase* and *decrease waves*. The former provide information about an increase of the available bandwidth. They are propagated locally and they are periodical. The latter provide information about a decrease of the available bandwidth. They are propagated distantly and they are sent immediately after each bandwidth change. These waves are generated every time when an estimate of the available bandwidth changes by some threshold value. CEDAR propagates information about the state of stable high bandwidth links throughout the core, and keeps information about the sate of low bandwidth or unstable links locally. This is possible because for an unstable link the *decrease wave* stops the *increase wave* from propagating.

The QoS route computation scheme in CEDAR involves the following three phases:

- **Establishment of the core path:** Firstly, a source node sends a request to its *dominator*. Then the *dominator* forwards this message to each of its nearby core nodes using the core broadcast algorithm. After the *dominator* of the destination node receives this message it responds with a source routed unicast *core_path_ack* message. When the *core_path_ack* message is received by the source *dominator*, the core path establishment phase is finished and the QoS route computation phase can be started.

Figure 7. Reestablishment of a route (Adapted from (Sinha, Sivakumar, & Bharghavan, 1999))

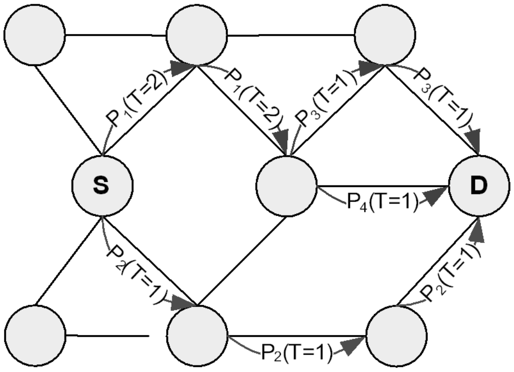


- **QoS route computation:** The *dominators* of the source and the destination nodes have partial knowledge about the local topology and the remote stable high bandwidth links. On this basis the *dominator* of the source node is able to compute a path to the furthest, intermediate core node in the core path which can guarantee the requested bandwidth. The route selection is based on a two phase Dijkstra's single source shortest path algorithm in order to find the shortest-widest path. Then, the intermediate core node starts a QoS route computation using its local state. As a result, the concatenation of the partial paths computed by the core nodes provides a QoS core route from the *dominator* of the source node to the *dominator* of the destination node.
- **Dynamical re-establishment of routes for ongoing connections** (Figure 7)**:** In the case of link failure or a topology change two mechanisms can be used: QoS route re-computation at the failure point and QoS route re-computation at the source. The former is suitable for failures occurring near the destination. The latter is effective when the failures occur near the source.

TBP

Ticket-Based Probing (TBP) (Shigang & Nahrstedt, 1999) is a multipath distributed routing scheme which uses tickets to limit the number of candidate paths. It can handle different QoS constrains (such as bandwidth and delay). The basic scheme of TBP is as follows. When a source node wants to find QoS paths to a destination node, based on the available network state information, it issues routing messages (probes, P) with a given number of tickets (T). When a probe message with more than one ticket is received by an intermediate node, based on its local state information, the node may decide to split the probe and forward different tickets on different downstream sub-paths. When the destination node receives the probe, a possible path from the source to the destination is found. TPB assumes that one ticket is a permission to search a single path and that one probe should

Figure 8. TBP operation (Adapted from (Shigang & Nahrstedt, 1999))



carry at least one ticket. Therefore, the maximum number of probes and the maximum number of searched paths are dependent on the number of tickets issued from the source node.

An example of TBP operation is presented in Figure 8 where S is the source node, and D is the destination node. Two probes are initiated at S: P1 with two tickets, and P2, with one ticket. P1 is split into two probes (P3 and P4) at one of the intermediate nodes. As a result three paths are found.

BR

The Bandwidth Routing (BR) protocol (Lin & Liu, 1999) uses bandwidth QoS constraints to establish paths between pairs of nodes. The protocol is designed for TDMA networks. Therefore, bandwidth is measured in terms of free timeslots available. BR works in conjunction with the Destination-Sequenced Distance Vector (DSDV) routing scheme.

The performance of BR is based on the source node's knowledge of end-to-end bandwidth available to any possible destination, which enables efficient support of real-time applications and helps establish QoS routes. Additionally, the protocol supports admission control. The assumptions made by Lin and Liu are the following. Transmissions are half-duplex, i.e., each node can either transmit or receive data. The channel is time slotted and either a time synchronization mechanism or a global clock is provided. In each data slot one data packet can be transmitted.

Bandwidth Reservation and Slot Assignment

In this protocol each node has its own set of free slots. A common set of free slots between two adjacent nodes denotes the *link bandwidth* between these two nodes. The *path bandwidth* is calculated on a hop-by-hop basis along the whole path from a source node to a destination node and it is the set of available slots between the two nodes.

The protocol assumes that each frame is divided into two phases: the control phase and the data phase. The control phase is used to perform the control functions (e.g., slot and frame synchronization, power measurement, setup of virtual connections, building of routing tables). The amount of slots/frames assigned to a path is determined by a QoS requirement. The control phase uses pure TDMA with full power transmission in a common code, i.e., each node broadcasts its routing information (obtained by DSDV) and its QoS requirements to its neighboring nodes in predefined timeslots. In noisy environments an additional ACK mechanism is employed to assure correct data exchange. At the end of the control phase each node can schedule free slots, verify the failures of reserved slots and drop expired packets. This is possible because nodes have information about channel reservations made by their neighbors. In the data phase, the required bandwidth resources are first pre-allocated and then traffic exchange may take place. Therefore, the protocol assumes that the amount of bandwidth along the path is computed and known to all nodes. This information becomes useful when a new request enters the network because it can immediately determine if a new flow can be accepted or not.

The free slots are assigned during session setup by the slot assignment algorithm. Every intermediate node and the destination node, after receiving a reservation request from the source node, checks whether it has enough free slots to receive and forward the data packets. If the required number of slots is available, they become reserved, the routing table gets updated and the session setup is forwarded to the next neighbor. Otherwise, the reservation fails and all current reservations on the path back to the source node are cancelled with the use of a RESET packet. When the end-to-end path reservation is successful, the destination node sends a REPLY packet to the source node to acknowledge a positive connection setup.

Route Maintenance

Each node holds two secondary paths in its routing table which can be used when the primary route fails. The secondary path is chosen for a new primary path if it satisfies the QoS requirements of a particular flow. The primary route does not have to be the highest bandwidth path; it must be the shortest one meeting the QoS requirements.

BRuIT

Bandwidth Reservation under InTerferences influence (BRuIT) (Chaudet & Guérin Lassous, 2001) is a distributed signaling protocol for bandwidth reservation which takes into account the existence of interferences between nodes. The authors concentrate on the bandwidth metric because it may affect such parameters as delay or jitter. The performance of BRuIT is based on periodically determining which nodes interfere with other nodes and what are their bandwidth reservations. BRuIT is implemented over a reactive routing protocol.

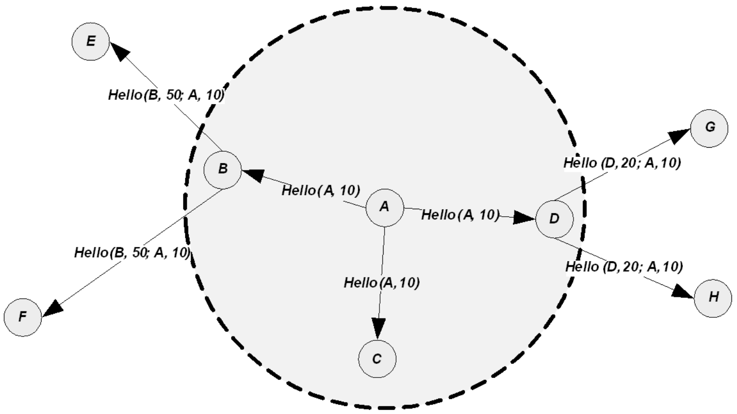
Neighborhood Discovery

In order to obtain knowledge about its neighborhood, each node periodically broadcasts a *Hello* packet. This packet contains the sender's address, the total bandwidth which it will use for already accepted flows and the information learnt from its neighbors which are *k* hops away. Propagation of *Hello* packets within two hops is presented in Figure 9. The reception of these packets helps each node compute the remaining bandwidth and allows more precise admission control.

Route Discovery

Every source node, to reserve bandwidth for a single flow, broadcasts a route request message including information about the destination node's address and the amount of requested bandwidth. Then, the admission control procedure is performed at each intermediate node. If it fails, the request is dropped. Otherwise, the request is further broadcasted until it reaches the destination node. Upon reception of more than one request for a particular flow, only the first one is accepted. After the destination node receives the request packet, it does admission control and, if the amount of free bandwidth is sufficient it replies with a route reply message. When intermediate nodes receive the route reply packet, they check if they still have enough bandwidth to satisfy the QoS requirements of the flow. If the amount of available bandwidth is not satisfactory, the reply message is dropped. Otherwise, the requested amount of bandwidth is reserved. Finally, after the source node receives

Figure 9. Broadcasting of Hello packets (Adapted from (Chaudet & Guérin Lassous, 2001))



the reply from the destination node, it starts its data transmission.

Route Maintenance

To deal with mobility each node periodically sends *Hello* packets. Lack of *Hello* packets from a given neighbor informs about its movement or failure. Additionally, to deal with route breaks, reservations made by intermediate nodes have a soft-state. Therefore, if there is no data exchange on a given path for a certain time, the reserved bandwidth is released. Finally, to deal with misbehaving applications (i.e., applications using more bandwidth than requested) and to avoid frequent interferences caused by the exchange of false information, each sender node shapes its traffic using token bucket filters.

TDR

Trigger-based Distributed Routing (TDR) (De et al., 2002) is an on-demand routing protocol designed to support QoS-aware real time applications. It aims to effectively deal with link failures. Additionally, to reduce control traffic, TDR utilizes GPS-based location information and on-demand route discovery.

Database Management

In TDR all hosts must keep local neighborhood and routing information. Therefore, each host maintains two databases:

- **Local Neighbor Database**. It stores information about the location and the mobility of neighboring nodes (carried in periodically sent beacons) and the power level of these beacons.
- **Activity-Based Database**. It stores routing information valid for each session. Depending on the role of the node which stores the routing table it is called the source, destination or intermediate node database. Obviously, nodes may require maintaining some or all types of databases for different ongoing sessions.

Route Discovery

In TDR the source node floods route discovery packets to its neighbors. In order to reduce the signaling overhead, only some of the neighboring nodes are considered as possible next hops in the route. They are selected on the basis of the received power level which has to be greater than a predefined threshold.

