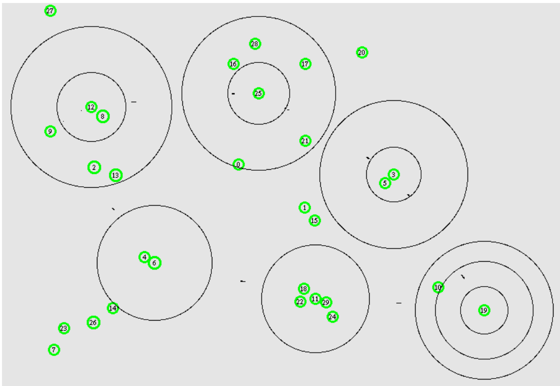
INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa



Study of Multipath Routing Protocols for Wireless Multimedia Sensor Networks

Naércio David Pedro Magaia

Dissertação para obtenção de Grau de Mestre em

Engenharia de Redes de Comunicação

Jurí

Presidente:	Prof. Doutor Paulo Jorge Pires Ferreira
Orientador:	Prof. Doutor Paulo Rogério Barreiros D'Almeida Pereira
Co-orientador:	Prof. Doutor António Manuel Raminhos Cordeiro Grilo
Vogal:	Prof. Doutor Luís Filipe Lourenço Bernardo

October 2011

Acknowledgments

From the formative stages of this thesis to the final version, I owe an immense debt of gratitude to both my advisers, Prof. Doutor Paulo Rogério Barreiros D'Almeida Pereira and Prof. Doutor António Manuel Raminhos Cordeiro Grilo, who were abundantly helpful and offered invaluable assistance, support and guidance. Their sound advices and careful guidance were invaluable during the course of the whole work.

I would also like to thank my family, who offered me unconditional love and support throughout the course of this thesis.

This work was supported in part by the FP7 WSAN4CIP (Wireless Sensor and Actuator Networks for the Protection of Critical Infrastructures) project under grant agreement no. 225186 as well as by national funding from FCT - Fundação para a Ciência e a Tecnologia (INESC-ID multiannual funding) through the PIDDAC program funds and by the MPSat project (FCT Ref. PTDC/EEA-TEL/099074/2008).

Lisbon, October 2011
Naércio David Pedro Magaia

Resumo

Encaminhamento multicaminho permite a construção e o uso de múltiplos caminhos para o encaminhamento entre um par origem-destino. Isto é conseguido explorando a redundância de recursos e diversidade na rede subjacente para fornecer benefícios tais como tolerância a falhas, balanceamento de carga, a agregação de largura de banda, e melhoria nos indicadores de qualidade de serviço (QoS), como desempenho, atraso e perda.

Neste trabalho, propomos uma extensão multicaminho para o Dynamic Source Routing Protocol (MeDSR) apropriado para Redes Multimédia de Sensores Sem fio (WMSN) e estudamos o seu desempenho através de simulação.

Primeiro, foi projectado um algoritmo para encaminhamento multicaminho que toma em conta aspectos importantes, tais como: selecção de caminhos em que as transmissões entre os nós num caminho não interfiram com as dos nós noutro caminho, e o atraso ponto a ponto. Como o nó destino recebe pacotes de solicitação de rotas, ele agrupa-os e responde ao nó origem, resposta esta contendo um conjunto de caminhos para alcançá-lo. O nó destino envia tantas respostas quantos os pedidos recebidos, cujos caminhos sejam independentes em relação aos nós. Isto porque é necessário recolher a informação acerca dos nós vizinhos de cada nó que faz parte de cada caminho independente. O nó origem selecciona os melhores caminhos tomando em conta o menor número de vizinhos em comum para os nós dos diferentes caminhos recebidos, tentando assim minimizar as interferências rádio entre os caminhos a utilizar.

Por último, foi utilizado um mecanismo de intercalação de transmissões de pacotes no nó origem para reduzir ainda mais a interferência entre os vários caminhos.

Abstract

Multipath routing allows building and using multiple paths for routing between a source-destination pair. It exploits the resource redundancy and diversity in the underlying network to provide benefits such as fault tolerance, load balancing, bandwidth aggregation, and improvement in Quality of Service (QoS) metrics such as throughput, delay and loss.

In this work, we propose a multipath extension to Dynamic Source Routing protocol (MeDSR) appropriate for Wireless Multimedia Sensor Network (WMSN) and study its performance through simulation.

First, a multipath routing algorithm was proposed, which takes into account important aspects such as selection of paths in which the transmissions between nodes of one path do not interfere with those of the other path, and the end to end delay. As the destination node receives route request packets, it groups them and responds to the source node, a response containing a set of paths to reach it. The destination node sends as many answers as the requests received whose paths were node independent between them. This is done because it is necessary to gather information about the neighboring nodes of each node that is part of the independent path. The source node selects the best paths taking into account the lowest number of neighbors in common among the different paths received. The purpose is to minimize radio interference between the paths to be used.

Finally, we used a mechanism for interleaving packet transmissions at the source node to further reduce interference between the multiple paths.

Palavras-chave
Keywords

Palavras-Chave

Rede Multimédia de Sensores Sem Fio

Encaminhamento Multicaminho

Acoplamento de Rotas

Keywords

Wireless Multimedia Sensor Networks

Multipath Routing

Route Coupling

Index

1 Introduction....1
1.1 Motivation....1
1.2 Objectives....2
1.3 Results....2
1.4 Structure of the Document....2
2 Related Work....3
2.1 Wireless Multimedia Sensor Networks....3
2.1.1 Multimedia Encoding Techniques....5
2.1.2 Multistream Video Coding....6
2.1.3 Cross-Layer Issues....8
2.2 Routing Protocols....8
2.3 Multipath Routing....13
2.3.1 Benefits of multipath routing....13
2.3.2 Elements of a multipath routing protocol....14
2.3.3 Multipath Video Transmission....15
2.3.4 Multipath Routing Protocols....16
2.4 Routing Metrics....20
2.5 Comparison and Discussion....21
3 Multipath Extension to the DSR....24
3.1 The Routing Algorithm....24
3.1.1 Route Discovery....25
3.1.2 Route Maintenance....27
3.2 Neighbor Information Management....27
3.2.1 Obtaining neighbor Information....27
3.2.2 Maintaining Neighbor Information....27
3.3 Multipath Selection Process....28
3.4 Route Coupling Technique....29
3.5 Implementation Decisions....29

4 Evaluation 33
4.1 Simulation Model 33
4.2 Simulation Results 34
4.2.1 Throughput 34
4.2.2 Delay 36
4.2.3 Packet Drop Ratio 38
4.2.4 Control Overhead 39
4.2.5 Energy Consumption 41
5 Conclusions 44
5.1 Conclusions 44
5.2 Future Work 44
6 Bibliography 46

List of Figures

Figure 2.1 A taxonomy of routing protocols in WSNs. 10

Figure 2.2 The SPIN protocol [W. K. Heinzelman 1999] (extracted from [I. S. Akyildiz 2002]). 12

Figure 2.3 An example of directed diffusion [Intanagonwiwat 2000]: (a) propagate interest, (b) set up gradient and (c) send data (extracted from [I. S. Akyildiz 2002]). 12

Figure 3.1 An illustration of the design goal of MeDSR. 24

Figure 3.2 The use of independent node disjoint path from the source node 9 to the destination node 19. Nodes on the first path are 9, 12, 25, 3 and 19 and on the other path: 9, 6, 11 and 19. 28

Figure 3.3 The use of node disjoint paths from the source node 12 to the destination node 22. Nodes on the first path are: 12, 6, 16, 13 and 22 and on the other, 12, 18, 27, 5 and 22. 29

Figure 4.1 Throughput comparison between DSR, AOMDV, MeDSR1 and MEDSR2 using CBR traffic. 35

Figure 4.2 Throughput comparison between DSR, AOMDV, MeDSR1 and MEDSR2 using Pareto traffic. 36

Figure 4.3 Delay comparison between DSR, AOMDV, MeDSR1 and MEDSR2 using CBR traffic. 37

Figure 4.4 Delay comparison between DSR, AOMDV, MeDSR1 and MEDSR2 using Pareto traffic. 37

Figure 4.5 Drops comparison between DSR, AOMDV, MeDSR1 and MEDSR2 using CBR traffic. 39

Figure 4.6 Drops comparison between DSR, AOMDV, MeDSR1 and MEDSR2 using Pareto traffic. 39

Figure 4.7 Control overhead comparison DSR, AOMDV, MeDSR1 and MEDSR2 using CBR traffic 40

Figure 4.8 Control overhead comparison between DSR, AOMDV, MeDSR1 and MEDSR2 using Pareto traffic 41

Figure 4.9 Energy consumption comparison DSR, AOMDV, MeDSR1 and MEDSR2 using CBR traffic. 42

Figure 4.10 Energy consumption comparison between DSR, AOMDV, MeDSR1 and MEDSR2 using Pareto traffic. 42

List of Tables

Table 2.1 Checklist for choosing a multipath routing protocol [Extracted from [Tarique 2009]] 22
Table 4.1 Simulation Parameters 34
Table 4.2 Average Throughput 95% Confidence Intervals 35
Table 4.3 Throughput Gains Comparison (%) 36
Table 4.4 Average Delay 95% Confidence Intervals 37
Table 4.5 Delay Gains Comparison (%) 38
Table 4.6 Average Packet Drops 95% Confidence Intervals 38
Table 4.7 Packet Drops Gains Comparison (%) 39
Table 4.8 Average Overhead 95% Confidence Intervals 40
Table 4.9 Overhead Gains Comparison (%) 41
Table 4.10 Average Energy Consumption 95% Confidence Intervals 42
Table 4.11 Energy Consumption Gains Comparison (%) 43

Acronyms

ACK Acknowledgment

ADV Advertisement

AODV Ad hoc On-Demand Distance Vector

AOMDV Ad hoc On-Demand Multipath Distance Vector Routing

ARQ Automatic Repeat Request

BS Base Station

CMOS Complementary Metal-Oxide-Semiconductor

DSR Dynamic Source Routing

EIFS Extended InterFrame Space

ID Identifier

IP Internet Protocol

JPEG Joint Photographic Experts Group

LC Layered Coding

LEACH Low Energy Adaptive Clustering Hierarchy

MAC Medium Access Control

MDC Multiple Description Coding

MDR Multipath on-Demand Routing

MDSR Multipath Dynamic Source Routing

MeDSR Multipath Extension to Dynamic Source Routing

MPEG Moving Picture Experts Group

NACK Negative Acknowledgment

NS-2 The Network Simulator 2

PLCP Physical Layer Convergence Protocol

PUMA Protocol for Unified Multicasting Through Announcements

QoS Quality of Service

REQ Request

RMPSR Robust Multipath Source Routing Protocol

RRER Route Error

RREP Route Reply

RREQ Route Request

RPS Reference Picture Selection

SMR Split Multipath Routing

SPIN Sensor Protocol for Information via Negotiation

TDMA Time Division Multiple Access

TORA Temporally-ordered Routing Algorithm

TPGF Two Phase Geographical Greedy Forwarding Routing

TTL Time-to-Live

UWB Ultra-Wideband

WMSN Wireless Multimedia Source Routing

WSN Wireless Sensor Networks

1 Introduction

This thesis addresses the problem of multipath routing in Wireless Multimedia Sensor Networks. We study the use of multiple paths to facilitate the discovery and deployment of non-interfering multiple path routes in these settings. Our aim is to create a solution that offers a good tradeoff between use of multiple paths, the interference among nodes in those paths and the end-to-end delay.

1.1 Motivation

Recent advances in electronics and wireless communications have led to the development of tiny, low cost, low energy and active sensors. Sensor nodes consist of sensing, data processing and communication components.

Wireless Sensor Networks (WSN) [I. S. Akyildiz 2002] are ad-hoc networks that can be established with no need for a pre-existent communications infrastructure. WSN nodes (sensors or actuators) collaborate to forward sensor data hop-by-hop from the source node to the sink nodes and vice versa.

By adding multimedia capabilities to sensors (CMOS cameras and microphones), sensor networks are able to capture multimedia contents from the environment, enabling the development of Wireless Multimedia Sensor Networks (WMSN) [I. F. Akyildiz 2007], i.e. networks of wirelessly interconnected devices that allow retrieving video and audio streams, still images and scalar data. WMSN enable several new applications such as: multimedia surveillance sensor networks, storage of potentially relevant activities, traffic avoidance, enforcement and control systems, advanced health care delivery, automated assistance for the elderly and family monitors, environmental monitoring, person locator services, industrial process control and many others.

A multimedia content such as a video stream requires a transmission bandwidth that is orders of magnitude higher than those typically necessary for a scalar sensor. So the use of a single path routing protocol may not provide enough performance for demanding applications.

Through multipath routing (i.e. using more than one path in parallel), load can be balanced and network capacity can be leveraged by increasing spatial reuse. According to [Harish 2005], spatial reuse refers to the scheduling of multiple (mutually non-interfering) transmissions simultaneously when all the links are operating in the same channel.

The maximum achievable throughput improvement of multipath is determined by the achievable degree of spatial reuse rather than the number of paths used, and having more than 2 paths does not significantly improve the performance, as the source and destination nodes are common to all multiple paths [Liaw 2004].

The route coupling effect (i.e. interference between multiple paths) can limit the effectiveness of load balancing in wireless networks, even with node disjointed paths. So, it is highly desirable for multipath routing the use of independent paths, where transmissions along different paths do not interfere with each other (except at the end nodes) [Liaw 2004].

1.2 Objectives

This work addresses the problem of providing support to maximize throughput and network life time and minimizing the end-to-end delay, drops and control overhead in Wireless Multimedia Sensor Networks. This thesis studies, implements and evaluates techniques to discover and maintain routes in an efficient manner, aiming at minimizing interferences among transmissions on neighboring nodes. As a result, this thesis proposes a Multipath extension to Dynamic Source Routing (MeDSR) protocol that considers multiple paths that do not interfere with each other, minimizing interference between transmissions of neighboring nodes.

1.3 Results

The results produced by this thesis can be enumerated as follows:

- A study to determine the best set of properties a multipath routing algorithm should have in order to find non-interfering paths.
- A specification of the MeDSR protocol and its implementation for the NS-2 simulation platform.
- An experimental evaluation of the implemented MeDSR comparing it with the Dynamic Source Routing (DSR), and with the Ad hoc On-demand Multipath Distance Vector (AOMDV), a single path and a multipath routing protocol respectively.
- A mechanism of interleaving packet transmissions at the source for MeDSR, to reduce interference between the multiple paths, and its performance evaluation.

1.4 Structure of the Document

The rest of this document is organized as follows. Chapter 2 provides an introduction to the different technical areas related to this work. Chapter 3 introduces MeDSR and Chapter 4 presents the results of the experimental evaluation study. Finally, Chapter 5 concludes this document by summarizing its main points and future work.

2 Related Work

This section starts by addressing WMSNs characteristics and routing protocols. Then, we survey multipath routing including protocol operation, main multipath routing problems, evaluation of routing protocols and some examples.

2.1 Wireless Multimedia Sensor Networks

As mentioned before, the availability of inexpensive hardware such as CMOS cameras and microphones that are able to ubiquitously capture multimedia content from the environment has fostered the development of WMSNs.

WMSNs will not only enhance existing sensor network applications such as tracking, home automation, and environmental monitoring, but they will also enable several new applications [I. F. Akyildiz 2007] such as:

- Multimedia surveillance sensor networks
- Storage of potentially relevant activities
- Traffic avoidance, enforcement and control systems
- Advanced health care delivery
- Automated assistance for the elderly and family monitors
- Environmental monitoring
- Person locator services
- Industrial process control

WMSNs will be enabled by the convergence of communication and computation with signal processing and several branches of control theory and embedded computing. This cross-disciplinary research will enable distributed systems of heterogeneous embedded devices that sense, interact, and control the physical environment. There are several factors [I. F. Akyildiz 2007]that mainly influence the design of a WMSN, which are outlined below in this section.

- *Application-specific QoS requirements*. The wide variety of applications envisaged on WMSNs will have different requirements. In addition to data delivery modes typical of scalar sensor networks, multimedia data include snapshot and streaming multimedia content. Snapshot-type multimedia data contain event triggered observations obtained in a short time period. Streaming multimedia content is generated over longer time periods and requires sustained information delivery. Hence, a strong foundation is needed in terms of hardware and supporting high-level algorithms to deliver QoS and consider application-specific requirements. These requirements may pertain to multiple domains and can be expressed, amongst others, in terms of a combination of bounds on energy consumption, delay, reliability, distortion, or network lifetime.

- *High bandwidth demand*. Multimedia content, especially video streams require transmission bandwidth that is orders of magnitude higher than those typically necessary for scalar sensors. Hence, high data rate and low-power consumption transmission techniques need to be leveraged. In this respect, the ultra wideband (UWB) [FCC 2002] transmission technique seems particularly promising for WMSNs.
- *Multimedia source coding techniques*. Uncompressed raw video streams require excessive bandwidth for a multi-hop wireless environment. Hence, it is apparent that efficient processing techniques for lossy compression are necessary for multimedia sensor networks. It has recently been shown [Girod 2005] that the traditional balance of complex encoder and simple decoder can be reversed within the framework of the so-called distributed source coding, which exploits the source statistics at the decoder, and by shifting the complexity to this end, allows the use of simple encoders. Clearly, such algorithms are very promising for WMSNs and especially for networks of video sensors, where it may not be feasible to use existing video encoders at the source node due to processing and energy constraints.
- *Multimedia in-network processing*. WMSNs allow performing multimedia in-network processing algorithms on the raw data extracted from the environment. This requires new architectures for collaborative, distributed, and resource-constrained processing that allow for filtering and extraction of semantically relevant information at the edge of the sensor network. This may increase the system scalability by reducing the transmission of redundant information, merging data originated from multiple views, on different media, and with multiple resolutions. However, the cost of multimedia processing algorithms may be prohibitive for low-end multimedia sensors. Hence, it is necessary to develop scalable and energy-efficient distributed filtering architectures to enable processing of redundant data as close as possible to the periphery of the network.
- *Power consumption*. Power consumption is a fundamental concern in WMSNs, even more than in traditional wireless sensor networks. In fact, sensors are battery-constrained devices, while multimedia applications produce high volumes of data, which require high transmission rates, and extensive processing. While the energy consumption of traditional sensor nodes is known to be dominated by the communication functionalities, this may not necessarily be true in WMSNs. Therefore, protocols, algorithms and architectures to maximize the network lifetime while providing the QoS required by the application are a critical issue.

- *Flexible architecture to support heterogeneous applications*. WMSN architectures will support several heterogeneous and independent applications with different requirements. It is necessary to develop flexible, hierarchical architectures that can accommodate the requirements of all these applications in the same infrastructure.
- *Multimedia coverage*. Some multimedia sensors, in particular video sensors, have larger sensing radii and are sensitive to direction of acquisition (directivity). Furthermore, video sensors can capture images only when there is unobstructed line of sight between the event and the sensor. Hence, coverage models developed for traditional wireless sensor networks are not sufficient for planning the deployment of a multimedia sensor network.
- *Integration with Internet (IP) architecture*. It is of fundamental importance for the commercial development of sensor networks to provide services that allow querying the network to retrieve useful information from anywhere and at any time. For this reason, future WMSNs will be remotely accessible from the Internet, and will therefore need to be integrated with the IP architecture. The characteristics of WSNs rule out the possibility of all-IP sensor networks and recommend the use of application level gateways or overlay IP networks as the best approach for integration between WSNs and the Internet [Montenegro 2007].
- *Integration with other wireless technologies*. Large-scale sensor networks may be created by interconnecting local ‘’islands’’ of sensors through other wireless technologies. This needs to be achieved without sacrificing on the efficiency of the operation within each individual technology.

2.1.1 Multimedia Encoding Techniques

The captured multimedia content should ideally be represented in such a way as to allow reliable transmission over lossy channels (error-resilient coding), using algorithms that minimize processing power and the amount of information to be transmitted. The main design objectives of a coder for multimedia sensor networks are thus [I. F. Akyildiz 2007]:

- *High compression efficiency*. Uncompressed raw video streams require high data rates and thus consume excessive bandwidth and energy. It is necessary to achieve a high ratio of compression to effectively limit bandwidth and energy consumption.
- *Low complexity*. Multimedia encoders are embedded in sensor devices. Hence, they need to be of low complexity to reduce cost and form factors, and low-power to prolong the lifetime of sensor nodes.
- *Error resiliency*. The source coder should provide robust and error-resilient coding of source data.

It is known from information-theoretic bounds established by Slepian and Wolf for lossless coding [Slepian 1973] and by Wyner and Ziv [Wyner 1976] for lossy coding with decoder side information, that efficient compression can be achieved by leveraging knowledge of the source statistics at the decoder only. In this way, the traditional balance of complex encoder and simple decoder can be reversed [Girod 2005]. Techniques that build upon these results are usually referred to as distributed source coding. Distributed source coding refers to the compression of multiple correlated sensor outputs that do not communicate with each other [Xiong 2004]. Joint decoding is performed by a central entity that receives data independently compressed by different sensors. The encoder can be simple and low-power, while the decoder at the sink will be complex and loaded with most of the processing and energy burden. Other encoding and compression schemes that may be considered for source coding of multimedia streams include JPEG with differential encoding, distributed coding of images taken by cameras having overlapping fields of view, or multi-layer coding with wavelet compression.

The objective of a Wyner–Ziv video coder is to achieve lossy compression of video streams and achieve performance comparable to that of interframe encoding (e.g., MPEG), with complexity at the encoder comparable to that of intra-frame coders (e.g., Motion-JPEG).

2.1.2 Multistream Video Coding

In video communications, a receiver usually displays the received video continuously. Such continuous display requires timely delivery of video data, which further translates to stringent quality of service (QoS) requirements (e.g., delay, bandwidth, and loss) on the underlying network. For the successful reconstruction of received video, the path used for the video session should be stable for most of the video session period. Using multiple paths in parallel for a real-time multimedia session (called multipath transport) provides a new degree of freedom in designing robust multimedia transport systems.

For multipath transport to be helpful for sending compressed video, one must carefully design the video coder to generate streams so that the loss in one stream does not adversely affect the decoding of other streams. Therefore, a multi-stream encoder should strive to achieve a good trade-off between coding efficiency and error resilience. Three representative coding schemes [S. L. Mao 2005] are used as examples, which differ in terms of their operations and network requirements.

- *Feedback-Based Reference Picture Selection*. One simple way to generate multiple video streams is to code a video into one stream in a standard way and then disperse that stream onto multiple paths (e.g., sending bits corresponding to the even frames on one path and those for the odd frames on the other). This simple method, however, has poor performance since the streams on the two paths are dependent on each other. That is, the even frames are predicted from the previous (odd) frame, and vice versa. This method was improved by exploring the reference picture selection (RPS) technique [S. L. Mao 2003]. Specifically, the same time domain partitioning method (i.e., sending coded even and odd frames separately) is still used. However, a more network-aware coding method is used, which selects the reference picture based on feedback and estimated path status. Assume that the decoder sends a negative acknowledgment (NACK) for a frame if it is damaged or lost, and a positive one (ACK) otherwise. The encoder can then estimate the status of the paths and infer which of the previous frames are damaged. Based on the estimation, for a picture to be coded, the closest picture for which itself as well as its reference pictures have been transmitted on the better path is selected as the reference picture.

- *Layered Coding With Selective Automatic Repeat Request (ARQ)*. A second option is based on the popular layered coding technique, where a video frame is coded into a base layer and one or more enhancement layers. Reception of the base layer can provide low but acceptable quality, while reception of the enhancement layer(s) can further improve the quality over the base layer alone, but the enhancement layers cannot be decoded without the base layer. When the layered video is transmitted over multiple paths (e.g., two paths), the traffic allocator sends the base layer packets on one path and the enhancement layer packets on the other one. The path with a lower packet loss rate is used for the base layer if the two paths have different qualities. The receiver returns selective ARQ requests to the sender to report base layer packet losses. When the sender receives such requests, it retransmits the requested base layer packets on the enhancement layer path. The transmission bit rate for the enhancement layer will be reduced correspondingly, according to the bandwidth reallocated for base layer retransmissions.
- *Multiple Description Coding*. The third option is to use multiple description coding. MDC is a technique that generates multiple equally important substreams called descriptions. The decoder reconstructs the video from any subset of received descriptions, yielding a quality commensurate with the number of received descriptions.