At the beginning of the route discovery process, the source node must check if it has enough available bandwidth to satisfy the bandwidth requirement in the request. If enough bandwidth is available it is temporarily reserved by the source node. The reservation time is equal to the time in which it is expected that an acknowledgement packet from the destination node will be received. A valid route to the destination node is found with the help of a modified breadth first search algorithm. All intermediate nodes, upon receiving the first discovery packet, perform admission control based on available bandwidth. When the destination node receives the discovery packet and meets the QoS requirements of the session, it accepts the discovered route and sends an acknowledgement to the source node. If the location of the destination node changes it sends an appropriate location update message. Additionally, in order to avoid routing loops, TDR requires intermediate nodes to accept only one route discovery packet per session.

Route Maintenance

Route maintenance in TDR is based on three received power levels: P_{th1}, P_{th2}, $P_{cr,}$ where $P_{th1} > P_{th2} > P_{cr}$. When the downstream received power level is lower than the critical limit P_{cr}, the source-destination route gets disrupted until an alternate route is set up by the source node. If the power level is between P_{th2} and P_{cr}, the source node is notified by an intermediate node with a rerouting request. Finally, if the power level is between P_{th1} and P_{th2}, the intermediate node initiates the rerouting process.

QoS-AODV and QoS-TORA

Gerasimov and Simon (2002) propose QoS extensions to the AODV and TORA (Temporally Ordered Routing Algorithm) routing protocols. These extensions add scheduling and resource reservation for a TDMA-like data link mechanism. The modified protocols are called QoS-AODV and QoS-TORA, respectively. They combine information from the data link layer and the network layer.

QoS-AODV

At the beginning of the path discovery procedure a source node checks if it has enough residual bandwidth to any of its neighbors to meet the requirements of an application. If there is enough bandwidth available the source node floods a modified route request (RREQ) packet to its neighboring nodes. The modifications include application ID and number of slots required for successful reservation. Upon receiving RREQ, each intermediate node performs admission control based on available bandwidth. Additionally, each intermediate node checks if it has an entry in its routing table corresponding to the received application ID. If the entry does not exist, it is created. Otherwise, the node checks if the RREQ received is newer than the one it has and, if necessary, updates its routing table. Each entry in a routing table contains addresses of three downstream nodes (in the direction to the source) as well as bandwidth schedules between those nodes. The bandwidth information is included in order to inform of the calculated bandwidth, prevent direct collisions and avoid the hidden node problem. Finally, upon receiving the RREQ packet, the destination node checks its residual bandwidth. If the admission control succeeds, it starts the reservation protocol. Additionally, if more than one RREQ is received by the destination node, it chooses the one with enough bandwidth, not the one with the fewer number of hops as in the original AODV.

QoS-TORA

There are two possible means of route discovery in QoS-TORA. Firstly, if a best-effort path from a source to a destination node does not exist a TORA *Query* packet is sent. Secondly, if any path exists, the source node sends a *Bandwidth Query* packet, which contains the number of slots needed and the application ID, on a known path. Upon receiving this packet, the destination node does admission control based on its residual bandwidth and broadcasts an *Update Bandwidth* packet, which contains the application ID, number of slots required and the source node ID. After an intermediate node receives this packet it calculates a new path bandwidth, checks if it has an entry for the received application ID, and updates its routing table with the best QoS path. The source node has to wait for several *Update Bandwidth* packets from its neighboring nodes before it can start the reservation protocol. This is done in order to make the selection of a QoS path possible and, upon path break, skip the route discovery procedure and immediately use an alternative path.

QoS-AODV

QoS-AODV (Chenxi & Corson, 2002) is a routing protocol using TDMA. Its operation is limited to small networks. The basic idea is that QoS routes are built only if necessary. The authors assume that applications are session-oriented and have a constant bandwidth requirement. The QoS requirement of a session is specified by the number of time slots needed on a route from a source to a destination node. Therefore, QoS-AODV finds both the route between the two nodes and the slots for each link on a path.

Bandwidth Calculation

The source node specifies the required number of slots along a QoS path. Each node along this path must find at least the required number of free slots for a transmission to its downstream neighbor. The algorithm looks for non-conflicting slots only on three adjacent links. Therefore, QoS-AODV aims for the local rather than the global maximum bandwidth. After the local maximum bandwidth is found, the calculation is propagated along the path to the destination node.

Route Discovery

Network bandwidth is calculated in conjunction with route discovery, i.e., to find a QoS path a source node floods a route request (RREQ) packet and, simultaneously, bandwidth is calculated on a hop-by-hop basis. If the requested bandwidth is not available at any intermediate node, RREQ is dropped. Otherwise, upon receiving the request, a destination node sends a route reply (RREP) packet to the source node and reserves necessary transmission slots. Additionally, if the destination node receives multiple RREQs, the first request satisfying the bandwidth requirement is accepted and the others are ignored. This is done in order to reduce the delay of the route discovery procedure.

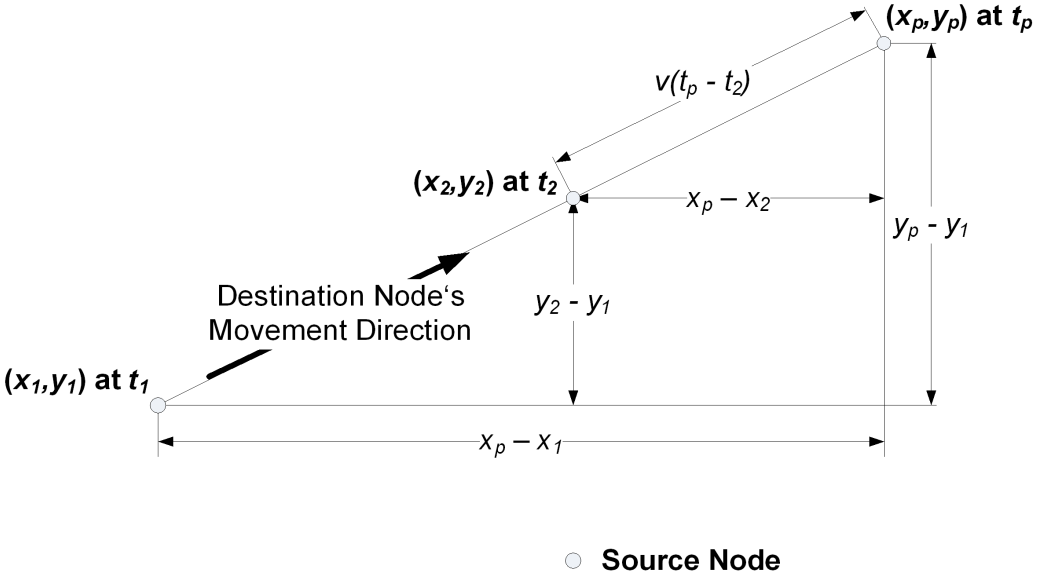
Bandwidth Reservation

QoS-AODV proposes to use soft-sate bandwidth reservations for QoS paths. If a particular path is not used for some time, its entry is dropped from the routing tables. The authors define several states of a QoS route, e.g., indicating that the route is established or broken. Transitions between states are triggered by receiving or transmitting a packet or by the expiration of a timer associated with a particular state. In addition, QoS-AODV uses two timers, *Route setup time* and *Route life time*. The former is used during route discovery and route repair. It is equal to the round-trip time from the source to the destination node. The second timer is used as a maximum interval for data arrival. It helps detect broken paths.

PLBQR

Predictive Location-Based QoS Routing (PLBQR) (Shah & Nahrstedt, 2002) consists of an update

Figure 10. Location prediction using last two updates (Adapted from (Shah & Nahrstedt, 2002))



protocol, a location-delay prediction scheme and a QoS routing protocol.

Update Protocol

Distribution of geographical location (obtained through GPS) and resource information is done by the update protocol. PLBQR considers two types of updates: a *Type 1 update* (generated periodically by each node) and a *Type 2 update* (generated when there is a considerable change in a node's speed or direction). All update packets contain timestamps, current geometric coordinates, movement direction, velocity, resource information (for QoS routing) and a single-bit *motion stability parameter*. This parameter informs the QoS routing protocol about the type of an update, i.e., it helps assign dynamic nodes as intermediate nodes only for connections without strict delay or jitter constraints.

Location-Delay Prediction Scheme

Before any source node can establish a connection to a particular destination node, it has to predict the geographical location of all intermediate nodes and the destination node at time t_p when a packet from the source node will reach them. Additionally, the propagation delay must be known in order to estimate t_p. Location predictions are performed based on updates received from other nodes.

An exemplary location prediction is presented in Figure 10. In the figure, $(x_1; y_1)$ *at* t_1 and $(x_2; y_2)$ *at* t_2, where $t_2 > t_1$, are the latest two updates from a destination node received by the source node and v is the velocity of the destination node. The value of t_p is set to a sum of the current time and the time to reach the destination node from the correspondent node. Using simple calculations (based on triangle similarity and the Pythagoras theorem), the location $(x_p; y_p)$ of the destination node at time t_p can be predicted.

The destination node uses the same calculations to judge if there is a considerable change in its location and decide if a *Type 2 update* message must be sent to its neighbors. Additionally, PLBQR assumes that the end-to-end delay of a data packet transmission from the source to the destination node will be the same as the delay experienced by the latest update from the destination to the source node.

QoS Routing Protocol

Based on the information received from the update messages, each node has up-to-date information about the whole network. On this basis it can compute a route to any destination. Each node maintains two tables. The update table contains information learnt from the update messages and, for each node, a *proximity list* with a list of nodes lying within a distance of one and a half transmission range. The route table contains information about active connections set up by the source node. Thanks to the *proximity list* a given source node, during its QoS routing process, may take into account nodes which moved into the transmission ranges of other nodes as possible intermediate nodes.

When an update message is received at the source node, it checks if any of the known routes is damaged or is about to be broken by either node movements or by being unable to satisfy the QoS requirements of a connection. In both situations a route re-computation must be initiated. Because such information as remaining battery power, transmission range, and CPU utilization are exchanged in the update packets, the re-computation of a route may begin before it is really broken.

The routing algorithm itself works as follows. At first the source node runs the location and delay predictions for each node in its *proximity list*. On this basis it determines which nodes have enough resources available to satisfy the QoS requirements of a request. Then, it finds all possible routes towards the destination node by simultaneously performing route discovery and admission control on a hop-by-hop basis. If more than one route satisfies the QoS requirements, the geographically shortest one is selected for data transmission.

QoS-OLSR

Ying, Kunz, and Lamont (2003) propose several algorithms which allow OLSR (Optimized Link State Routing) to support QoS routing by selecting the highest-bandwidth paths between any two nodes. Their basic ideas are based on changing the way of selecting multipoint relays, depending on the bandwidth QoS constraints. Additionally, instead of using the shortest path algorithm, the maximum bandwidth spanning tree method is used. To achieve this goal, the bandwidth of each link is considered as its weight and all nodes compute trees in which the total weight of links are maximal among all possible trees.

AQOR

Ad Hoc QoS On-demand Routing (AQOR) (Xue & Ganz, 2003) is a reservation-based routing and signaling scheme. It provides end-to-end QoS support and admission control. Additionally it supports two QoS maintenance mechanisms: temporary reservation and destination-initiated recovery.

Neighborhood Maintenance

AQOR is based on the exchange of neighborhood information consisting of local topology, traffic and mobility information. This is crucial for traffic measurement, QoS violation detection and route recovery. Therefore, each node within a network periodically sends *Hello* packets with information about its traffic. Other nodes, upon receiving these packets, maintain lists of their neighbors with their corresponding traffic. If the *Hello* packets from a particular neighbor are not received during a predefined period, the connection to this neighbor is broken.

Route Discovery

Route discovery is done on-demand with the use of route request (RREQ) and route reply (RREP) packets exchanged between the source and the destination nodes. If the destination node is not within the neighborhood of the source node it broadcasts a RREQ packet. RREQ contains information about the minimum requested bandwidth and the maximum end-to-end delay. Every intermediate node does admission control on a hop-by-hop basis. If multiple routes are found during the exploration process the source node chooses the best path for its data transmission. Additionally, in order to avoid routing loops, all control packets are sequenced.

Admission Control and Temporary Reservation

During its operation AQOR takes two QoS constraints into account, namely bandwidth and end-to-end delay. In order to determine the available bandwidth the total traffic load is calculated for each node. The end-to-end delay is measured as the round-trip delay. After the route discovery phase is completed the path with the lowest end-to-end delay is selected for data transmission. If the source node, however, does not receive any answer to its RREQ during a predefined time interval it has to either backoff and initiate the route re-discovery procedure or resign from sending its flow. Finally, if the path is found, intermediate nodes make temporary reservations for the source node's data flow in order to guarantee the availability of the resources.

Route Maintenance

Route maintenance in AQOR includes the detection of the following end-to-end QoS violations:

- **Channel deterioration/congestion**. Detection is possible through one-way delay measurements. If the destination node receives a number of consecutive data packets with delay exceeding the maximum delay requirement, it triggers the QoS route recovery procedure.
- **Route breaks**. For best effort traffic detection is possible thanks to *Hello* messages. When a route break is detected, the source node is notified with an error message. Upon receiving the notification the source node can start the rerouting process. This approach is not appropriate for real-time traffic because of large delays. Additionally, in AQOR the bandwidth reservation timeout at the destination node is utilized. After the timeout is exceeded the source node starts the QoS route recovery procedure again.

In both cases, the destination node initiates the QoS route recovery procedure by sending an update message, which is treated in the same manner as a typical RREQ. When the message reaches the source node it can either immediately switch its flow to the new route or suspend it.

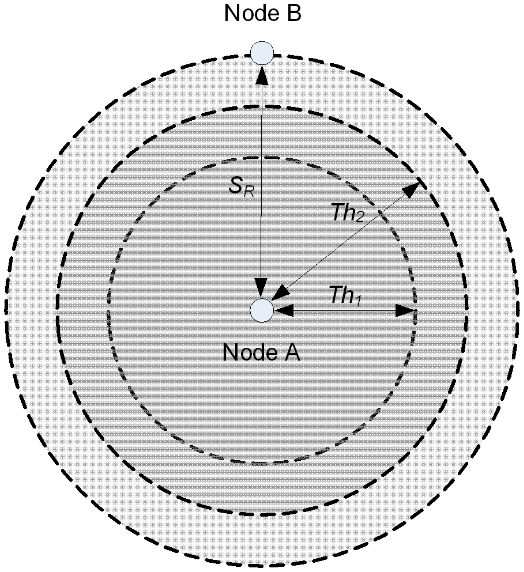
ADQR

Adaptive Dispersity QoS Routing (ADQR) (Youngki & Varshney, 2003) is a source initiated on-demand routing protocol. It uses signal strength information to predict link breaks and initiate fast data rerouting. ADQR assumes that lower layers provide information about the estimated bandwidth.

Sets and Classes

ADQR defines three signal strength levels: Th_1, Th_2, and S_R (Figure 11), where S_R is the minimum signal strength required to successfully receive a

Figure 11. ADQR thresholds (Adapted from (Youngki & Varshney, 2003))



data packet from a neighbor and R is the transmission range to this neighbor.

Furthermore, ADQR defines three sets: *node*, *link* and *route*. Each set is divided into three classes. If for a given node the received signal strength from its neighbor is higher than Th_1, the node belongs to the *first node class* of this neighbor. Correspondingly, all connections between *first node class* nodes are in the *first link class*. If the received signal strength from a neighbor is between Th_1 and Th_2, the node belongs to the *second node class* of the neighbor. The connections between these nodes are in the *second link class*. Finally, if the received signal strength is between Th_2 and S_R, the node is in the *third node class* of the neighbor and the connections between such nodes are in the *third link class*. The route classes are determined by their weakest links.

Neighbor and Routing Table

In ADQR each node keeps two tables. The neighbor table contains an updated list of the node's neighbors and their corresponding received signal strength and the minimum signal strength S_R. The routing table contains a list of all possible routes to destination nodes. Each entry includes information about the available and reserved bandwidth along the path and the list of links in the link classes.

Route Discovery

Route discovery begins when a source node broadcasts a *Route_Request* packet to its neighbors in order to find multiple disjoint paths to a destination node. *Route_Request* is then appended with the addresses of all intermediate nodes. In addition, it is updated with available bandwidth information. Upon receiving this message, the destination node updates its routing table. The destination node replies with a *Route_Reply* packet. All intermediate nodes check *Route_Reply* and update their routing tables. If the source node receives multiple routes it selects the one with the best signal strength of the links (i.e., with the best *route class*). Finally, when an appropriate route is selected, a *QoS_Reserve* packet is sent along this path (from the source to the destination node) in order to reserve the required bandwidth. Additional *QoS_ACK* packets, sent to the neighbor from which the *QoS_Reserve* packet was received, guarantee correct reservation.

Route Maintenance

In AQDR route maintenance is based on the measurement of the received signal strength. When the received signal becomes too weak, *Route_Update* packets are generated for inactive paths. They are sent to the source node by nodes which detected the problem in order to update the *link class* information. For active paths AQDR proposes a fast route maintenance scheme called *two-phase monitored rerouting*. When the signal strength of a particular link on the active route becomes lower than Th_1 the *Pre_Routing* phase is started. If the signal strength does not stop to deteriorate (i.e., it drops below Th_2) the *Rerouting* phase is started. The *Pre_Routing* phase is introduced to prepare rerouting. It is invoked in order to find an alternative route before the currently used route becomes unavailable. In addition to these procedures, the source node caches all possible routes to the destination node and, therefore, alternative routes are practically always known. Furthermore, when a particular link breaks, a *Route_Error* packet is sent to the source node, which triggers all intermediate nodes to release the reserved bandwidth. Each intermediate node replies with a *Route_Ack* packet to its previous neighbor and forwards the packet to the next neighbor. Additionally, when a node reserves network resources for its currently active paths it sends a *QoS_Update* packet to the source node of each non-active path to which it belongs. This helps intermediate nodes and the source node to keep up-to-date network resource information in their routing tables.

QS-AODV

QS-AODV (Yihai & Gulliver, 2005) is a QoS routing protocol based on AODV which creates routes according to the QoS requirements of an application. In order to improve the packet delivery ratio QS-AODV employs a local repair mechanism. Its performance is comparable to AODV under light traffic conditions and considerably better (in terms of packet delivery ratio and signaling overhead) under heavy traffic conditions. On the other hand, under heavy network load, QS-AODV has longer end-to-end delays than AODV.

QS-AODV adds additional QoS information into route request (RREQ), route reply (RREP) and route error (RERR) packets of AODV in order to create and discover routes. Information about application bandwidth requirements and session ID are added.

Route Discovery

To begin route discovery, a source node sends a RREQ packet with the QoS extension to its neighbors which perform admission control based on available bandwidth. When a destination node receives the request, after updating its routing table and reserving the required bandwidth, it can send a RREP packet to the source node. However, if the destination node has already received a similar RREQ packet, the request is buffered in case of a route reply failure. Each intermediate node, upon receiving the RREP packet, checks if it still has the required bandwidth. If the admission control fails, a RERR packet is sent to the downstream node, otherwise, the RREP packet is forwarded to the upstream node. Upon receiving RERR, each node invalidates the route entry associated with the session ID carried in this RERR, releases the reserved bandwidth and forwards the RERR packet to its next hop neighbor. Finally, when the RERR packet reaches the destination node, it can use another available route to send a new RREP packet.

Route Maintenance

To provide fast rebuilding of routes, QS-AODV introduces the idea of local repair requests. Each repair request packet includes the session ID and the required amount of bandwidth of the considered flow and has its TTL (Time To Live) value set to 3 in order to limit the broadcast area. Upon detecting a link failure to the next hop neighbor, a repair request packet is sent to a node located after this neighbor. This is possible because QS-AODV assumes that routes to nodes which are two hops away are stored in the routing table of each node. If the local repair request expires, an error procedure is invoked.

QoS-aware AODV

QoS-aware AODV (also known as Bandwidth Estimation QoS-aware Routing, BEQR) (Lei & Heinzelman, 2005) is a QoS routing protocol based on the traditional AODV routing scheme, which utilizes a cross-layer design. It can either provide feedback about the available bandwidth to the application (feedback scheme) or admit a flow with the requested bandwidth (admission scheme). The former is suitable for applications which can adjust their coding rate on the basis of the received feedback. The latter is suitable for applications which have a predefined minimal required bandwidth. Both schemes require knowledge of the available end-to-end bandwidth along the path from a source to a destination node. Bandwidth estimation is therefore a key concern of this protocol.

Bandwidth Estimation

The goal of bandwidth estimation is to find the minimal available bandwidth along the path. Two methods of bandwidth estimation are possible:

- **"Listen" bandwidth estimation**. Each node estimates the residual bandwidth by listening to the channel using physical and virtual carrier sense. This method is inaccurate when a route is broken because nodes do not know how much bandwidth is consumed by other nodes in the broken path and, therefore, do not know the amount of bandwidth released.
- **"Hello" bandwidth estimation**. Information about the sender's current bandwidth usage and its one-hop neighbors' current bandwidth usage is piggybacked onto a modified *Hello* message. The residual bandwidth is based on the information from nodes within two-hops because, typically, the interference range is twice the transmission range.

Route Discovery

QoS-aware route discovery is initiated when the source node sends a route request (RREQ) packet. After an intermediate node receives this packet, it performs admission control based on available bandwidth. For the adaptive scheme, the intermediate node compares its residual bandwidth with the minimum bandwidth specified by the RREQ packet. The packet is forwarded if the residual bandwidth is greater than the minimum bandwidth. Otherwise, the RREQ is updated with the residual bandwidth. Later, after receiving the RREQ packet, the destination node does the same checking procedure. After the procedure is completed an additional checking procedure is invoked in which the upper bound of the minimum available bandwidth is re-estimated. In the end, the destination node sends the route reply (RREP) packet with an updated minimum bandwidth value to the source node. Upon receiving RREP, all intermediate nodes enable the path and update their routing tables with the new value of the minimum available bandwidth.