The three schemes have their respective advantages and disadvantages. While in LC, a layer can only be decoded if the base layer and the previous enhancement layers are received and decoded, in MDC all descriptions can be used to decode the original data stream. However, a MDC codec has higher complexity and less compression efficiency than in LC. If feedback is not available, only MDC is applicable. If the decoding delay requirement is very stringent, RPS and MDC are the possible choices. If the delay requirements are not stringent, LC with ARQ is an additional choice. In terms of the received quality for the same total bandwidth usage (in an application where all the three schemes are candidates), LC with ARQ has the best performance for medium and high loss rates. MDC is well suited for low loss rates, and RPS outperforms the other two when the loss rate is very low [S. L. Mao 2003]. However, because RPS adapts its encoder based on receiver feedback, it is not applicable for streaming of pre-encoded video.

2.1.3 Cross-Layer Issues

In multi-hop wireless networks there is a strict interdependence among functions handled at all layers of the communication stack [I. F. Akyildiz 2007]. The physical, MAC, and routing layers together impact the contention for network resources. The physical layer has a direct impact on multiple access of nodes in wireless channels by affecting the interference at the receivers. The MAC layer determines the bandwidth allocated to each transmitter, which naturally affects the performance of the physical layer in terms of successfully detecting the desired signals. Different routing decisions alter the set of links to be scheduled, and thereby influence the performance of the MAC layer. Furthermore, congestion control and power control are also inherently coupled, as the capacity available on each link depends on the transmission power [I. F. Akyildiz 2007]. Moreover, specifically to multimedia transmissions, the application layer does not require full insulation from lower layers, but needs instead to perform source coding based on information from the lower layers to maximize the multimedia performance.

2.2 Routing Protocols

In general, routing in WSNs can be divided into flat-based routing, hierarchical-based routing, and location-based routing depending on the network structure [Al-Karaki 2004], [Akkaya 2005].

- *Flat-based routing*. All nodes are typically assigned equal roles or functionality.
- *Hierarchical-based routing*. Nodes will play different roles in the network.
- *Location-based routing*. Sensor nodes' positions are exploited to route data in the network.

A routing protocol is considered adaptive if certain system parameters can be controlled in order to adapt to current network conditions and available energy levels. Furthermore, these protocols can be classified into multipath-based, query-based, negotiation-based, QoS-based, or coherent-based routing techniques depending on the protocol operation [Al-Karaki 2004], [Akkaya 2005].

- *Multipath-based routing*. Multiple paths between source and destination nodes are used in order to enhance network performance. The fault tolerance (resilience) of a protocol is measured by the likelihood that an alternate path exists between a source and a destination when the primary path fails.
- *Query-based routing*. The destination nodes propagate a query for data (sensing task) from a node through the network, and a node with this data sends the data that

matches the query back to the node that initiated the query. For example, client C1 may submit a query to node N1 and ask: Is it raining in region 1? All the nodes have tables consisting of the sensing task queries they receive, and send data that matches these tasks when they receive it.

- *Negotiation-Based routing*. High-level data descriptors are used in order to eliminate redundant data transmissions through negotiation. Hence, the main idea of negotiation-based routing in WSNs is to suppress duplicate information and prevent redundant data from being sent to the next sensor or the base-station (BS) by conducting a series of negotiation messages before the real data transmission begins.
- *QoS-based routing*. The network has to balance between energy consumption and data quality. In particular, the network has to satisfy certain QoS metrics (delay, energy, bandwidth, etc.) when delivering data to the BS.
- *Coherent-based routing*. In noncoherent data processing routing, nodes will locally process the raw data before it is sent to other nodes for further processing. In coherent routing, the data is forwarded to aggregators after minimum processing.

In addition to the above, routing protocols can be classified into three categories, proactive, reactive, and hybrid, depending on how the source finds a route to the destination [Al-Karaki 2004].

- *Proactive routing*. A proactive routing protocol is also called “table driven” routing protocol. Using a proactive routing protocol, nodes continuously evaluate routes to all reachable nodes and attempt to maintain consistent, up-to-date routing information. Therefore, a source node can get a routing path immediately if it needs one. Due to periodic information exchanges, a proactive routing protocol generates large number of control messages in the network. Hence, proactive routing protocols are not considered suitable for WSN. To overcome the limitations of proactive routing protocols, reactive routing protocols are used.
- *Reactive routing*. Reactive routing protocols are also called “on-demand” routing protocols. In a reactive routing protocol, routing paths are searched only when needed. A route discovery operation invokes a route-determination procedure. The discovery procedure terminates either when a route has been found or no route is available after examination for all route permutations.
- *Hybrid routing*. Hybrid routing protocols are proposed to combine the merits of both proactive and reactive routing protocols and overcome their shortcomings.

Another class of routing protocols is called cooperative. In cooperative routing, nodes send data to a central node where data can be aggregated and may be subject to further processing, hence reducing route cost in terms of energy use [Al-Karaki 2004], [Akkaya 2005].

The classification is shown in Figure 2.1. In the rest of this section we present a detailed overview of the main routing paradigms in WSNs.

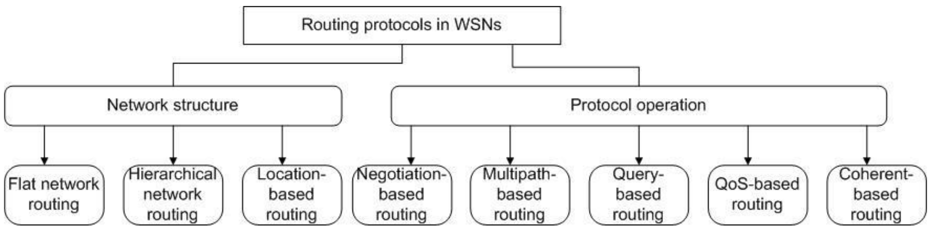


Figure 2.1 A taxonomy of routing protocols in WSNs.

In the next paragraphs we describe some relevant routing protocols used in WSNs. We start by giving a brief overview of Dynamic Source Routing (DSR) and Ad hoc On-demand Distance Vector Routing (AODV), given that many of the multipath routing protocols discussed in the next section are an extension of one of these two protocols.

DSR. The DSR protocol [Johnson 2007] consists of two basic mechanisms: (1) route discovery and (2) route maintenance.

Route discovery is the mechanism by which a source node discovers a route to a destination. When a source node wants to send a data packet, it first looks into the route cache to find a route. If a source cannot find a route in its route cache, the source initiates a route discovery mechanism by broadcasting a Route REQuest (RREQ) packet to its neighbors. When a neighbor of a source receives a RREQ packet, it first checks whether the packet is intended for it or not. If a neighbor discovers that it is the destination, it sends a reply back to the source after copying the accumulated routing information contained in the RREQ packet into a Route REPly (RREP) packet. If it is not the destination, it checks if there is any route available in the route cache for that destination. If this neighboring node is neither a destination nor does it has a route in the route cache to that destination, it appends its address in the RREQ packet, and then it rebroadcasts the RREQ packet to its neighbors. This process continues until a RREQ packet reaches the destination node. Then the destination node replies all RREQs. When a source node receives a RREP packet, it starts sending data packets using the route indicated in the reply packet. If multiple paths are discovered, it chooses the path that is the shortest one.

Route maintenance is the mechanism by which a node is able to detect any change in the network topology. When a node detects a broken link, for example, by using missing MAC layer acknowledgments, it removes the link from its route cache and sends a Route ERRor (RERR) message to each node that has sent packets over that link.

AODV. The AODV protocol [Perkins 2003] is called a pure on-demand routing protocol because a mobile node does not have to maintain any routing information if it is not located in an active path.

In AODV, when a source node wants to send packets to the destination but no route is available, it initiates a route discovery operation. In the route discovery operation, the source broadcasts RREQ packets. A RREQ includes addresses of the source and the destination, the broadcast ID, which is used as its identifier, the last seen sequence number of the destination as well as the source node's sequence number. Sequence numbers are important to ensure loop-free and up-to-date routes. To reduce the flooding overhead, a node discards RREQs that it has seen before and the expanding ring search algorithm is used in route discovery operation. The RREQ starts with a small TTL (Time-To-Live) value. If the destination is not found, the TTL is increased in following RREQs.

In AODV, each node maintains a cache to keep track of RREQs it has received. The cache also stores the path back to each RREQ originator. When the destination or a node that has a route to the destination receives the RREQ, it checks the destination sequence numbers it currently knows and the one specified in the RREQ. To guarantee the freshness of the routing information, a RREP packet is created and forwarded back to the source only if the destination sequence number is equal to or greater than the one specified in RREQ. AODV uses only symmetric links and a RREP follows the reverse path of the respective RREQ. Upon receiving the RREP packet, each intermediate node along the route updates its next-hop table entries with respect to the destination node. The redundant RREP packets or RREP packets with lower destination sequence number will be dropped.

In AODV, a node uses hello messages to notify its existence to its neighbors. Therefore, the link status to the next hop in an active route can be monitored. When a node discovers a link disconnection, it broadcasts a RERR packet to its neighbors, which in turn propagates the RERR packet towards nodes whose routes may be affected by the disconnected link. Then, the affected source can re-initiate a route discovery operation if the route is still needed.

Below are listed some relevant routing protocols for WSNs:

Sensor Protocol for Information via Negotiation. A family of adaptive protocols called SPIN [W. K. Heinzelman 1999] is designed to address the deficiencies of classic flooding by negotiation and resource adaptation. The SPIN family of protocols are designed based on two basic ideas: (1) Sensor nodes operate more efficiently and conserve energy by sending data that describe the sensor data instead of sending all the data; for example, image and sensor nodes must monitor the changes in their energy resources (battery lifetime); (2) Conventional protocols like flooding or gossiping-based routing protocols [Hedetniemi 1988] waste energy and bandwidth when sending extra and unnecessary copies of data by sensors covering overlapping areas.

SPIN has three types of messages, i.e., Advertisement (ADV), Request (REQ), and DATA. Before sending a DATA message, the sensor broadcasts an ADV message containing a descriptor, i.e., meta-data, of the DATA as shown in Step 1 of Figure 2.2. If a neighbor is interested in the data, it sends a REQ message for the DATA and DATA is sent to this neighbor sensor node as shown in Steps 2 and 3 of Fig. 2, respectively. The neighbor sensor node then repeats this process as illustrated in Steps 4, 5, and 6 of Figure 2.2. As a result, the sensor nodes in the entire sensor network, which are interested in the data, will get a copy.

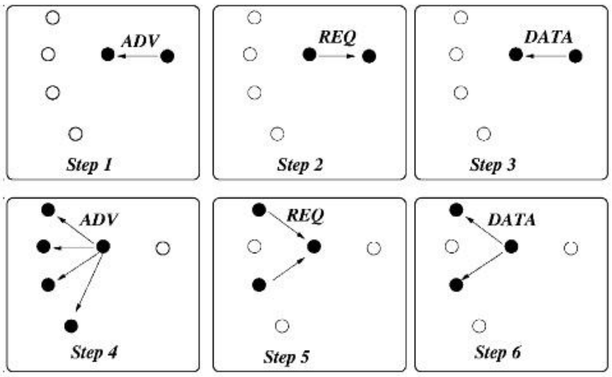


Figure 2.2 The SPIN protocol [W. K. Heinzelman 1999] (extracted from [I. S. Akyildiz 2002]).

Directed Diffusion. The directed diffusion data dissemination paradigm is proposed in [Intanagonwiwat 2000] where the sink sends out an interest, which is a task description, to all sensors as shown in Figure 2.3 (a). The task descriptors are named by assigning attribute value pairs that describe the task. Each sensor node then stores the interest entry in its cache. The interest entry contains a timestamp field and several gradient fields. As the interest is propagated throughout the sensor network, the gradients from the source back to the sink are set up as shown in Figure 2.3 (b). When the source has data for the interest, the source sends the data along the interest's gradient path as shown in Figure 2.3 (c). The interest and data propagation and aggregation are determined locally. Also, the sink must refresh and reinforce the interest when it starts to receive data from the source. Note that the directed diffusion is based on data-centric routing where the sink broadcasts the interest.

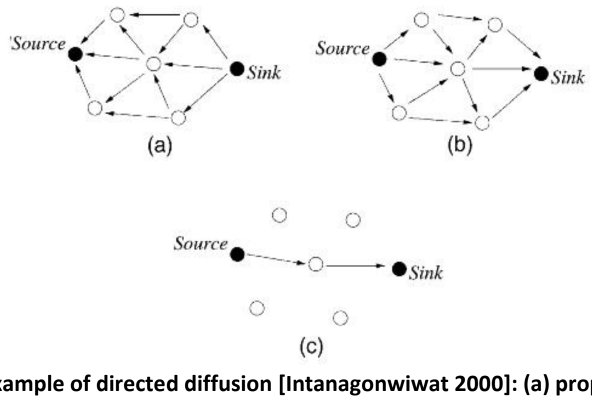


Figure 2.3 An example of directed diffusion [Intanagonwiwat 2000]: (a) propagate interest, (b) set up gradient and (c) send data (extracted from [I. S. Akyildiz 2002]).

Low Energy Adaptive Clustering Hierarchy. LEACH [W. C. Heinzelman 2000] is a clustering-based protocol that minimizes energy dissipation in sensor networks. The purpose of LEACH is to randomly select sensor nodes as cluster-heads, so the high energy dissipation in communicating with the base station is spread as equally as possible through all sensor nodes in the sensor network. The operation of

LEACH is separated into two phases: (1) the set-up phase and (2) the steady phase. The duration of the steady phase is longer than the duration of the set-up phase in order to minimize the overhead.

During the set-up phase, a sensor node chooses a random number between 0 and 1. If this random number is less than the threshold T, the sensor node is a cluster-head. After the cluster-heads are selected, they advertise to all sensor nodes in the network that they are the new cluster-heads. Once the sensor nodes receive the advertisement, they determine the cluster that they want to belong to based on the signal strength of the advertisement from the cluster-heads to the sensor nodes. The sensor nodes inform the appropriate cluster-heads that they will be a member of the cluster. Afterwards, the cluster-heads assign the time on which the sensor nodes can send data to the cluster-heads based on a Time Division Multiple Access (TDMA) approach.

During the steady phase, the sensor nodes can begin sensing and transmitting data to the cluster-heads. The cluster-heads also aggregate data from the nodes in their cluster before sending these data to the base station. After a certain period of time spent on the steady phase, the network goes into the set-up phase again and enters another round of selecting the cluster-heads.

2.3 Multipath Routing

As mentioned before, multipath routing can provide a range of benefits [Tsai 2006], [Mueller 2004]. In this section, we describe how these benefits are achieved, and give an overview of the main elements in multipath routing protocols.

2.3.1 Benefits of multipath routing

The main benefits of multipath routing are the following:

Fault tolerance – Multipath routing protocols can provide fault tolerance by having redundant information routed to the destination via alternative paths. This reduces the probability that communication is disrupted in case of link failure.

Load balancing – When a link becomes over-utilized and causes congestion, multipath routing protocols can choose to divert traffic through alternate paths to ease the burden of the congested link.

Bandwidth aggregation – By splitting data to the same destination into multiple streams, each routed through a different path, the effective bandwidth of different paths can be aggregated. This strategy is particularly beneficial when a node has multiple low bandwidth links but requires a bandwidth greater than an individual link can provide. End-to-end delay may also be reduced as a direct result of using a larger bandwidth.

Reduced delay – For wireless networks employing single path on-demand routing protocols, a route failure means that a new path discovery process needs to be initiated to find a new route. This results in a route discovery delay. The delay is minimized in multipath routing because backup routes are identified during route discovery.

2.3.2 Elements of a multipath routing protocol

There are three elements of multipath routing: path discovery, traffic distribution, and path maintenance [Mueller 2004].

Path Discovery. Path discovery is the process of determining the available paths for a source-destination pair. There are various criteria a protocol can use when deciding which subset, if not all, of possible paths it wants to find out in the discovery process.

Disjoint paths - The most commonly used criterion is the disjointedness of paths, which classifies the independence of paths in terms of shared resources. There are three main types of path disjointedness, namely non-disjoint, link-disjoint, and node-disjoint. A set of node-disjoint paths have no common nodes except the source and the destination. Similarly, link-disjoint paths have no common links, but may share some common intermediate nodes. And non-disjoint paths can have links (and therefore nodes) in common. Note that node-disjointedness implies link-disjointedness. In principle, when an intermediate node in a set of node-disjoint paths fails, only the path containing the failed node is affected, so there is minimum impact to the diversity of the routes. A link failure will only bring down one of multiple paths, whether they are link-disjoint or node-disjoint. However, a node failure could disable multiple links and break multiple link-disjoint paths. Non-disjoints paths offer the least degree of fault-tolerance as either node or link failure could affect multiple paths.

Route coupling – It was shown in [Pearlman 2000] that in wireless networks, route coupling caused by radio interference or contention between paths can have serious impacts on the performance of multipath routing protocols, even if the paths are topologically disjoint. In a wired network, route coupling is gauged by path disjointedness, but in a radio network routes are also considered heavily coupled if transmission on one route directly impedes the qualities of the other.

Traffic Distribution. There are various strategies for allocating traffic over available paths. A multipath protocol may decide to forward traffic using only the path with the best metric and keep other discovered paths as backups, or the paths may be used concurrently. A path selection algorithm is used to select a subset of available paths according to certain quality parameter of the paths. Hop-count has traditionally been a popular metric to use. Some other choices are: path reliability, disjointedness, available bandwidth, degree of route coupling, or a combination of metrics. In QoS routing, a subset of paths is only selected if the combined metric satisfies the QoS requirement.

Number of paths – As said before, a protocol can choose to use a single path and keep the rest as backups, or it can utilize multiple paths in a round-robin fashion, with only one path sending at a time. If multiple paths are used concurrently to carry traffic, the protocol needs to decide how traffic is split over the paths and how to handle out of sequence order packets at the destination. It is also possible to add a degree of redundancy when distributing traffic over multiple paths. A M-for-N diversity coding scheme is described in [Ayanoglu 1993]. In this coding scheme, M extra transmission lines are used to increase redundancy of an N transmission-line system. The traffic over the M + N system is coded in a way such that the system can tolerate less than M simultaneous line failures at any time. This idea is extended to multipath routing in packet networks in [Tsirigos 2001].

Allocation granularity – Some possible choices of traffic granularity include, in order of increased control overhead, per source-destination pair, per flow, per packet, per segment. With a fine granularity, load balancing can be more efficient, since a quicker adaptation to traffic fluctuations is achieved [Krishnan 1993]. Nevertheless, per packet or finer granularity require reordering at the destination, which may not suit some applications.

Path Maintenance. Over time, paths may fail due to link/node failures or node mobility. Path maintenance is the process of regenerating paths after the initial path discovery. It can be initiated after each path failure, or when all the paths have failed. Some multipath protocols use dynamic maintenance algorithms to constantly monitor and maintain the quality or combined QoS metric of available paths [Tsai 2006].

2.3.3 Multipath Video Transmission.

Video transportation over multipath routing has been addressed in [S. L. Mao 2005]. Real time video transmission has stringent delay, bandwidth and packet loss requirements. Real time video transport over WMSN is more challenging because of its dynamic topology, therefore being advantageous to use multiple paths in parallel for video transmission. The general multipath framework for video transmission is as follows: at the sender, the raw video is compressed by an encoder into M streams. When M > 1, the encoder is called a multi-stream coder. Then the streams are partitioned and allocated for K number of paths by a traffic allocator. These paths are maintained by a multipath routing protocol. When the video streams arrive at the receiver, these streams are put into a re-sequencing buffer to restore the original order. Finally, the video is extracted from the re-sequencing buffer to be decoded and displayed. The video decoder performs error correction when some data packets are lost.

In general, the quality of a path changes over time, and the system adjusts the transport and the coder accordingly to achieve the required QoS requirements. It is suggested in [S. L. Mao 2005] that multipath transport of video streams can cause even load distribution all over the network and hence can reduce congestion and improve delay. But this kind of distribution of traffic among different paths increases complexity and control overhead packets in the network. It is shown in [S. L. Mao 2005] that when the number of paths K is increased from 1 to 2, there is a significant improvement in network performance in terms of delay. But if K is greater than 2, there is not a significant performance improvement.

2.3.4 Multipath Routing Protocols

Multipath routing protocols can be broadly classified as follows: delay-aware, reliable, minimum overhead, energy efficient and hybrid multipath routing protocols [Tarique 2009].

- *Delay-aware multipath routing*. Delay-aware multipath routing protocols are those whose main objective is to ensure a fair load distribution among the mobile nodes so that no section of a network gets congested.
- *Reliable multipath routing*. Reliable multipath routing protocols have been proposed to provide reliable data communication between a source and a destination. These protocols try to cope with link errors, which arise due to communication through an unreliable wireless medium.
- *Minimum overhead multipath routing*. Minimum overhead multipath routing protocols are those that have been proposed for discovering and using multiple paths with minimum routing overhead. Multipath routing protocols inherently need to discover as many paths as possible. To discover as many paths as possible, a multipath routing protocol uses an additional type of control messages other than RREQ, RREP and RERR messages. The additional control messages are used by a mobile node to collect information about its neighbors so that suitable (node disjointed or link disjointed) paths are discovered. For example, a special control message called beckon is used in [Yao, et al. 2003] to create a neighbor table. Thus, a lot of control overhead messages are generated in the network to discover and to maintain these paths. That is why discovering multiple paths with low overhead is their main objective.
- *Energy efficient multipath routing*. Energy efficient multipath routing protocols are those that improve energy efficiency of a network. A mobile node is usually equipped with a battery of limited capacity. The major goal of energy aware routing is to maximize the network life by efficiently utilizing the battery of a mobile node.
- *Hybrid routing*. Hybrid routing protocols are those that use together both a single path and a multipath routing protocol. These multipath routing protocols use the shortest path algorithm at low traffic load conditions. But they switch to multipath routing when the network starts carrying higher traffic loads.

Below are listed some relevant routing protocols.

Multipath On-Demand Routing. An energy efficient multipath on-demand routing (MDR) has been introduced in [Dulman 2003]. The major objectives of the MDR protocol are: (1) provide on-demand disjoint paths between a source and a destination, (2) improve the reliability of data transmission while lowering the amount of overhead traffic, (3) achieve robustness against the changing network topology.

The MDR protocol originated from the DSR, but it has a different route maintenance option. The MDR protocol consists of two major phases: RREQ and RREP. When a source wants to discover a route to a destination, it floods the network with a request message. This message contains a source ID, destination ID and request ID. Unlike the DSR protocol, the RREQ packet does not collect routing information. Hence, a request packet does not get larger as it travels along a possible path. The destination will eventually receive one of the RREQ messages and returns a RREP to a neighbor from which it received the RREQ message. The request message contains a supplementary field to record the number of hops that it has traveled. When each node receives a RREP, it increments the hop count recorded in the reply packet of a message, and forwards the message to the neighbor from which it received the original RREQ.

In the MDR protocol, routing information is stored in the nodes. Since a data packet does not contain source routing information, each data packet has a smaller size compared to the DSR protocol. Moreover, the size of the RREQ is also smaller. Thus, each node can save energy while transmitting a packet of smaller size.

Split Multipath Routing. The main objective of SMR [Lee 2001] is to reduce the frequency of route discovery processes and thereby reduce the control overhead in the network. The protocol uses a per packet allocation scheme to distribute load into multiple paths. When a destination node receives RREQ packets from different paths, it chooses multiple disjoint routes and sends replies back to the source. The basic route discovery mechanism of the DSR protocol is used in the SMR protocol, but an intermediate node is not allowed to reply from its route cache if it has some routes available to that destination. To avoid overlapped multiple paths, instead of dropping every duplicate RREQs, intermediate nodes forward the duplicate packets that traversed through a different incoming link than the link from which the first RREQ is received, and whose hop count is not larger than that of the first received RREQ. When a destination node receives a RREQ message, it selects two paths that are maximally disjointed. Between these two routes, the first one is the shortest path. The shortest path is chosen to minimize the route discovery time because it is the earliest discovered route. After processing the first request, for the second path selection, a destination waits for a certain duration of time to receive more requests and learns all possible routes. After this, it selects a route from one of the alternative paths, which is maximally disjointed with the shortest path. A maximally disjointed path is the path that has the least number of common nodes compared to the shortest path. If there is more than one maximally disjointed path available, the shortest hop path is selected among them. Another major difference from DSR protocol is that an intermediate node does not need to maintain a route cache. For this reason, a node has a smaller cache. Although the SMR protocol uses less frequent route discovery mechanisms compared to the DSR protocol, one of the drawbacks of SMR is the redundant overhead packets. Since an intermediate node is not dropping a duplicate request message, the frequency of route discovery process needs to be reduced to curb the overhead.