Route Maintenance

AODV detects a broken path by monitoring *Hello* packets. If a node stops receiving *Hello* packets from its neighbor, it sends an error message to its upstream neighbors. Upon receiving the error message, only the source node reinitiates a routing discovery procedure. This scheme works correctly for the *Listen* bandwidth estimation method. However, it has to be modified for the *Hello* bandwidth estimation method. In this scheme the neighboring nodes' caches are not updated in a timely fashion and, therefore, there is a high probability that the bandwidth used by the broken path will not be released before a new RREQ will arrive. As a remedy, QoS-aware AODV introduces *Immediate Hello* messages which have the same function as the modified *Hello* packets but they are sent immediately after detecting each broken link to allow faster cache updates.

QMRPCAH

Layuan and Chunlin (2007) propose a QoS multicast routing protocol for clustering mobile ad-hoc networks (QMRPCAH). The protocol establishes paths based on QoS constraints in a scalable way and reduces the signaling overhead. In QMRPCAH each node maintains local multicast routing information and/or summary information of other clusters. Additionally, the protocol supports mobility of nodes, i.e., each node being a member of a multicast group can join and leave it dynamically. Finally, the protocol supports several QoS metrics although mainly delay and bandwidth are considered. The protocol does not provide hard QoS guarantees.

Clustering

Figure 12 shows an exemplary clustering of an ad-hoc network. All nodes within the network are divided into clusters (domains) of different levels. A 1st-level cluster consists of nodes with similar mobility characteristics. When several 1st-level clusters are combined a 2nd-level cluster is created etc. Clusters of different levels do not overlap with each other. Nodes within the 1st-level clusters are called local nodes. Nodes lying within the transmission range of one or more clusters are called bridge nodes. Each cluster contains also a single cluster head. The cluster head is a coordinator which decides on channel assignments, performs power control, maintains time division frame synchronization, and deals with the spatial reuse of bandwidth.

Mobility Support

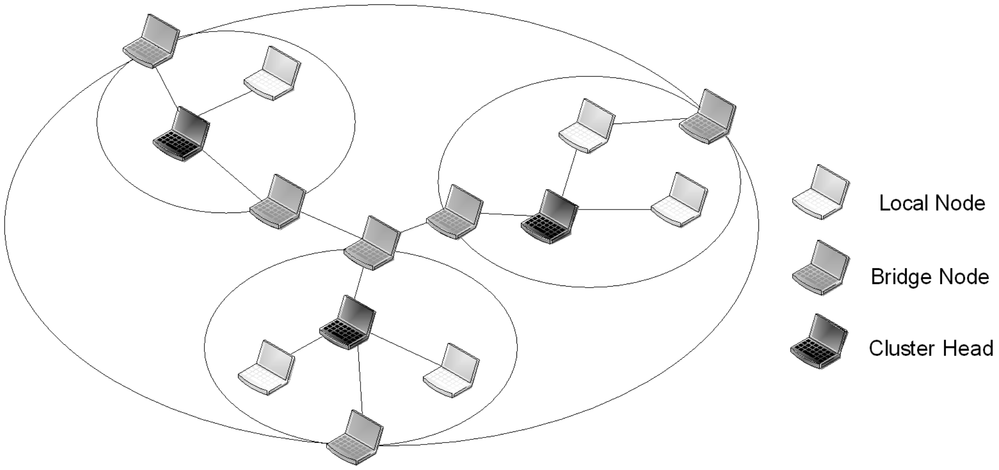
QMRPCAH assumes that each local node periodically measures the delay of its outgoing links and broadcasts the gathered information to all nodes within its cluster in the form of update messages. Upon receiving an update message the local nodes update their intra-cluster routing information. Similarly, each bridge node performs the same procedure and periodically sends update messages to other bridge nodes. On this basis the inter-cluster information becomes updated. Additionally, in order to handle mobility, each node joining a new cluster must subscribe to its multicast tree to become a local multicast node of the new domain.

QMRPCAH uses a receiver-initiated flooding algorithm. The algorithm assures that for all joining nodes only the paths which satisfy bandwidth requirements will be included in the multicast tree. Additionally, QMRPCAH can guarantee that all messages which have a larger path delay will arrive after the messages which have a smaller path delay. Flooding messages are forwarded only if all QoS constraints are met.

TORA-SHORT

Asokan, Natarajan, and Venkatesh (2008) integrate TORA with SHORT (Self-Healing and Optimized Routing Techniques) in order to improve QoS routing. This is done by monitoring routing paths and, if a shortcut route is available, redirecting the path. The performance of TORA-SHORT is based on the exchange of query and update packets. The

Figure 12. Exemplary clustering of an ad-hoc network



query packets are generated by source nodes in order to begin the route discovery process. The update packets are generated by destination nodes in order to terminate the query packets and by intermediate nodes to inform about route breaks.

Two classes of SHORT are proposed: path aware (PA)-SHORT and energy aware (EA)-SHORT. The former optimizes hop count, whereas, the latter conserves power. By incorporating (PA)-SHORT and (EA)-SHORT in TORA, QoS routing can be optimized by means of path lengths and the residual energy level of nodes. It assures that both the shortest paths and the nodes with sufficient available power are chosen during the route selection process. In order to meet this goal, TORA-SHORT makes use of two tables: the hop comparison table and the overhear table. Additionally, it is defined that each new entry has three obligatory fields, i.e., hop counter, residual energy level of a transmitting node, and the sender's address.

CROSS LAYER QOS ARCHITECTURES

Providing QoS in mobile ad-hoc networks is a challenging task because of the many factors involved. The previous parts have shown solutions to problems apparent at separate layers of the OSI/ISO model. However, these solutions usually aim to solve only a single problem. In order to provide end-to-end QoS in multi-hop ad-hoc networks a cross-layer approach is needed. Such a framework needs to integrate the singular solutions into a complete approach for QoS provisioning. The advantage of using this cooperative approach is being able to share relevant information between layers and provide feedback between components of the architecture. This leads to a more responsive, scalable, and flexible system and adaptability is required because MANETs are dynamic environments. There is one disadvantage of using a cross-layer approach. Namely, unintended interactions between components can occur (e.g., in the form of feedback loops) which makes locating problems troublesome.

The cross-layer approach offers a wide range of design possibilities. However, most of the solutions that are presented in this part have been influenced by each other and there are many similarities among them. Especially the first two approaches, INSIGNIA and SWAN, have been very influential. Table 1 presents the important building blocks of a cross-layer architecture.

Table 1 Building blocks of a cross-layer architecture

ISO/OSI Layer	Building blocks
PHY	channel monitoring (rate, SNR, BER), dynamic rate control
MAC	bandwidth estimation, priority queuing, traffic differentiation
Network	QoS routing, QoS signaling (resource reservation), traffic classification, traffic shaping, admission control

Most of them are present in all the mentioned solutions.

FQMM

Flexible Quality of service Model for Mobile ad-hoc networks (FQMM) (Xiao et al., 2000) is a QoS model developed for MANETs. It represents an interesting and unique approach to the problem of QoS provisioning. It is not associated with QoS negotiation procedures, however, it can be adopted to enhance existing solutions.

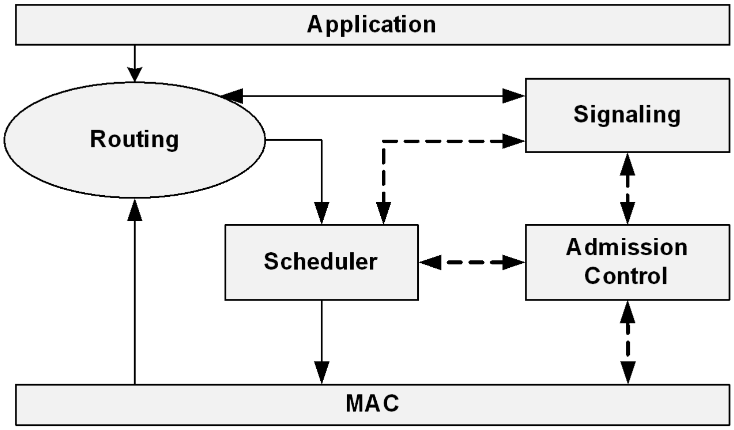
FQMM is a hybrid solution combining per-flow and per-class service provisioning. Therefore, it can be treated as a combination of the IntServ and DiffServ models. High priority traffic (which is a small percentage of overall traffic) is given per-flow provisioning. Traffic of other priorities is given per-class provisioning. FQMM defines three types of nodes (based on DiffServ): ingress, interior, and egress. The sender is an ingress node while the destination is an egress node. Ingress nodes perform classification, marking, policing and shaping. Interior nodes forward traffic of others and perform traffic shaping according to traffic profiles. The goal of the traffic profiles is to keep consistent differentiation between sessions. A profile is defined as the relative percentage of the effective link capacity, in order to keep the differentiation between classes predictable and consistent under different network dynamics.

INSIGNIA

INSIGNIA (Lee et al., 2000) is a QoS framework which operates mostly at the IP layer. Its main design consideration is the support of adaptive multimedia services. INSIGNIA aims to provide *base* QoS or *enhanced* QoS depending on available resources. It is based on in-band signaling and soft-state reservations. Its features include resource reservation, restoration control, and session adaptation between communicating nodes. Figure 13 presents an overview of the INSIGNIA architecture.

Admission control in INSIGNIA allocates bandwidth to flows based on information provided by the service (i.e., minimum and maximum bandwidth). This allows providing either *base* or *enhanced* QoS, respectively. The decision of admitting a flow to the network is based on the requested bandwidth, the measured channel capacity and current channel utilization. This ensures that new reservation requests do not impact existing reservations. Admission control is done on a hop-by-hop basis. Each intermediate node, upon receiving a reservation packet, accepts or denies the request. After a positive decision, the node maintains per-flow soft-state reservations and subsequent packets are scheduled accordingly. Reservations are maintained for the duration of the packet flow. If packets do not arrive before a certain timeout, the reserved resources are released. Packets of denied reservations are treated as best-effort. After receiving a reservation packet, the destination node sends a QoS report to the source to complete the reservation phase. Such reports are also sent at time intervals (specified by applications) and whenever requested. Furthermore, packets are scheduled to be sent to the network using a weighted round-robin scheduling discipline.

Figure 13. The INSIGNIA architecture (Adapted from (Lee et al., 2000))



INSIGNIA uses in-band signaling for establishing, adapting, restoring, and tearing down end-to-end QoS sessions. The IP option field of the IP packet header is used to deliver the signaling protocol commands. Packets are sent either in reservation mode or as best-effort.

INSIGNIA aims to be very flexible to varying network conditions. Whenever the available bandwidth changes adaptation algorithms are invoked. Flow restoration algorithms are used to respond to dynamic route changes. Once the routing protocol updates the routing table, admission control and resource reservation is performed on the new paths. Flows may have immediate restoration, or partial/permanent degradation of QoS depending on available resources.