Multipath Dynamic Source Routing. The MDSR protocol [Nasipuri 1999] reduces the flooding problem of the DSR protocol. One major disadvantage of the DSR protocol is the query flooding that is used to discover new source routes. Such query flooding generates a large number of overhead packets. These overhead packets occupy a substantial portion of the bandwidth of a network. It is shown in [Nasipuri 1999] that an intelligent multipath protocol can reduce the frequency of query flooding. In the DSR protocol, a destination node replies to every received request packet. But in the MDSR protocol, a destination replies only to a selected set of request messages. After receiving all requests, a destination replies back only to those RREQs that are link-disjointed from the primary source route (i.e., the shortest path route). A destination node keeps record of the shortest path and based on the shortest path information, it figures out which RREQ it should reply to. A source keeps all the routes in the cache. If the shortest route is broken, it uses an alternative route, which is the shortest among the remaining routes in the cache. If this route is also broken, it uses another alternative route. This process of choosing alternative paths continues until all paths are used. If all routes in a cache are broken, the source initiates another route discovery.

Simulation results show that although the route discovery frequency is reduced, an alternative path is usually longer and hence the delay per packet increases. In addition to this delay increase, there may be only a few intermediate nodes that have alternative paths to a destination and this can trigger frequent route discoveries and eventually increase overhead.

Ad hoc On-demand Multipath Distance Vector. AOMDV [Marina 2001] offers a multipath, loop-free extension to AODV. It ensures that alternate paths at every node are disjoint, therefore achieves path disjointedness without using source routing.

To support multipath routing, route tables in AOMDV contain a list of paths for each destination. All the paths to a destination have the same destination sequence number. Once a route advertisement with a higher sequence number is received, all routes with the old sequence number are removed. Two additional fields, hop count and last hop, are stored in the route entry to help address the problems of loop freedom, and path disjointedness, respectively.

Because the protocol implement multipath discovery, the loop freedom guarantee from AODV no longer holds. AOMDV address this issue as follows. The hop count field contains the length of the longest path for a particular destination sequence number, and is only initialized once, at the time of the first advertisement for that sequence number. Hence, the hop count remains unchanged until a path for a higher destination sequence number is received. It follows that loop freedom is ensured as long as a node never advertises a route shorter than one already advertised, and never accepts a route longer than one already advertised.

To ensure that paths in the route table are link-disjoint, a node discards a path advertisement that has either a common next hop or a common last hop as one already in the route table. It was observed that, as long as each node adheres to this rule, all paths for the same destination sequence number are guaranteed to be link-disjoint. Node-disjoint paths can be obtained with an additional restriction that for a particular destination sequence number, every node always advertises the same designated path to other nodes. Route maintenance in AOMDV is similar to that in AODV. A RERR for a destination is generated when the last path to that destination fails.

Robust Multipath Source Routing. In order to meet a stringent QoS requirement of a video application, the robust multipath source routing (RMPSR) protocol is introduced in [Wei 2004]. The basic idea behind RMPSR protocol is to discover multiple nearly disjoint routes between a source and a destination. The primary route connects a source and a destination node, and alternative routes connect an intermediate node to a destination. To increase the probability of discovering multiple disjoint routes, the path selection criteria [Wu 2001] includes the following properties: disjoint nodes, small distance between the primary (shortest) and the other paths, and small correlation factor. The correlation factor of two node disjoint paths is defined as the number of links connecting the two paths. A route set consists of a primary route and several alternative routes. The destination node collects multiple copies of RREQ packets of the same session within a time window, then builds multiple nearly disjoint route sets, and returns primary routes to the source node, and alternative routes to corresponding intermediate nodes.

The RMPSR protocol uses a per-packet allocation scheme to distribute video packets over two primary routes of two route sets. When one transmitting primary route is broken, the intermediate node that corresponds to the broken link will send a RERR packet to the source node. Upon receiving the RERR packet, the source node removes the broken primary route from its route cache, and switches the transmission to another primary route.

To support video applications, three schemes have been introduced as follows: (1) when a transmitting route is broken, an intermediate route is used to salvage the packets that are in the mid-way. That improves the delivery ratio, which is important in video communication, (2) the RMPSR protocol triggers a new RREQ process before the connectivity is entirely lost. That means the protocol triggers a new route discovery when there is only one primary route left in the route cache and (3) similar to other multipath extensions, RMPSR increases the probability of discovering multiple disjoint routes at the expense of an increase in control overhead. Both RMPSR and DSR are deployed at each node with different classes of traffic [I. F. Akyildiz 2007] being handled by different routing protocols, in order to alleviate the impact of routing overhead on the network. Video traffic is given higher priority using RMPSR, while other traffic is given lower priority using DSR. This scheme helps to maintain high quality of video applications when the data traffic in the network increases.

Two Phase geographical Greedy Forwarding Routing. TPGF [Shu 2007] focuses on exploring the maximum number of node-disjoint routing paths in the network layer in terms of minimizing the path length and the end to end transmission delay. Geographical routing uses the position of the destination to forward packets. In greedy forwarding, a node forwards a packet to the neighbor that has the least distance to the destination.

The TPGF routing algorithm includes two phases: Phase 1 is responsible for exploring the possible routing path. Phase 2 is responsible for optimizing the found routing path with the least number of hops. The TPGF routing algorithm finds one path per execution and can be executed repeatedly to find more node disjoint routing paths. It successfully addresses four important issues: 1) Hole-bypassing; 2) Guarantee path exploration result; 3) Node disjoint multipath transmission.

The Hole-bypassing has the highest priority in protocol realization, because it will highly affect the performance of multimedia streaming data transmission, if some holes cumber the routing paths. Holes are classified into the following two kinds: closed-circle hole and unclose-circle hole. Closed-Circle Hole means that the hole exists inside the sensor network, and it is fully surrounded by other active sensor nodes.

The Guarantee path exploration result has the second priority among the three design goals, because for most situations using only hole-bypassing routing algorithm can find the routing path for transmission. The feature of Guarantee path exploration result actually is designed to handle some special cases where using only hole-bypassing routing algorithm cannot find the existing routing paths. When a sensor node finds that it has no neighbor node available for the next-hop transmission (Block Situation), it will step back to its previous-hop node and mark itself as a block node (Step Back & Mark). The previous-hop node will attempt to find another available neighbor node as the next-hop node. The Step Back & Mark will be repeatedly executed until a sensor node successfully finds a routing path to the base station.

The Node disjoint multipath transmission has the lowest priority in the protocol realization, because once at least one routing path is built for a source node, such as an image node, some amount of data can be transmitted to the base station. The feature of Node disjoint multipath transmission can try to find additional routing paths for increasing the multimedia streaming data transmission when it is necessary. However, multipath transmission is not always guaranteed because there may be no more routing path exists inside the sensor network.

2.4 Routing Metrics

The route establishment phase includes the choice of paths among all the available to forward traffic. If various paths are available, we may want to choose a limited number: to use only the one with the best metric or to choose the n best ones. The number of disjoint paths to a node traversing a link can be considered as a metric for that link. Hop count is a widely used metric, since it allows quick path discovery in presence of mobility.

Path reliability and link quality are performance metrics used by a considerable number of quality-aware routing schemes. Path reliability can be defined as the probability of having a successful data transmission between two mobile nodes within a certain amount of time [Tsai 2006].

The following metrics that can be used to evaluate routing protocols [Baumann 2007]:

- *ETX (Expected Transmission Count)* is the expected number of transmissions a node needs to do to successfully transmit a packet to a neighbor. It is based on the delivery ratio of a number of packets, in a certain time interval. The ETX of a path is defined as the sum of the metric values of the links that form this path.
- *ETT (Expected Transmission Time)* considers link quality as a function of the time a data packet takes to be successfully transmitted to each neighbor.

In order to evaluate the multipath routing protocols, the following metrics have been chosen:

- *Throughput*: This metric represents the ratio between the number of data packets that are sent by the source and received by the sink during the simulation over the simulation time.
- *Average End-to-End Delay*: The end-to-end delay is averaged over all surviving packets from the source to the destination.
- *Packet Drops Ratio*: This metric represent the ratio between the number of dropped packets over all packets.
- *Energy Consumption*: The energy consumption measures the energy dissipated by the nodes throughout the entire simulation.
- *Control Overhead:* The control overhead represents the ratio between the total number of routing control packets over all packets.

2.5 Comparison and Discussion

The multipath routing protocols discussed in the previous section have their own advantages and disadvantages. Choosing a suitable multipath routing protocol is a very challenging task and must take into consideration the following factors [Tarique 2009]: (1) the size of a network, (2) the lifetime of a network, (3) environmental conditions, (4) types of applications and (5) node mobility.

The size of a network affects the performance of routing protocols. A small network does not affect the performance of the routing protocol. When a network gets larger, a large number of nodes participate in network operation and generate a large number of overhead packets in the network. The overhead packets occupy a considerable portion of the network bandwidth.

Energy constraint is another important factor while selecting a multipath routing protocol. Mobile nodes inherently have limited batteries unless they are attached to a power source.

Environmental conditions are another factor while choosing a multipath routing protocol. One of the major motivations for using an ad hoc network is that it can be easily deployed in an area where an infrastructure based network cannot be built or where infrastructure has been destroyed by a natural disaster or nuclear explosion. In such an environment, there are high interference and noise levels.

Choosing a multipath routing protocol is also determined by specific application requirements. Military applications require high reliability. The type of data that a network should support is also an important parameter. For example, the multipath routing protocol proposed in [S. L. Mao 2005] is a good candidate for video transmission.

The topology in a WMSN may change constantly due to the mobility of nodes. As nodes move in and out of range of each other, some links break while new links between nodes are created.

Based on the aforementioned factors, it is hard to choose a single multipath routing protocol that can satisfy all the performance objectives of a given network. Table 2.1 provides a summary of all the multipath protocols mentioned in this report.

Table 2.1 Checklist for choosing a multipath routing protocol [Extracted from [Tarique 2009]]

Protocol	Delay	Overhead	Reliability	Energy efficiency	Mobility support
Multipath video streaming	Low	High	High	Low	High
Multipath On-Demand Routing	Low	High	High	Low	High
Split multipath routing	High	Low	High	High	High
Multipath AODV	High	High	High	Medium	High
Multipath DSR	High	Low	High	Medium	Low
TPGF routing	Low	Low	High	Medium	Medium
Robust multipath source routing	Low	Low	High	High	Medium

The presented routing protocols are mostly based on DSR and AODV. SMR is one of the best known multipath extensions to DSR. It uses a modified RREQ packets flooding scheme in the process of route requesting. The destination node returns the shortest path and another path that is the most disjoint with the shortest path to the source node. The main goal of RMPSR is to minimize video packet loss caused by network topology changes, by building multiple disjoint route sets for the communication pair. RMPSR is similarly to DSR and SMR as also uses source routing, but it builds the route sets at the destination node and has smaller delay than SMR. The TPGF routing algorithm explores single or multiple optimized node disjoint hole-bypassing transmission paths, guaranteeing to find the routing paths if they exist in sensor networks. In [S. L. Mao 2005], for multipath transport of video streams, it is recommended to use an optimal number of 2 paths. But none of the above mentioned protocols takes into account the route coupling issue that have great influence over the nodes throughput.

Summary

This chapter introduced some fundamental concepts which will be central to the discussion in the next chapter. A survey of the main protocols and applications of multipath routing in Wireless Multimedia Sensor Networks was presented; however, there are no solutions that uses multipath to improve throughput, using low coupling routes to forward data.

As none of the existing multipath solutions considers the route coupling issue, the next Chapter introduces a protocol that addresses this gap.