In this QoS framework neither the routing nor the MAC protocol are defined. However, INSIGNIA can be used with already existing protocols. There exists a combination of INSIGNIA and TORA known as INORA (Dharmaraju & Roy-Chowdhury, 2002). The TORA routing protocol provides multiple paths (between sender and destination) to the signaling protocol, and the latter checks if they meet the necessary QoS requirements.

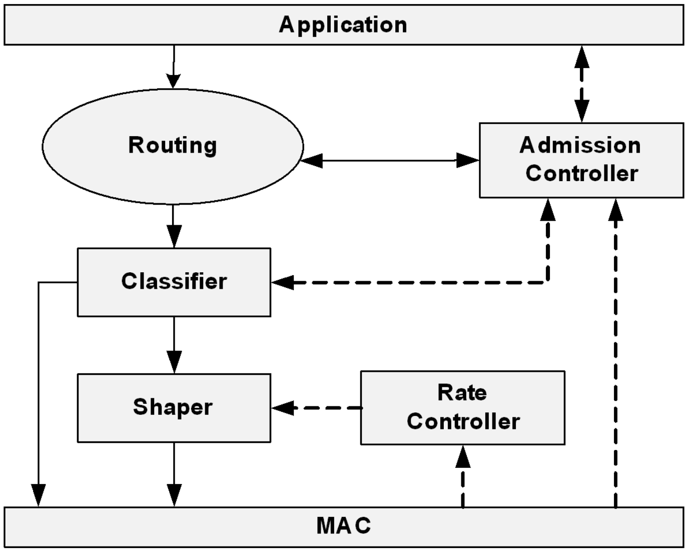
SWAN

SWAN (Stateless Wireless Ad-hoc Networks) (Ahn et al., 2002) is a distributed, cross-layer QoS framework for MANETs. It provides service differentiation, QoS negotiation, admission control, and dynamic regulation in case of congestion. Two traffic classes are considered: real-time and best-effort traffic. Figure 14 presents an overview of the SWAN architecture.

The SWAN model consists of a number of mechanisms present in every node. The classifier differentiates traffic between the two classes and marks packets accordingly. Differentiation is achieved because real-time packets go directly to the MAC layer, while best-effort packets are sent to the shaper. The shaper is a leaky bucket which delays packets accordingly to the rate calculated by the rate controller. The operation of the rate controller is based on delay measurements in the MAC layer.

SWAN is stateless because intermediate nodes do not keep any per-flow or aggregate state information. It is only the source node which performs admission control and, therefore, efficient estimation of bandwidth availability is required. This is provided by the admission controller which sends request/response probes to the destination to determine the bottleneck bandwidth along the

Figure 14. The SWAN architecture (Adapted from (Ahn et al., 2002))



path. Based on the received information an admission decision is made. If a session is admitted, its packets are marked by the classifier as real-time packets; otherwise they are treated as best-effort packets.

Dynamic regulation of real-time sessions is needed when congestion conditions occur. This can be caused by node mobility and dynamic rerouting. When a mobile node detects a violation of the real-time traffic utilization limits it begins marking ECN (Explicit Congestion Notification) bits in the IP header of real-time packets. As a result, the destination node, which monitors incoming packets, can send a regulate message to inform the source node about the congestion. The source node should then re-establish the real-time session by sending a new probing request to the destination. The result of this re-establishment is either achieving the previous QoS or the session being dropped. Bandwidth adaptation of real-time sessions is not considered.

2LQoS

2LQoS (Two-Layered Quality of Service) (Nikaen et al., 2002) is a cross-layer QoS routing scheme with traffic differentiation and shaping. The network and application layers cooperate to determine the most suitable path through the network. Metrics from both layers are used. Path discovery is based on network layer metrics, such as: hop count, power level, buffer level, and stability level. The first metric is related to resource consumption, the second and third to load balancing, while the final one is a measure of mobility. Path selection is done according to application layer metrics: delay, throughput and cost (which is a function of power and buffer level of a node). Traffic is categorized into three classes. The first provides low delay (for voice applications), the second – high throughput (for video applications) and the last one has no constraints (best effort). Traffic can be shaped to meet QoS conditions in the network. This scheme does not perform any resource reservation. Service differentiation is done at each ad-hoc node through scheduling. Packets from each class are assigned to their appropriate queues, with each queue having a different, user defined weight. Packet classification is done at the source node.

DS-SWAN

DS-SWAN (Differentiated Services-Stateless Wireless Ad Hoc Networks) (Domingo & Remondo, 2004) provides end-to-end QoS in ad-hoc networks which are connected to a fixed IP network. DS-SWAN utilizes SWAN in the ad-hoc network and DiffServ in the infrastructure network. Additionally, it allows both these mechanisms to cooperate. The parameters of SWAN change dynamically according to the conditions in both the ad-hoc network and the infrastructure network. When packet delays exceed a given threshold special messages (*QoS Lost*) are sent. These messages cause more aggressive shaping of best effort traffic. The authors also propose a new routing protocol, SD-AODV (Service Differentiation AODV). It is aware of the *QoS Lost* messages and is able to re-route new flows away from congested zones.

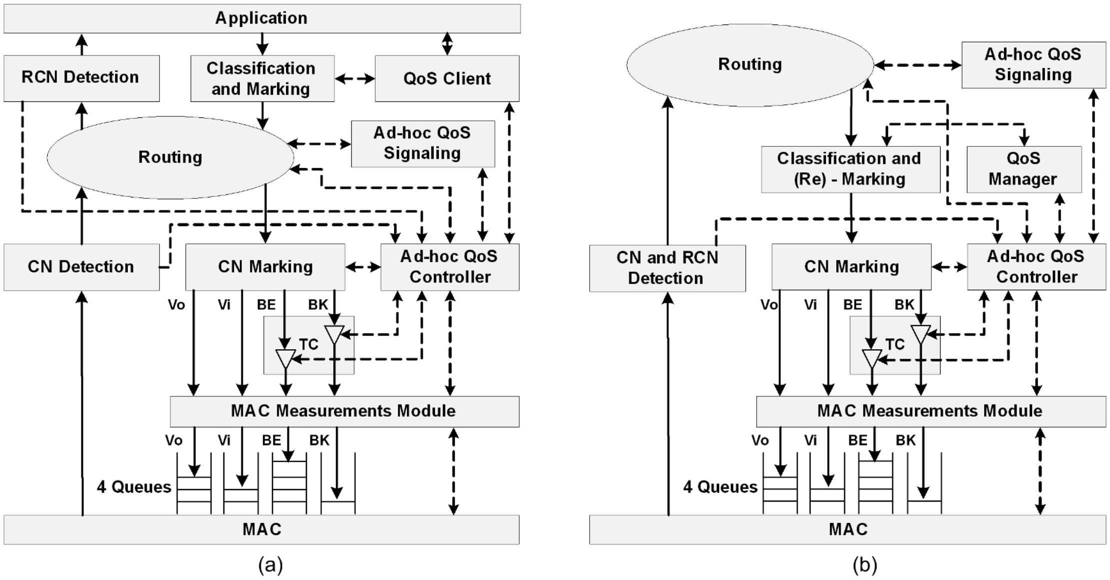
The DAIDALOS Approach

Another QoS architecture was developed within the DAIDALOS I research project (Crisostomo et al., 2005). It was further developed in the DAIDALOS II project (Natkaniec, Gozdecki, & Sargento, 2007). The goal of this architecture is the integration of ad-hoc networks with infrastructure networks, a very useful scenario in hotspot environments. It supports the following set of features. It is a cross-layer solution providing stateless QoS to MANETs integrated with infrastructure. The architecture supports MAC layer traffic differentiation in the four IEEE 802.11 EDCA access categories. The implemented end-to-end signaling is not only simple but also allows resource reservations to originate from either the ad-hoc or the infrastructure part. The ad-hoc signaling mechanism is integrated with the Next Steps in Signaling (NSIS) protocol suite (Hancock et al., 2005) in the infrastructure network. NSIS provides flexibility in end-to-end signaling, supporting different resource management models and bidirectional reservations. Additionally, the architecture provides MAC layer measurements, traffic shaping, dynamic regulation, and admission control. Finally, it features the first real-life implementation of EDCA in a MANET (Natkaniec et al., 2009).

The ad-hoc part of the proposed QoS architecture consists of two main logical units, the mobile node (MN) and the gateway (GW) (Figure 15). MN has a double role acting both as a user terminal which provides QoS support to the end user, and as a router which forwards the traffic of neighboring nodes. The physical interconnection between the MANET and the infrastructure planes occurs in GW. The GW provides connectivity with infrastructure, participates in admission control and dynamic regulation. It is also responsible for the interaction of QoS signaling. Separate models exist for MN and for GW. End-to-end QoS resource management is supported through the interoperation between the MN and the GW (Figure 15). More specifically, this occurs through the interaction between the QoS Client (QoSC) in the MN and QoS Manager (QoSM) in the GW. QoSC retrieves the necessary QoS parameters from applications (through an API) and maps them to network QoS parameters. QoSC also performs per-flow end-to-end QoS signaling, controls the Classification and Marking module (which marks the Traffic Class and Flow Label fields in IPv6 headers), notifies the applications of the state of network interfaces, and synchronizes network resource reservation. QoSM provides the same functionalities as QoSC but without the interface to the application layer.

The other modules have the following functionalities. The Ad-hoc QoS Controller (AHQoSC) coordinates the work of the other modules, collects information about available resources in the wireless medium, and performs admission control in

Figure 15. The DAIDALOS model for the mobile node (a) and gateway (b) (Adapted from (Natkaniec, Gozdecki, & Sargento, 2007))



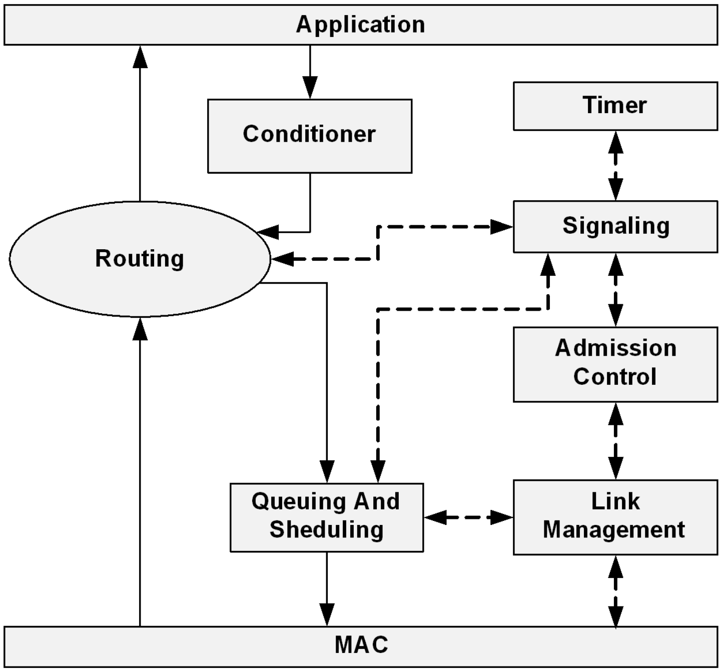
the ad-hoc path. It is also responsible for traffic control, reaction to congestion, and participating in resource management. The MAC Measurements Module (MMM) provides AHQoSC with information regarding bandwidth utilization, transmission delay, current transmission rate, frame statistics and idle intervals. The Ad-hoc QoS Signaling (AHQoSSig) module is responsible for QoS negotiations, probing for available bandwidth (like in SWAN), and session setup between infrastructure and ad-hoc. Traffic differentiation is performed by the IEEE 802.11 EDCA function and the Traffic Controller (TC) module, which is used to shape lower-priority flows. The architecture is aware of overload situations through the Congestion Notification (CN) signaling mechanism, which is implemented in three modules: CN Marking (CNM), CN Detection (CND), and Receiver CN Detection (RCND).