3 Multipath Extension to the DSR

As discussed in the previous chapter, the use of multiple paths at the same time to route traffic can reduce the available bandwidth, instead of improving it. This Chapter introduces the Multipath extension to Dynamic Source Routing protocol (MeDSR), a multipath routing algorithm designed to improve bandwidth usage between a given source and destination. This improvement is achieved by using multiple non-interfering paths to communicate. A description of the system and its components can be found in the following sections.

To describe MeDSR, we need to introduce some definitions:

Definition 3.1: The ***correlation factor (α)*** of two paths is defined as the number of shared neighbor nodes between the 2 paths. If there are no shared neighbor nodes between two paths, we say the two paths have are unrelated ($\alpha = 0$). Otherwise, the two paths have a correlation factor of α.

Definition 3.2: The ***route set*** consists only of a primary route and neighborhood information for each node of the primary route. The primary route connects the source node and the destination node. We do not use alternative routes as in RMPSR [Wei 2004], because that information is obtained by overhearing messages. We consider neighbor nodes the ones within successful transmission of range. Neighborhood information corresponds to the neighbor nodes of each node in the path.

Consider Figure 3.2, where there are two node disjoint paths from the source node S to the destination node D. Two primary routes are being used to forward packets. Nodes on the first primary route are S, 1, 2, 3, 4 and D and on the other S, 5, 6, 7, 8 and D. The neighborhood information for the first path is {S, 1, 2, 3, 4, D} and for the other path is{S, 5, 6, 7, 8, D}. The correlation factor for this case is $\alpha = 0$ as these two paths are unrelated.

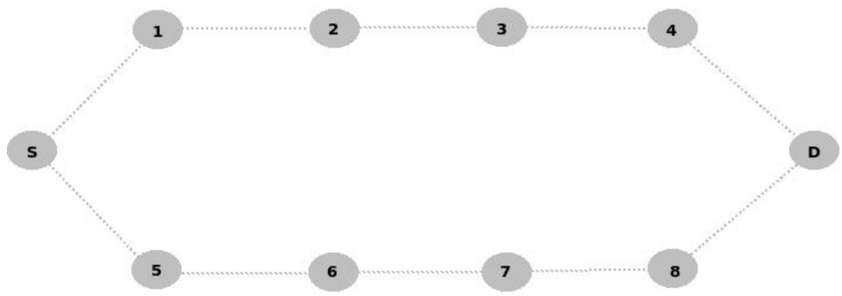


Figure 3.1 An illustration of the design goal of MeDSR.

3.1 The Routing Algorithm

The idea of MeDSR is to build node disjoint route sets for the source-destination pairs as in Figure 3.1. MeDSR inherits both desirable features of other routing protocol approaches, and applies some new features to address requirements of WMSN. According to its functionality, MeDSR is a reactive protocol, as it finds and maintains paths to route traffic between a source and a destination when they are needed. A RREQ is broadcasted when a path is requested by a source node and the destination replies to the request by sending a RREP containing the location of the destination node.

The decision about which paths should be used to route traffic is made by the source node but to make this decision it needs to know about the distribution of nodes in the network in order to select non-interfering paths.

Similar to DSR, we use an on-demand source routing approach. In source routing, each data packet contains complete routing information to reach its destination. The reasons for choosing source routing are that (a) it has been shown to outperform table based approaches in many scenarios [Broch 1998], and (b) it is possible to build node disjoint route sets using source routing, since the destination node knows the entire path of all the available routes.

As DSR, our routing protocol consists of two basic mechanisms: route discovery and route maintenance.

3.1.1 Route Discovery

When a source needs to establish routes to a destination, it originates a route discovery process. If it is the first time, the route discovery process typically involves a network-wide flood of RREQ packets targeting the destination, and the return of RREP packets from the destination. In DSR protocol, duplicate copies of the RREQ packets at the intermediate nodes are discarded. Although this approach can minimize the number of RREQ packets for one route discovery process, it also reduces the probability of discovering multiple disjoint routes [Wu 2001]. In order to increase the probability of discovering multiple disjoint routes, while keeping the number of RREQ packets small, we use a modified form of the RREQ packet forwarding scheme as in [Wu 2001], i.e. when a node receives a RREQ, if it is the first time this RREQ is received or the path included in this message is node disjoint with the path included in a previously cached copy of the same RREQ, then the node will cache it and broadcast it again. In other cases, the node will discard this message. This approach increases the probability of discovering multiple disjoint routes at the expense of an increase in control overhead.

As mentioned before, we choose to build the route sets at the destination node, since the destination node knows the entire path of all the available routes. The destination collects multiple copies of RREQ packets of the same session within a time window, then builds the route sets, and returns them to the source.

3.1.1.1 Managing Route Request Packets

The destination node collects RREQ packets after the time window to receive RREQ packets expires. There is a tradeoff between the network size and the time window, i.e. the larger the network is, greater the time window should be. But if we keep increasing the time window, then we would increase the route discovery latency which is not good for multimedia applications. If the time window is too short then the destination may not have received enough RREQ packets, or if the destination issues a RREP packet too soon, it might collide with incoming RREQ packets. In our implementation, we defined a time window of 150ms taking into account the RREQ flooding duration in networks with 10 to 200 nodes. The collection of routes on the RREQ packets received during that time window is sorted according to the following criteria:

- Path disjointedness in relation to other paths
- Number of hops (path length)

If two or more routes in the RREQ packets with node disjoint paths are received, they are sorted according to path length. If non-disjoint paths are received too, they are discarded. A route set is built using all node disjoint routes (primary routes) received. At this point, as we do not have neighborhood information, it is impossible to ensure that the node disjoint routes are non-interfering too. A RREP packet is built for every node disjoint route on the route set and is sent back to the source route through that path. The route set is included in all RREP messages sent. We send RREP packet for all node disjoint paths in the route set because: (1) it is almost impossible to predict which RREP packet will reach the source node and (2) we need to collect as much neighborhood information as possible.

In case there are no node disjointed paths, and more than two RREQ packets were received, only 2 RREP packets are sent. They are chosen trying to minimize the number of common nodes.

3.1.1.2 Managing Route Reply Packets

We consider as expected RREP packets, all RREP packets that are sent by the destination node with the intention of collecting neighborhood information from all nodes along the path. The number of expected RREP packets is equal to the number of routes in the route set. In our implementation, we set a maximum of 32 neighbors that each node can add to a RREP packet. This value can be configured considering the network size. We defined a RREP timeout window of 25ms.

When the source node receives a RREP packet, it waits for the time window for the arrival of other RREP packets. The RREP time window was introduced because: (1) if by the time the first RREP packet was received and the correlation factor was calculated, it would contain only little neighborhood information, (2) by starting to transmit packets with the existing routes, the risk of interference between the incoming RREP packets and outgoing traffic would be greater, causing avoidable link error messages.

After the RREP time window expires, the paths of all received RREP packets are added to the source node route cache and the neighborhood information is added to the source node neighborhood cache. If the expected RREPs are 2 or more and not all were received, we cannot ensure that the paths will not interfere with each other, even if they are node disjoint, as we did not receive all neighborhood information. If we receive RREP packet after the time window expires, the route cache will be updated with the new routes and a new multipath selection process will start.

The route and neighborhood caches are used in the multipath selection process that is explained in section 3.3.

3.1.2 Route Maintenance

Route maintenance is the mechanism by which a node is able to detect any change in the network topology. A link of a route can be disconnected because of mobility, congestion, and packet collision. When a node fails to deliver a data packet to the next hop of the route, it considers the link to be disconnected and sends a Route ERRor (RERR) packet to the upstream direction of the route. Upon receiving this RERR packet, the source removes every entry in its route cache that uses the broken link regardless of the destination. As the route table is changed, a new multipath selection process is initiated. The multipath selection process is explained in section 3.3.

As in [Wei 2004], a new route discovery process is initiated if only one primary route remains in the route cache, so that alternative primary routes may be found before the last primary route fails.

3.2 Neighbor Information Management

In this section we describe how neighborhood information is obtained and maintained by the node. Our procedure has the advantage of not increasing control overhead, but has the disadvantage of not being able ensure that the neighbor information is always updated.

3.2.1 Obtaining neighbor Information

As mentioned before, the route discovery process typically involves a network-wide flood of RREQ packets targeting the destination. Every node has a cache of neighbors. Upon receiving RREQ packets, a node knows which nodes are its direct neighbors and adds them to its neighbors' cache.

As the RREP packets are passed node by node until they reach the source node, all nodes in the path add their neighborhood information.

3.2.2 Maintaining Neighbor Information

As nodes fail, the neighborhood information is updated: (1) upon receiving a RERR packet or (2) if the node tries to forward a packet and the intended node is unavailable.

3.3 Multipath Selection Process

The multipath selection process consists of selecting the most appropriate paths to send the data packets. As in DSR[1], there are 2 route caches, the primary and secondary. The primary stores routes received by RREP packets and the secondary those overheard.

When a route is needed to send data packets, all paths on these two caches for that destination are grouped according to the criteria described on section 3.1.1.1. The pairs are sorted according to the correlation factor. The pair with the smallest correlation factor is selected and its routes are used for multipath routing.

The selected multipath routes are used until: (1) the route cache is updated or (2) a notification of dead link is received. If any of the previous events happens, the routes are discarded and new routes are selected. Figure 3.2 shows a situation where independent node disjoint paths are used, as discovered by our algorithms. Figure 3.3 shows a situation, also generated by our algorithms, where node disjoint paths are used, but the paths are not independent, as some of the intermediate nodes are too close, causing radio interference with each other. Even if node disjointed paths exist, we cannot ensure that the algorithms will use them due to dropped packets.

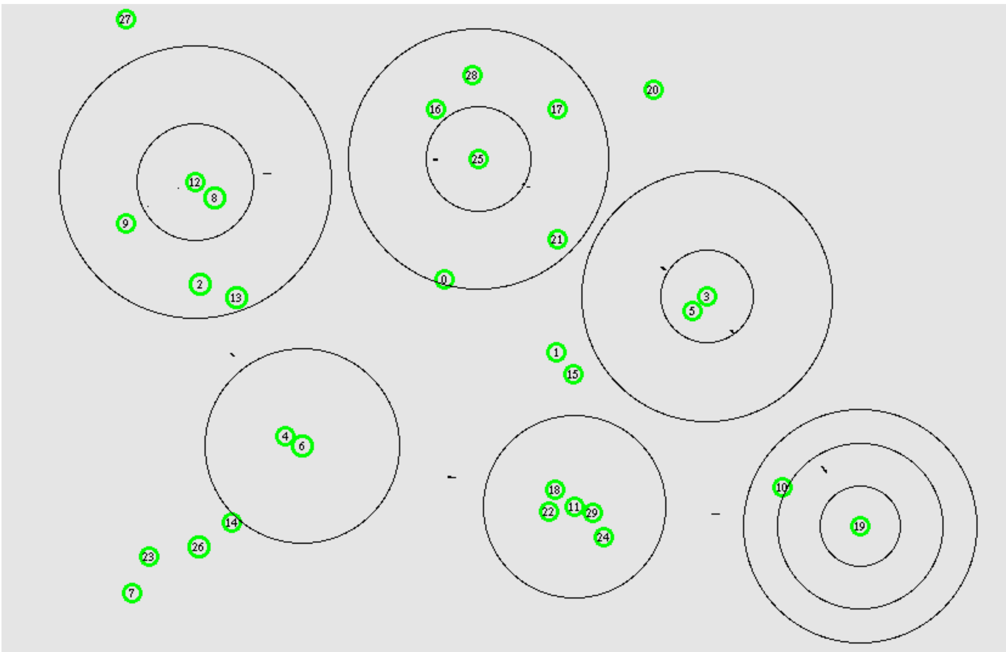


Figure 3.2 The use of independent node disjoint path from the source node 9 to the destination node 19. Nodes on the first path are 9, 12, 25, 3 and 19 and on the other path: 9, 6, 11 and 19.

[1] http://www.isi.edu/nsnam/ns/

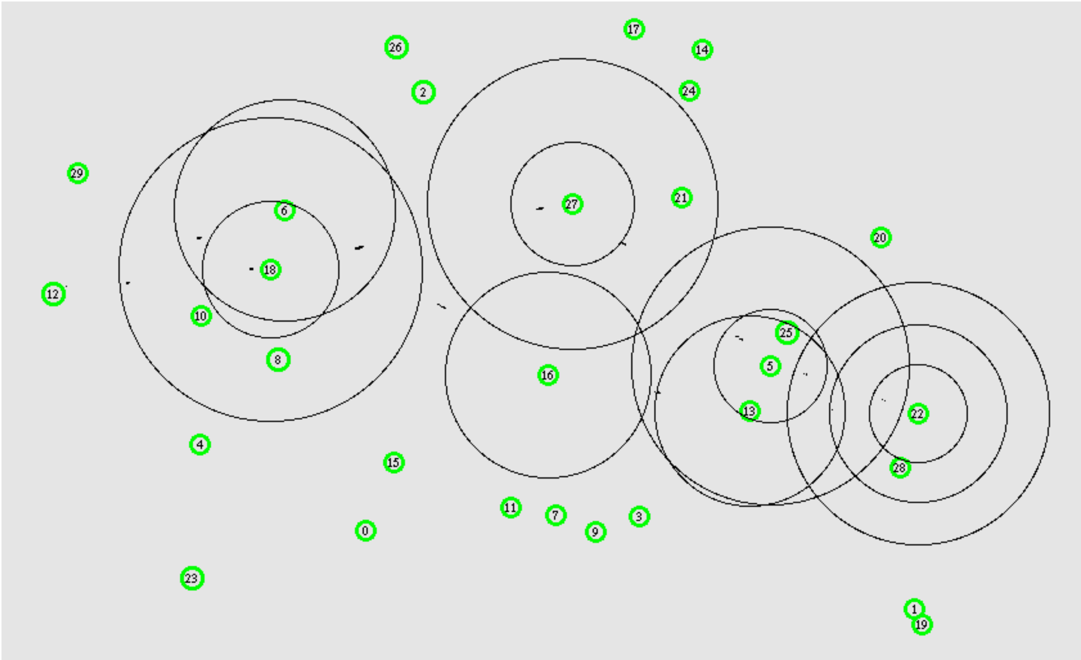


Figure 3.3 The use of node disjoint paths from the source node 12 to the destination node 22. Nodes on the first path are: 12, 6, 16, 13 and 22 and on the other, 12, 18, 27, 5 and 22.

3.4 Route Coupling Technique

As mentioned before, in order to achieve maximum spatial reuse when using multiple independent paths, it is necessary for the traffic on one path to not interfere with the traffic on the other paths, as in the example of Figure 3.2.

According to [Liaw 2004], as the transmission rate at the source is limited by the contention among the first two hops of the path, using two independent paths does not double the achievable throughput. So the maximum spatial reuse is achievable by the source when distributing the load between the two paths, by transmitting 1 packet every two hops, i.e. at times *t*, 3*t*, 5*t*, 7*t*, 9*t*… assuming *t* the necessary time to successfully forward a packet to the next hop node. So, when using two paths, the source transmits at time *t* through the first path, at time 3*t* through the second path, at time 5*t* through the first path and so on. Comparing with a single path, this provides a theoretical improvement of 50% [Liaw 2004].

In our implementation, we consider t_{data} as the average transmission time taken to successfully forward a UDP data packet, and is set to 1.817msec.

3.5 Implementation Decisions

The proposed protocol was implemented in the widely used NS-2[1] simulator. Having been under constant investigation and enhancement for years, NS-2 has become the most widely used open source network simulator, and one of the most used network simulators. It is popular in academia for its extensibility (due to its open source model) and plentiful online documentation. NS-2 is a discrete event simulator targeted at networking research and provides modules for numerous network components such as routing, transport or application. Moreover, it allows the definition of new modules. It was built in C++ and provides a simulation interface through OTcl, an object-oriented dialect of Tcl. The user describes a network topology by writing OTcl scripts, and then the main NS program simulates that topology with the specified parameters.

The five different ad-hoc routing protocols currently implemented for mobile networking in NS-2 are DSDV, DSR, AODV, TORA and PUMA [Fall 2010]. Currently, AOMDV is the only multipath routing protocol implemented for NS-2.

As mentioned in the previous section, the WMSN characteristics must be considered before sending video streams over them. Multimedia traffic can be modeled better using the Pareto distribution [Neame 1999]. It is also important to test the performance of the protocol over constant bit rate traffic, which adequately models multimedia traffic for some codec, such as voice.

According to [Fall 2010], there are four C++ classes derived from the class *TrafficGenerator*. The *POO_Traffic* class generates traffic according to a Pareto On/Off distribution. Packets are sent at a fixed rate during on periods, and no packets are sent during off periods. Both on and off periods are taken from a Pareto distribution. These sources can be used to generate aggregate traffic that exhibits long range dependency. The *CBR_Traffic* class generates traffic according to a deterministic rate, using constant size packets. Optionally, some randomizing dither can be enabled on the interpacket departure intervals.

Performance evaluation of higher layer protocols is affected by a set of factors in the physical and MAC layer. Such factors include signal reception, path loss, fading, interference and noise computation, and preamble length. Computation of interference and noise determines the probability of successful signal reception for a given frame. The power of interference and noise is calculated as the sum of all signals on the channel other than the one being received by the radio plus the thermal (receiver) noise. There are essentially two models of doing this estimation, the Signal to Noise Ratio (SNR) threshold based and the Bit Error Rate (BER) based. The SNR threshold based model uses the SNR value directly by comparing it with an SNR threshold (SNRT), and accepts only signals whose SNR values have been above SNRT at any time during the reception. The BER based model probabilistically decides whether or not each frame is received successfully based on the frame length and the BER resulting from the SNR and modulation scheme used at the transceiver. As the model evaluates each segment of a frame with a BER value every time the interference power changes, it is considered to be more realistic and accurate than the SNR threshold based model. However, the SNR threshold based model requires less computational cost and can be a good abstraction if each frame length is long [Takai 2001].

The default NS-2 radio model does not calculate the noise power as described below, but calculates pseudo SNR values by treating a signal that has arrived prior to the receiving signal to represent the noise on the channel. NS-2 then applies the SNR threshold based model to determine the successful reception of each signal. The default NS-2 radio model has the following disadvantages [Q. S.-E.-M. Chen 2007]:

- Wrong collision handling
- No preamble and PLCP header modeling
- No cumulative SINR implementation
- Wrong back-off handling
- Misusage of the network allocation vector (e.g. EIFS)
- Incomplete support for capture

A team from Mercedes-Benz Research and Development North America and from University of Karlsruhe has collaborated to develop a completely new 802.11 MAC and PHY models, called Mac802_11Ext and WirelessPhyExt, respectively. The new model contains the following features [Fall 2010]:

- Structured design of MAC functionality modules: transmission, reception, transmission coordination, reception coordination, back-off manager, and channel state monitor
- Cumulative SINR computation
- MAC frame capture capabilities
- Multiple modulation scheme support
- Packet drop tracing at the PHY layer
- Nakagami fading model

The only disadvantage of this model is that it does not include an energy model. We developed an energy model for the Ext radio model similar to the existing energy model for the default NS-2 radio model.

Another important factor is the path loss, which defines the average signal power loss of a path. While the free space model is the default model in some simulation platforms, the two-ray model (that is the default in NS-2 simulator) is more realistic [Takai 2001]. For this reason, we decided to use the two-ray model.

Summary

In this chapter we have described MeDSR, a node disjoint multipath routing protocol that uses neighborhood information to calculate non-interfering paths to route traffic. As we have seen, the protocol has two mechanisms, route discovery and maintenance. A new routing algorithm was designed as the solutions we have found in the literature were considered to be not suitable for our goals.

In the following chapter an extensive evaluation of the solution is presented.

4 Evaluation

In this Chapter we describe the experiments that were carried out in order to evaluate the proposed MeDSR protocol and compare it with other routing schemes. To accomplish this goal we executed a series of experiments that compare the performance, signaling costs, latency of MeDSR against single path and multiple paths routing protocols (as it also considers interferences between paths).

We begin by describing the experimental settings and the criteria used in the evaluation. Then we present and discuss the performance, signaling costs, latency of these routing protocols.

4.1 Simulation Model

We use a simulation model based on NS-2[1] [9]. We study the performance of our protocol in comparison with a single path and a multipath routing protocol, DSR and AOMDV respectively. The channel capacity of mobile hosts is set to 6Mbps. All the transmitters have the same transmission range. We use the distributed coordination function (DCF) of IEEE 802.11 for wireless LANs. We use 802.11Ext [10] as the MAC protocol for NS-2.

In our simulation, we consider 60 different network scenarios with 30 nodes randomly distributed over an area of 1500m x 1000m. The source and destination nodes were also randomly selected for each of the 60 simulation scenarios. The simulation duration is of 50s. We use 2 simulation traffic types: Constant Bit Rate (CBR) and Pareto On/Off. The size of the packets is set to 210 bytes. There is only one source/destination traffic pair. For each traffic source, we used the following data rates: 64, 128, 196, 256, 320, 384, 448, 512, 576, 640, 704, 768, 832, 896, 960 and 1024kbps. For Pareto, the burst and idle average time periods are set to 250ms. A packet is dropped when no MAC acknowledgment is received after several retransmissions or there is no buffer to hold the packet. We run all scenarios under the different protocols with the same random seed, and the results were averaged. By using the same seed, we ensured that the traffic source would behave in the same way in all similar simulation scenarios.

Each node contains an initial energy of 1J. Considering a group of 30 nodes, it gives an amount of 30J for all nodes. As nodes participate in the sensor network they use their energy by transmitting or receiving packets or even by being in standby mode. They can only participate in the network if they have energy. Table I shows the simulation parameters.

The values of power consumption in the four radio states, including idle, transmit, receive, and sleep state, are set according to the study results in [Bougard 2005]: P_{idl} = 712 µW, P_{tx} = 31.32 mW, P_{rx} = 35.28 mW, and P_{slp} =144 nW.

Table 4.1 Simulation Parameters

Network field	1500m x 1000m
Number of Sensor	30
Number of Sinks/Number of Sources	1/1
Packet Size	210 bytes
Node Energy	1J
Idle power	712µW
Receive power	35.28mW
Transmit power	31.32mW
Sleep power	144nW
Radio Propagation Model	Two Ray Ground
Source Data rates	64 – 1024kbps
Traffic types	CBR, Pareto On/Off
MAC Layer	IEEE 802.11a
Physical Layer data rate	6 Mbps
Simulation time	50seconds

4.2 Simulation Results

As defined in session 2.5, we evaluate the performance according to the following metrics: throughput, average end-to-end delay, drops, energy consumption, and control overhead.

We evaluate and compare the performance of the following protocols:

- *DSR*: Dynamic Source Routing protocol which uses a single path routing protocol.
- *AOMDV*: Ad hoc On-demand Multipath Distance Vector which uses a multipath routing protocol.
- *MeDSR1*: Normal Multipath Extension to Dynamic Source Routing protocol.
- *MeDSR2*: MeDSR where a 2t delay is added before sending each packet from the source node, as explained in section 3.4.

4.2.1 Throughput

Figure 4.1 and Figure 4.2 show the throughput comparison between DSR, AOMDV, MeDSR1 and MeDSR2 routing protocols using CBR and Pareto traffic.

We can observe that for CBR traffic, DSR, MeDSR1 and MeDSR2 behave similarly until the source rate reaches 320kbps. This is expected as for low data rates, a single path is enough to carry all the traffic. As AOMDV is a multipath routing protocol and it is configured to find node disjoint paths, it behaves similarly to MeDSR1 and MeDSR2 until the source route reaches 576kbps. For Pareto traffic, as the traffic is less intense, as we have pause and burst periods, all protocols behave similarly until we reach a source rate of 512kbps.

As we increase the source rate, we observe that both MeDSR schemes outperform DSR. This is achieved by using simultaneously the transmission resources of multiple paths. For higher source rates, we observe a decrease on the throughput, due to collisions and contention. As for multipath we have two available paths, load can be balanced between them. But as we increase the source rate for the multipath case, the increase of throughput is not linear. As 60 different random scenarios for the 30 nodes configuration were considered, and the nodes were randomly distributed, for some scenarios it was not possible to use multiples paths as they were not available. For other scenarios, multiple paths were used, but they were not independent or in other scenarios, link disjoint paths were used. For all these situations, there was some interference between nodes, decreasing the throughput. Between the multipath routing protocols, as the source rate increases beyond 576kbps for CBR traffic, only MeDSR2 continues increasing its throughput because it tries to avoid packet collision by means of spatial reuse.