The GW is able to support the same functionalities as the MN, but does not interact with applications (since it works only at the IP layer and below) (Figure 15b). Additionally, it collects, generates and processes QoS signaling messages in the ad-hoc and infrastructure part. It is also responsible for the enforcement of QoS in the infrastructure network.

HQMM

HQMM (Hybrid QoS Model for Mobile Ad-hoc Networks) (He & Abdel Wahab, 2006) combines the IntServ and DiffServ models for ad-hoc networks. In this approach, per-flow provisioning (based on INSIGNIA) is given to traffic of the highest priority. Other priorities receive per-class provisioning (based on DiffServ). Differentiation is done based on bandwidth. HQMM defines three types of nodes (as DiffServ): ingress, interior and egress. Since each node can perform any of these functions, even simultaneously, they all have the same architecture (Figure 16). Traffic is DSCP marked in the conditioner, according to a pre-established agreement (which is out of scope of HQMM). The choice of routing protocol is also beyond the scope of this scheme. The signaling module provides the capabilities of INSIGNIA signaling. The link management module monitors the

Figure 16. The HQMM architecture (Adapted from (He & Abdel Wahab, 2006))



wireless channel and reports current and average bandwidth availability. Service differentiation is achieved through scheduling, queue management algorithms, and packet classification. In particular, INSIGNIA reservation packets are treated with the highest priority, while other packets are put into DiffServ classes.

FQM

FQM (Framework for QoS Multicast) (Saghir, Wan, & Budiarto, 2006) aims to support QoS for multicast applications in MANETs. Multiple paths meeting QoS requirements are found through a new on-demand routing protocol. FQM implements features from both the IntServ and the DiffServ models. For every accepted QoS route request, IntServ is used. Data packets from other sources are given DiffServ. Bandwidth is therefore divided between fixed-bandwidth for flows that have been admitted, and shared-bandwidth for other data. Depending on resource availability, either redundant or single paths are used. Route requests are intelligently flooded through the network. Cross-layer bandwidth estimation is used for admission control. This estimation is performed at each node by passively listening to the channel. The route request reaches the destination if there is enough bandwidth on the path. In such a case, a route reply message originates at the destination and traverses intermediate nodes on the path. This constructs the multicast tree. Service differentiation in FQM is achieved with the use of a classifier, shaper, dynamic rate control, and priority queues. Packets are mapped into two traffic classes: real-time and best effort. The mapping is based on the Type of Service (ToS) field in the IP header. Shaping is applied only to best effort traffic. The rate of best effort traffic is changed dynamically with an additive increase multiplicative decrease algorithm and, additionally, a drop tail algorithm is used in the queues.

CLQM

Sarma and Nandi (2008) propose a Cross-layer QoS Mapping (CLQM) framework for MANETs. It works on three layers: application, network and data link. Four service classes are considered. At the application layer the QoS classes are characterized by maximum end-to-end delay, minimum throughput or as best-effort. These metrics are mapped to network layer metrics which are in turn mapped to data link layer metrics. The QoS metrics at the network layer are path bandwidth, path delay, path stability and hop count. The data link layer metrics are MAC delay, link bandwidth and link stability. They are used in the process of admission control.

Admission control is performed during route discovery, based on available throughput. Modified AODV packets (RREQ and RREP) carry network layer QoS metrics. RREQ packets are dropped if their QoS requirements are not met. Network monitoring is periodically done at both the data link and network layers. At the data link layer, in order to adapt to network conditions, the framework dynamically adjusts the values of $CW_{\min}$ and $CW_{\max}$ for each class with the use of the CW adaptor module. Bandwidth estimation is performed at each node by measuring network utilization, i.e., the fraction of the time the channel is busy to the total measured time. MAC delay is calculated as the time from when a packet arrives at the MAC layer to the time an acknowledgement is received. An exponentially weighted moving average is used to estimate the average delay. At the network layer, the RSQR (Route Stability based QoS Routing) protocol is used to choose the best path for each class. Service differentiation among the QoS classes is achieved by a class-based scheduler. Congestions in the network detected using the ECN mechanism. They are resolved through flow rerouting or termination. The architecture of CLQM is presented in Figure 17.

Summary

In this part we have presented multiple cross-layer solutions to the problem of QoS provisioning in ad-hoc networks. Even though the problem is complex, most solutions are based on a mixture of similar features, such as bandwidth estimation, traffic differentiation, QoS routing, resource reservation, traffic shaping and admission control. Table 2 gives a comparison of the presented solutions and the features they support.

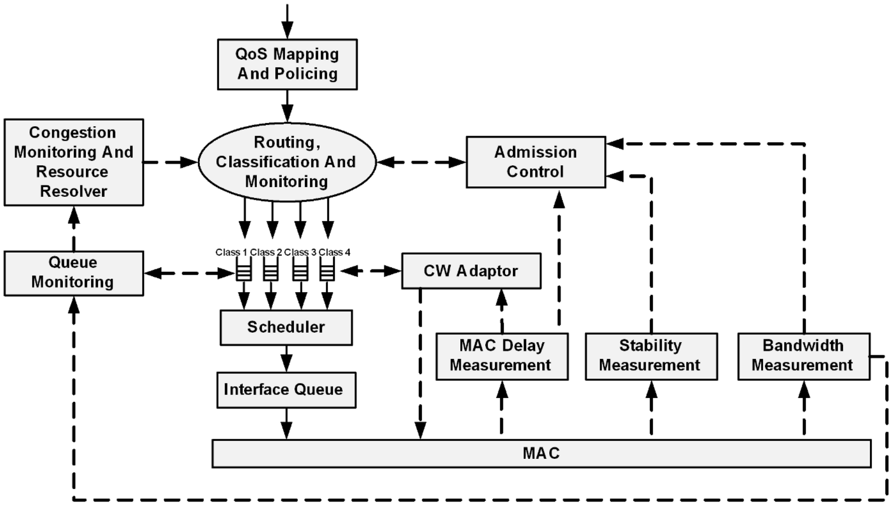
FUTURE RESEARCH DIRECTIONS

Even though there are a number of protocols and solutions for QoS-aware ad-hoc networks, further exploration in this field is required. There are several unsolved challenges that need to be addressed when providing QoS support in future multi-hop ad-hoc networks. Some of the possible research directions are shortly discussed next. We look at the three lower OSI/ISO layers and at cross-layer challenges.

The physical layer is responsible for effective bit transmission. Even when the BER of the radio channel is high the physical layer parameters, e.g., modulation and channel coding, need to be correctly selected. Therefore, efficient multi-rate algorithms are required. They should interact with the MAC sub-layer and schedule the transmitted traffic classes to avoid bad channel conditions.

A number of QoS-aware MAC schemes have been proposed, however, most of them have drawbacks. They focus on specific QoS features and assume a simple network topology, one traffic priority per node, limited node mobility, ideal channel conditions, etc. In a multi-hop ad-hoc network, nodes usually forward traffic which belongs to different flows, and support many traffic classes with different delay bounds and bandwidth requirements. A MAC protocol for multi-hop ad-hoc networks should therefore optimize the trade-off between fairness, efficient

Figure 17. The CLQM framework architecture (Adapted from (Sarma & Nandi, 2008))



resource utilization, support of multiple traffic classes, strict priority guarantees, bandwidth on demand allocation, traffic scheduling, and fast reaction to transmission errors. It seems that close interactions between the MAC protocol, and the physical and network layers are needed to achieve all these requirements.

The designers of new network layer QoS mechanisms must take into account several requirements. Firstly, the accuracy of QoS routing protocols must be maximized. Secondly, control overhead should be kept as small as possible. Thirdly, because intermediate nodes are forwarders of traffic they may limit available resources and, furthermore, their unreliability may lead to route breaks. Another concern is that currently not all QoS routing protocols estimate the available network resources. Some of them assume that the network capacity is known *a priori*. As a result, all schemes based on bandwidth and delay estimation (e.g., admission control, location prediction) are not done appropriately. Route maintenance is also a challenging task due to the frequently changing topology of MANETs. The best way to deal with this problem would be to have mechanisms which could predict possible route breaks, find redundant routes, and perform re-computation of broken routes. However, each of these techniques increases overhead and, therefore, it is crucial to find a trade-off between full mobility support and required routing overhead. Additionally, it has to be decided if global or local state information has to be obtained to perform QoS routing. The former estimates network resources more accurately, but the latter requires lower control overhead. Furthermore, it has not been decided yet if reactive, proactive or hybrid routing protocols are the best way to find feasible paths. Finally, an appropriate resource reservation scheme must be decided on as well.

Because a cross-layer solution seems to be the most promising approach to provide end-to-end QoS provisioning in multi-hop ad-hoc networks, the interaction between protocols at different layers is crucial. Even though QoS frameworks have been proposed, more advanced solutions should be investigated. Cross layer design should obviously integrate QoS solutions from the physical, data link and network layers. In particular, most current QoS schemes lack dynamic PHY rate control, accurate channel measurements, a QoS-aware MAC, restoration control, session adaptation

Table 2. Comparison of the presented cross-layer solutions

	Channel Monitoring (rate, SNR, BER)	**MAC Measurements**	**QoS MAC (e.g., EDCA)**	**Scheduler (Priority Queuing)**	**Number of Traffic Classes**	**QoS Routing**	**QoS Signaling (Resource Reservation)**	**Traffic Shaping**	**Admission Control**	**Restoration Control**	**Session Adaptation / Dynamic Regulation (Like ECN)**	**Soft/Hard QoS Guarantees**	**Complies with 802.11**	**Per-session or Per-class**	**Verification (Implementation or Simulation)**	**Integration with Infrastructure**
2LQoS	-	-	-	X	3	X	-	X	-	-	-	S	X	C	-	-
CLQM	-	X	-	X	4	X	-	-	X	-	X	S	X	C	-	-
Daidalos	X	X	X	-	4	-	X	X	X	-	X	S	X	C	I	X
FQM	-	X	-	X	2	X	-	X	X	-	-	S	X	C	S	-
FQMM	-	X	-	X	2	-	-	X	X	-	-	B	X	Both	S	-
HQMM	-	X	-	X	5	-	X	X	X	X	X	S	X	Both	S	-
INSIGNIA	X	X	-	X	-	-	X	-	X	X	X	S	X	S	S	-
SWAN (DS-SWAN)	-	X	-	-	2	-	-	X	X	-	X	S	X	C	S/I	X

and integration with infrastructure. Additionally, QoS frameworks are difficult to implement and verify because of their complex architectures. Many architectures can be proposed, but until they are tested by users they are only theoretical accomplishments. Furthermore, it is up to standard organizations and equipment vendors to attempt to deploy reliable QoS solutions in multi-hop ad-hoc networks.

CONCLUSION

In this chapter we have discussed QoS issues for multi-hop ad-hoc networks. Supporting appropriate QoS for these networks is a very complex problem and has become an active area of research in recent years. Several QoS protocols designed for the physical, data link and network layers have been shortly presented. The analysis of single layer QoS solutions shows that they are unsuitable for multi-hop ad-hoc networks, where complex QoS mechanisms are needed. We have also described and compared the most interesting cross-layer QoS solutions. It is clear that these approaches should provide a general model, which can be dynamically tuned to support applications with different QoS requirements. We found that there are a number of unsolved challenges that need to be addressed to design complete QoS-aware solutions for multi-hop ad-hoc networks. Therefore, there are many research possibilities in this field of engineering. Solving the existing

QoS issues will allow future ad-hoc networks to meet user expectations.

REFERENCES

Ahn, G. S., Campbell, A., Veres, A., & Sun, L. H. (2002). Supporting Service Differentiation for Real-Time and Best-Effort Traffic in Stateless Wireless Ad Hoc Networks (SWAN). *IEEE Transactions on Mobile Computing*, *1*(3), 192–207. doi:10.1109/TMC.2002.1081755

Asokan, R., Natarajan, A. M., & Venkatesh, C. (2008). Quality-of-service routing using path and power aware techniques in mobile ad hoc networks. *Journal of Computer Systems, Networks, and Communications*, *2008*(1).

Awerbuch, B., Holmer, D., & Rubens, H. (2003). High throughput route selection in multi-rate ad hoc wireless networks. Tech. Rep., Johns Hopkins University, Computer Science Dept., Baltimore, MD.

Bicket, J. (2005). *Bit-rate Selection in Wireless Networks*. Unpublished master's thesis, Massachusetts Institute of Technology, Cambridge, MA.

Chaudet, C., & Guérin Lassous, I. (2001). BRuIT: Bandwidth Reservation under Interferences Influence. (Research Rep. No. 4308), INRIA.

Chenxi, Z., & Corson, M. S. (2002). QoS routing for mobile ad hoc networks. In *Proceedings of Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies, IEEE INFOCOM 2002* (pp. 958-967). New York: IEEE Inc.

Crisostomo, S., Sargento, S., Natkaniec, M., & Vicari, N. (2005). A QoS architecture integrating mobile Ad-Hoc and infrastructure networks. In *Proceedings of the ACS/IEEE 2005 international Conference on Computer Systems and Applications* (pp. 897-903). Washington, DC: IEEE Computer Society.

De, S., Das, S. K., Wu, H., & Qiao, C. (2002). A resource efficient RT-QoS routing protocol for mobile ad hoc networks. In *The 5th International Symposium on Wireless Personal Multimedia Communications,* (pp. 257-261).

Deng, D. J., & Chang, R. S. (1999). A priority scheme for IEEE 802.11 DCF access method. *IEICE Transactions on Communications*. *E (Norwalk, Conn.)*, *82-B*(1), 96–102.

Dharmaraju, D., & Roy-Chowdhury, A. (2002). INORA - A Unified Signaling and Routing Mechanism for QoS Support in Mobile Ad hoc Networks. In *Proceedings of the 2002 International Conference on Parallel Processing Workshops* (pp. 86-93). Washington, DC: IEEE Computer Society.

Domingo, M. C., & Remondo, D. (2004). A cooperation model between ad hoc networks and fixed networks for service differentiation. In *29th Annual IEEE International Conference on Local Computer Networks,* (pp. 692-693). New York: IEEE Inc.

Ferrari, G., Tonguz, O. K., & Bhatt, M. (2004). Impact of receiver sensitivity on the performance of sensor networks. In IEEE Vehicular Tech. Conf. VTC Spring, Milan, Italy.

Gerasimov, I., & Simon, R. (2002). Performance analysis for ad hoc QoS routing protocols. In International Mobility and Wireless Access Workshop, MobiWac'02, (pp. 87-94).

Głowacz, A., Natkaniec, M., Kosek, K., & Szott, S. (2008). Guarantying QoS in MANETs using MAC Layer Measurements. In 17th ICT Mobile & Wireless Communications Summit. Dublin: IIMC International Information Management Corporation.

Hancock, R., Karagiannis, G., Loughney, J., & Van den Bosch, S. (2005). *Next Steps in Signaling (NSIS): Framework*. RFC 4080.

He, Y., & Abdel Wahab, H. (2006). HQMM: A Hybrid QoS Model for Mobile Ad-hoc Networks. *11th IEEE Symposium on Computers and Communications, ISCC '06*, (pp. 194-200). New York: IEEE Inc.

Holland, G., Vaidya, N. H., & Bahl, P. (2001). A rate-adaptive MAC protocol for multi-hop wireless networks. *International Conference on Mobile Computing and Networking* (pp. 236-251). New York: ACM.

IEEE. 802.11 Standard for Wireless LAN: Medium Access Control (MAC) and Physical Layer (PHY) Specification. (2007). New York: IEEE Inc.

Kamerman, A., & Monteban, L. (1997). WaveLAN-II: A high-performance wireless LAN for the unlicensed band. *Bell Labs Technical Journal*, *2*(3), 118–133. doi:10.1002/bltj.2069

Lacage, M., Manshaei, M. H., & Turletti, T. (2004). IEEE 802.11 Rate Adaptation: A Practical Approach (Research Rep. No. 5208), INRIA.

Layuan, L., & Chunlin, L. (2007). A QoS multicast routing protocol for clustering mobile ad hoc networks. *Computer Communications*, *30*(7), 1641–1654. doi:10.1016/j.comcom.2007.01.017

Lee, S., Ahn, G. S., Zhang, X., & Campbell, A. T. (2000). INSIGNIA: An IP-Based Quality of Service Framework for Mobile Ad Hoc Networks. *Journal of Parallel and Distributed Computing, Special issue on Wireless and Mobile Computing and Communications, 60*, 374-406.

Lei, C., & Heinzelman, W. B. (2005). QoS-aware routing based on bandwidth estimation for mobile ad hoc networks. *IEEE Journal on Selected Areas in Communications*, *23*(3), 561–572. doi:10.1109/JSAC.2004.842560

Lin, C. R., & Gerla, M. (1997). Asynchronous multimedia multihop wireless network. In *Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, INFOCOM '97 (pp. 118-125). New York: IEEE Inc.

Lin, C. R., & Liu, J. S. (1999). QoS routing in ad hoc wireless networks. *Journal on Selected Areas in Communications*, *17*(8), 1426–1438. doi:10.1109/49.779924

Natkaniec, M., Gozdecki, J., & Sargento, S. (2007). A QoS Architecture Integrating Ad-Hoc and Infrastructure in Next Generation Networks. In B. Simak, R. Bestak, & E. Kozowska (Eds.), IFIP International Federation for Information Processing, Personal Wireless Communications (pp. 25–37). New York: Springer.

Natkaniec, M., Kosek-Szott, K., Szott, S., Gozdecki, J., Głowacz, A., & Sargento, S. (2009). Supporting QoS in Integrated Ad-Hoc Networks. *Wireless Personal Communications*, Springer, ISSN 1572-834X (Online).

Natkaniec, M., & Pach, A. R. (2000). An Analysis of Modified Backoff Mechanism in IEEE 802.11 Networks. *Polish-German Teletraffic Symposium* (pp. 89-96).

Natkaniec, M., & Pach, A. R. (2002). PUMA - a new channel access protocol for wireless LANs. *International Symposium on Wireless Personal Multimedia Communications*, (vol. 3, pp. 1351-1355).

Nikaen, N., Bonnet, C., Moret, Y., & Rai, I. A. (2002). 2LQoS - Two-layered QoS model for reactive routing protocols for mobile ad-hoc networks. In *Proceedings of SCI - 6th World Multiconference on Systemics, Cybernetics and Informatics*.

Pal, A., Dogan, A., & Ozguner, F. (2002). MAC Layer Protocols for Real-time Traffic in Ad hoc Wireless Networks. *IEEE International Conference on Parallel Processing* (pp. 539-546). New York: IEEE Inc.

Sadeghi, B., Kanodia, V., Sabharwal, V., & Knightly, E. (2002). Opportunistic media access for multirate ad hoc networks. *International Conference on Mobile Computing and Networking* (pp. 24-35). New York: ACM

Saghir, M., Wan, T. C., & Budiarto, R. (2006). A new cross-layer framework for QoS multicast applications in mobile ad hoc networks. *International Journal of Computer Science and Network Security*, *6*(10), 142–151.

Sarma, N., & Nandi, S. (2008). A priority based QoS-Aware MAC protocol (PQAMP) in mobile ad hoc networks. 4th ACM symposium on QoS and security for wireless and mobile networks, Q2SWinet (pp. 79-82), New York: ACM.

Shah, S. H., & Nahrstedt, K. (2002). Predictive location-based QoS routing in mobile ad hoc networks. *IEEE International Conference on Communications, ICC'02* (pp. 1022-1027). New York: IEEE Inc.

Shigang, C., & Nahrstedt, K. (1999). Distributed quality-of-service routing in ad hoc networks. *IEEE Journal on Selected Areas in Communications*, *17*(8), 1488–1505. doi:10.1109/49.780354

Sinha, P., Sivakumar, R., & Bharghavan, V. (1999). CEDAR: a core-extraction distributed ad hoc routing algorithm. In *Proceedings of Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM '99,* (pp. 202-209). New York: IEEE Inc.

Sobrinho, J. L., & Krishnakumar, A. S. (1999). Quality-of-service in ad hoc carrier sense multiple access wireless networks. *IEEE Journal on Selected Areas in Communications*, *17*(8), 1353–1368. doi:10.1109/49.779919

Van der Vegt, A. J. (2002). Auto Rate Fallback Algorithm for the IEEE 802.11a Standard, (Tech. Rep.), Utrecht University.

Wang, D., & Liu, K. (2007). A Novel QoS MAC Protocol for Multi-hop Ad Hoc Networks. *International Conference on Wireless Communications, Networking and Mobile Computing, WiCom* (pp. 1484-1487).

Xiao, H., Seah, W. K. G., Lo, A., & Chua, K. C. (2000). A Flexible Quality of Service Model for Mobile Ad hoc Networks. In *IEEE Vehicular Technology Conference Proceedings,* (pp. 445-449). New York: IEEE Inc.

Xue, Q., & Ganz, A. (2003). Ad hoc QoS on-demand routing (AQOR) in mobile ad hoc networks. *Special Issue on Routing in Mobile and Wireless Ad Hoc Networks*, *63*(2), 154–165.

Yihai, Z., & Gulliver, T. A. (2005). Quality of service for ad hoc on-demand distance vector routing. *IEEE International Conference on Wireless And Mobile Computing, Networking And Communications, WiMob'05* (pp. 192-196).

Ying, G., Kunz, T., & Lamont, L. (2003). Quality of service routing in ad-hoc networks using OLSR. In *Proceedings of the 36th Annual Hawaii International Conference on System Sciences*.

Ying, Z., Anand, A. L., & Jacob, L. (2003). A QoS enabled MAC protocol for multi-hop ad hoc wireless networks. *IEEE International Performance, Computing, and Communications Conference, IPCCC'03* (pp. 149- 156). New York: IEEE Inc.

Youngki, H., & Varshney, P. (2003). An adaptive QoS routing protocol with dispersity for ad-hoc networks. In *Proceedings of the 36th Annual Hawaii International Conference on System Sciences*.

ADDITIONAL READING

Aad, I., & Castelluccia, C. (2001). Differentiation mechanisms for IEEE 802.11. In *Proceedings of Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies,* IEEE INFOCOM'01, vol. 1, (pp. 209-218). New York: IEEE Inc.

Ahn, C. W., Kang, C. G., & Cho, Y. Z. (2000). Soft reservation multiple access with priority assignment (SRMA/PA): a novel MAC protocol for QoS-guaranteed integrated services in mobile ad hoc networks. *IEEE Vehicular Technology Conference*, vol. 2, (pp. 942–947).

An, H.-Y., Lu, X., Gong, Z., & Peng, W. (2005). A Cluster-Based QoS Multipath Routing Protocol for Large-Scale MANET. High Performance Computing and Communications (pp. 321-330).

An, H.-Y., Zhong, L., Lu, X.-C., & Peng, W. (2005). A cluster-based multipath dynamic source routing in MANET. *IEEE International Conference on Wireless And Mobile Computing, Networking And Communications* (pp. 369-376).

Badis, H., & Khaldoun, K. A. (2007). *Quality of Service for Ad hoc Optimized Link State Routing Protocol (QOLSR)*. (Internet draft, expired August 9, 2007). March 2007. The URL was http://ietfreport.isoc.org/all-ids/draft-badis-manet-qolsr-05.txt.

Baldwin, R. O., Davis, N. J., & Midkiff, S. F. (1999). A real-time Medium Access Control protocol for ad hoc wireless local area networks. *Mobile Computing and Communications Review*, *3*(2), 20–27. doi:10.1145/584027.584028

Barua, G., & Chakraborty, I. (2002). Adaptive routing for ad hoc wireless networks providing QoS guarantees. *IEEE International Conference on Personal Wireless Communications* (pp. 196-200).

Bernhardt, R. C., Cain, J. B., & Windham, W. A. (2007). QoS Architecture for a Mobile Ad Hoc Network. In [New York: IEEE Inc.]. *Proceedings of Military Communications Conference*, *2007*, 1–6. doi:10.1109/MILCOM.2007.4455142

Canales Compes, M., Gallego Martinez, J. R., Hernández Solana, A., & Valdovinos Bardaji, A. (2008). QoS provision in mobile ad hoc networks with an adaptive cross-layer architecture. Wireless Networks.

Cardei, I., Varadarajan, S., Pavan, A., Graba, L., Cardei, M., & Min, M. (2004). Resource Management for Ad-Hoc Wireless Networks with Cluster Organization. *Cluster Computing*, *7*(1), 91–103. doi:10.1023/B:CLUS.0000003946.65880.25

Cerda, L., Voorhaen, M., Guimaraes, R., Barcelo, J. M., Garcia, J., & Blondia, C. (2005). A Reservation Scheme Satisfying Bandwidth QoS Constraints for Ad-Hoc Networks. EURO-NGI network of excellence, International Workshop, (pp. 176-188).

Chen, K., Shah, S. H., & Nahrstedt, K. (2002). Cross-Layer Design for Data Accessibility in Mobile Ad Hoc Networks. *Journal of Wireless Personal Communications*. *Special Issue on Multimedia Network Protocols and Enabling Radio Technologies*, *21*, 49–75.

Chen, L., & Heinzelman, W. B. (2004). Network architecture to support QoS in mobile ad hoc networks. In *Proceedings of IEEE International Conference on Multimedia and Expo* (pp. 1715-1718). New York: IEEE Inc.

Chen, L., & Heinzelman, W. B. (2007). A Survey of Routing Protocols that Support QoS in Mobile Ad Hoc Networks. *IEEE Network Magazine*, *21*(6), 30–38. doi:10.1109/MNET.2007.4395108

Chen, N., Li, L., & Dong, W. (2006). Multicast routing algorithm of multiple QoS based on widest-bandwidth. *Journal of Systems Engineering and Electronics*, *17*, 642–647. doi:10.1016/S1004-4132(06)60110-2

Choi, S., del Pedro, J., Shankar, N. S., & Mangold, S. (2003). IEEE 802.11e contention-based channel access (EDCF) performance evaluation. *IEEE International Conference on Communications* (pp. 1151-1156). New York: IEEE Inc.

Hanzo-II, L., & Tafazolli, R. (2007). A Survey of QoS Routing Solutions for Mobile Ad Hoc Networks. *IEEE Communications Surveys & Tutorials*, 9(2), 50-70.

Jawhar, I., & Wu, J. (2005). QoS support in TDMA-based mobile ad hoc networks. *Journal of Computer Science and Technology*, *20*(6), 797–810. doi:10.1007/s11390-005-0797-3

Jurdak, R. (2007). Wireless Ad Hoc and Sensor Networks: A Cross-Layer Design Perspective. New York, NY: Springer Science+Business Media.

Kosek, K., Szott, S., Natkaniec, M., & Pach, A. R. (2008). Practical Analysis of IEEE 802.11b/g Cards in Multirate Ad-hoc Mode. *Journal of Telecommunications and Information Technology*. *JTIT*, *08*, 1.

Kumar, S., Raghavan, V. S., & Deng, J. (2004). Medium Access Control protocols for ad hoc wireless networks: A survey. [Elsevier.]. *Ad Hoc Networks*, *4*(3), 326–358. doi:10.1016/j.adhoc.2004.10.001

Mangold, S., Choi, S., May, P., Klein, O., Hiertz, G., & Stibor, L. (2002). IEEE 802.11e wireless LAN for quality of service. *In Proceedings of the European Wireless* (pp. 32-39).

Marwaha, S., Indulska, J., & Portmann, M. (2008). Challenges and Recent Advances in QoS Provisioning, Signaling, Routing and MAC protocols for MANETs. *Australian Telecommunication Networks and Applications Conference,* ATNAC'08 (pp. 97-102).

Masoudifar, M. (2009). A review and performance comparison of QoS multicast routing protocols for MANETs. *Ad Hoc Networks*, *7*, 1150–1155. doi:10.1016/j.adhoc.2008.10.004

Mishra, A. (2008). Security and Quality of Service in Ad Hoc Wireless Networks. New York, NY: Cambridge University Press.

Mohapatra, P., & Krishnamurthy, S. (Eds.). (2005). Ad Hoc Networks: Technologies and Protocols. Boston, MA: Springer Science+Business Media.

Mohapatra, P., & Li, J., & Gui, C. (2003). QoS in Mobile Ad hoc Networks. Special Issue on QoS in Next-Generation Wireless Multimedia Communications Systems in IEEE Wireless Communications Magazine, 10(3).

Perkins, C. E. (2003). *Quality of Service for Ad hoc On-Demand Distance Vector Routing*. (Internet draft, expired April 14, 2004). October 14, 2003. The URL was http://www.psg.com/~charliep/txt/aodvid/qos.txt.

Reddy, T. B., Karthigeyan, I., Manoj, B. S., & Murthy, C. S. R. (2006). Quality of service provisioning in ad hoc wireless networks: a survey of issues and solutions. *Ad Hoc Networks*, *4*(1), 83–124. doi:10.1016/j.adhoc.2004.04.008

Sarkar, S. K., Basavaraju, T. G., & Puttamadappa, C. (2008). Ad Hoc Mobile Wireless Networks: Principles, Protocols, and Applications. Boca Raton, FL: Auerbach Publications.

Sarma, N., & Nandi, S. (2006). QoS support in mobile ad hoc networks. IFIP International Conference on Wireless and Optical Communications Networks.

Sarma, N., & Nandi, S. (2008). A Cross-layer QoS Mapping Framework for Mobile Ad Hoc Networks. [Mumbai, India: Medknow Publications.]. *IETE Technical Review*, *25*(6), 346–358. doi:10.4103/0256-4602.45427

Sethi, P., & Barua, G. (2003). CRESQ: providing QoS and security in ad hoc networks. In *Proceedings Eleventh Euromicro Conference on Parallel, Distributed and Network-Based Processing* (pp. 544-550).

Sheu, S. T., & Sheu, T. F. (2001). A bandwidth allocation/sharing/extension protocol for multimedia over IEEE 802.11 ad hoc wireless LANs. *IEEE Journal on Selected Areas in Communications*, *19*(10), 2065–2080. doi:10.1109/49.957320

Sun, B., & Li, L. (2006). QoS-aware multicast routing protocol for Ad hoc networks. *Journal of Systems Engineering and Electronics*, *17*, 417–422. doi:10.1016/S1004-4132(06)60071-6

Sun, B., Pi, S., Gui, C., Zeng, Y., Yan, B., & Wang, W. (2008). Multiple constraints QoS multicast routing optimization algorithm in MANET based on GA. *Progress in Natural Science*, *18*, 331–336. doi:10.1016/j.pnsc.2007.11.006

Vaidya, N. H., Bahl, S., & Gupta, S. (2000). Distributed fair scheduling in a wireless LAN. *6th Annual International Conference on Mobile Computing and Networking* (pp. 167-178).

Vishnumurthy, V., Sandeep, T., Manoj, B. S., & Murthy, C. S. R. (2004). A Novel Out-of-band Signaling Mechanism for Enhanced Real-time Support in Tactical Ad hoc Wireless Networks. In *Proceedings of the 10th IEEE Real-Time and Embedded Technology and Applications Symposium* (pp. 56-63). Washington: IEEE Computer Society.

Yeh, C. H., & You, T. (2003). A QoS MAC Protocol for Differentiated Service in Mobile Ad Hoc Networks. *International Conference on Parallel Processing*, ICPP'03 (pp. 349-356).

You, T., Yeh, C. H., & Hassanein, H. (2005). DRCE: A High Throughput QoS MAC Protocol for Wireless Ad Hoc Networks. *10th IEEE Symposium on Computers and Communications*, ISCC'05 (pp. 671- 676). New York: IEEE Inc.

Zhang, L., Cai, L., Li, M., & Wang, F. (2009). A method for least-cost QoS multicast routing based on genetic simulated annealing algorithm. *Computer Communications*, *32*, 105–110. doi:10.1016/j.comcom.2008.09.023