Table 4.2 shows the average throughput 95% confidence intervals of the results obtained during the simulation.

Table 4.2 Average Throughput 95% Confidence Intervals

	DSR	AOMDV	MeDSR1	MeDSR2
CBR	60,81	21,14	47,51	53,39
Pareto	21,13	12,10	14,65	15,12

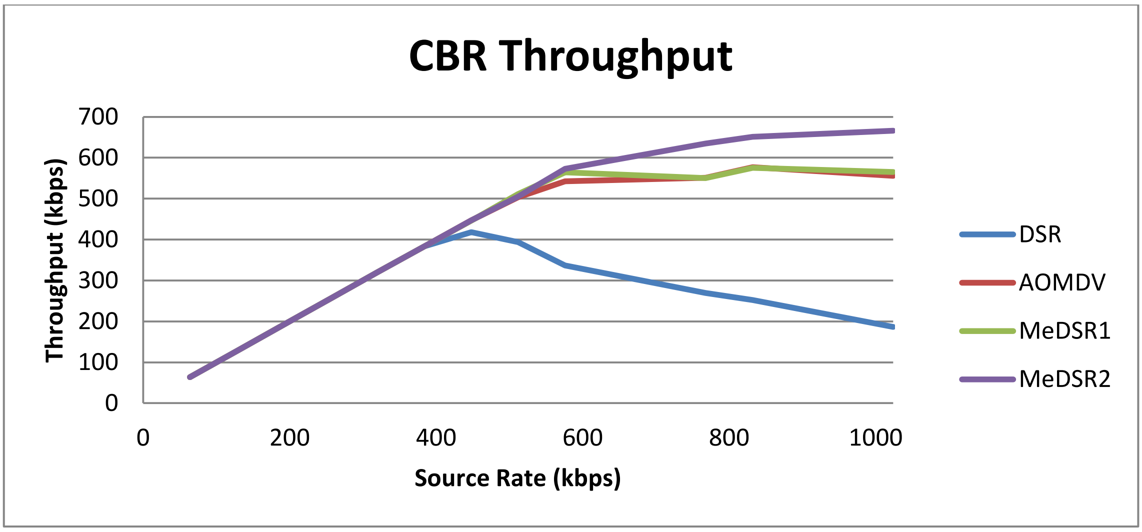


Figure 4.1 Throughput comparison between DSR, AOMDV, MeDSR1 and MEDSR2 using CBR traffic.

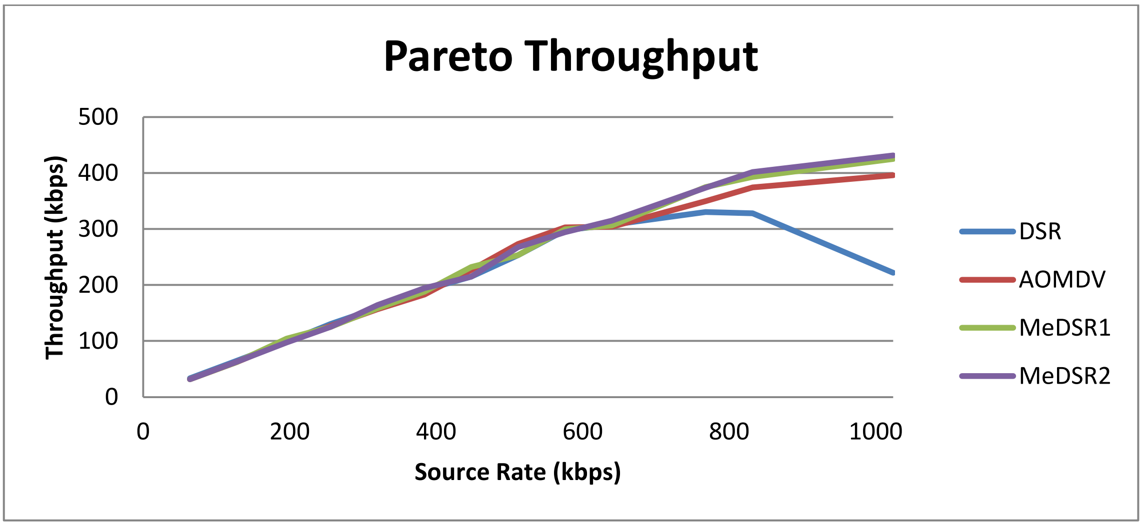


Figure 4.2 Throughput comparison between DSR, AOMDV, MeDSR1 and MEDSR2 using Pareto traffic.

Table 4.3 shows a throughput gain comparison between the routing protocols. As can be seen, MeDSR2 is on average of 5.82% better the MeDSR1 for CBR traffic and 0.66% better for Pareto traffic.

Table 4.3 Throughput Gains Comparison (%)

	CBR		Pareto	
Protocol	**MeDSR1**	**MeDSR2**	**MeDSR1**	**MeDSR2**
DSR	42.29	50.58	12.32	13.07
AOMDV	0.80	6.67	2.35	3.03
MeDSR1		5.82		0.66

4.2.2 Delay

Figure 4.3 and Figure 4.4 show the delay comparison between DSR, AOMDV, MeDSR1 and MeDSR2 routing protocols using CBR and Pareto traffic.

The single path routing protocol behaves better than the multiple path protocols for lower source rates. The single path routing protocol uses only the shortest path between the source and destination nodes, so it minimizes the delay.

As we increase the source rate, the single path delay increases drastically, as the number of collisions due to link saturation becomes significant. For the multipath case, as MeDSR2 adds a delay in the source to reduce collisions, for smaller source rates the delay is slightly higher than for MeDSR1. But as the source rate increases, the delay mechanism from MeDSR2 starts paying back, as it is able to reduce more collisions than MeDSR1.Table 4.4 shows the average delay 95% confidence intervals of the results obtained during the simulation.

Table 4.4 Average Delay 95% Confidence Intervals

	DSR	AOMDV	MeDSR1	MeDSR2
CBR	0.664	0.020	0.042	0.039
Pareto	0.206	0.005	0.009	0.004

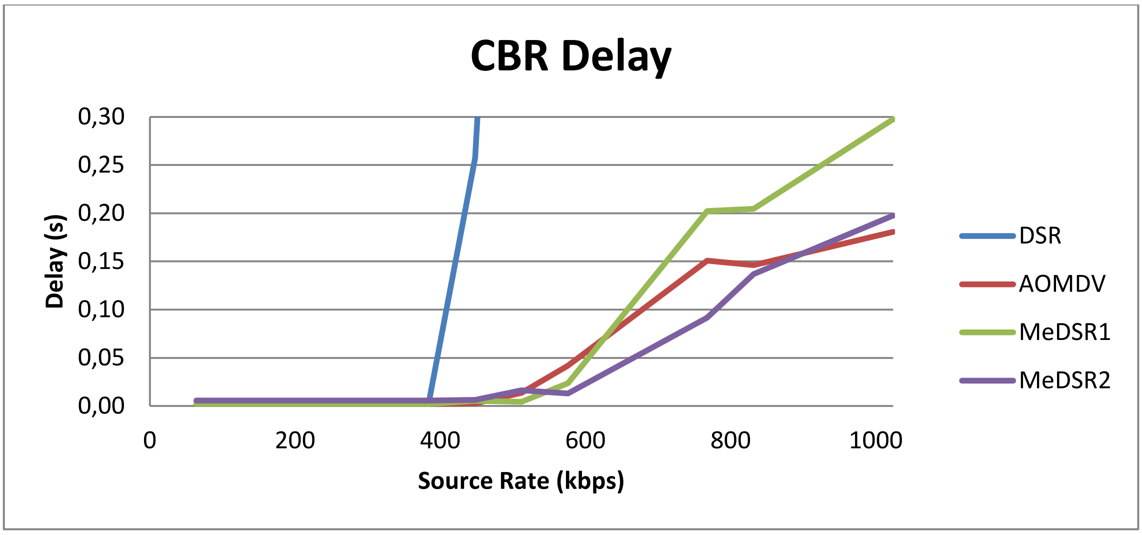


Figure 4.3 Delay comparison between DSR, AOMDV, MeDSR1 and MEDSR2 using CBR traffic.

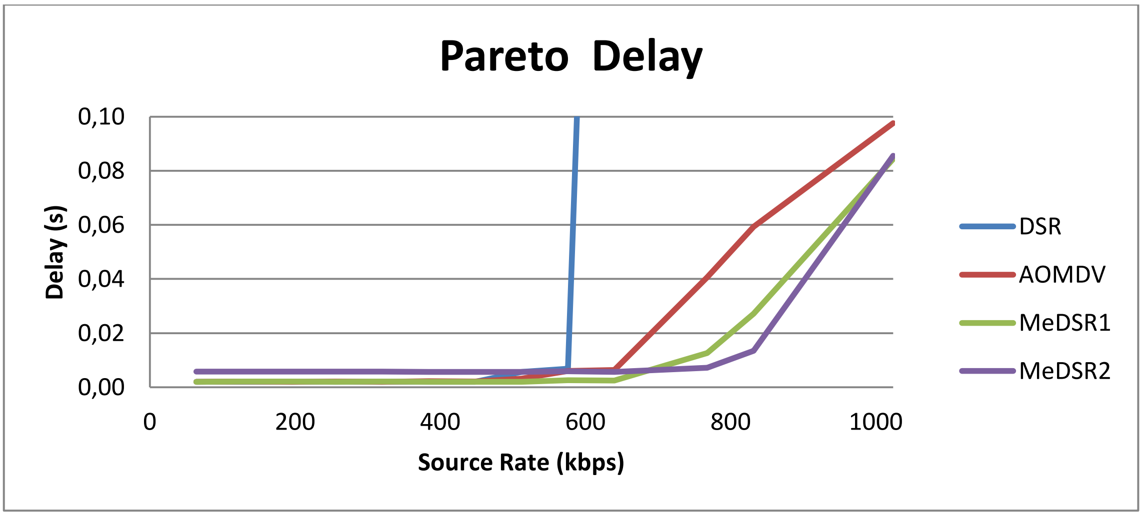


Figure 4.4 Delay comparison between DSR, AOMDV, MeDSR1 and MEDSR2 using Pareto traffic.

Table 4.5 shows the delay gain comparison between the routing protocols. As can be seen, MeDSR2 is on an average of 33.88% better than MeDSR1 for CBR traffic and 12.20% worst for Pareto traffic.

Table 4.5 Delay Gains Comparison (%)

	CBR		Pareto	
Protocol	**MeDSR1**	**MeDSR2**	**MeDSR1**	**MeDSR2**
DSR	93.05	95.40	93.01	92.16
AOMDV	-36.88	9.49	36.02	28.22
MeDSR1		33.88		-12.20

4.2.3 Packet Drop Ratio

Figure 4.5 and Figure 4.6 show the packet drop comparison between DSR, AOMDV, MeDSR1 and MeDSR2 routing protocols using CBR and Pareto traffic. The reasons for packet drops can be collisions and congestion.

By increasing the source rate above a certain threshold, the drop ratio increases too. But for multipath protocols, that increase is not as fast as for single path protocols even considering situations where we have route coupling. If we were only considering scenarios where independent node disjoint paths were used, the only limitation for multipath would be the source and destination nodes, as mentioned in section 3.4.

MeDSR2 outperforms both AOMDV and MeDSR1, as the source delay introduced reduces collisions and, consequently, packet drops. For smaller sources rates in both traffic patterns, our protocol performs better than AOMDV, as the latest does not take into account non interfering paths upon its route discovery process. Routing algorithms for finding node disjoint paths do not ensure non interfering paths, but routing algorithms for finding non interfering paths ensure that those paths are node disjoint and when nodes on each path are distant enough to not interfere with each other.

For CBR traffic, MeDSR1 tends to behave similarly to AOMDV that is also a multipath routing protocol, for higher source rates. This can be concluded also by analyzing the gains it has in comparison with AOMDV. As Pareto traffic is less intense, we obtained more gains than in comparison with CBR.

Table 4.6 shows the average packet drops 95% confidence intervals of the results obtained during the simulation.

Table 4.6 Average Packet Drops 95% Confidence Intervals

	DSR	AOMDV	MeDSR1	MeDSR2
CBR	0.194	0.067	0.095	0.100
Pareto	0.088	0.026	0.031	0.020

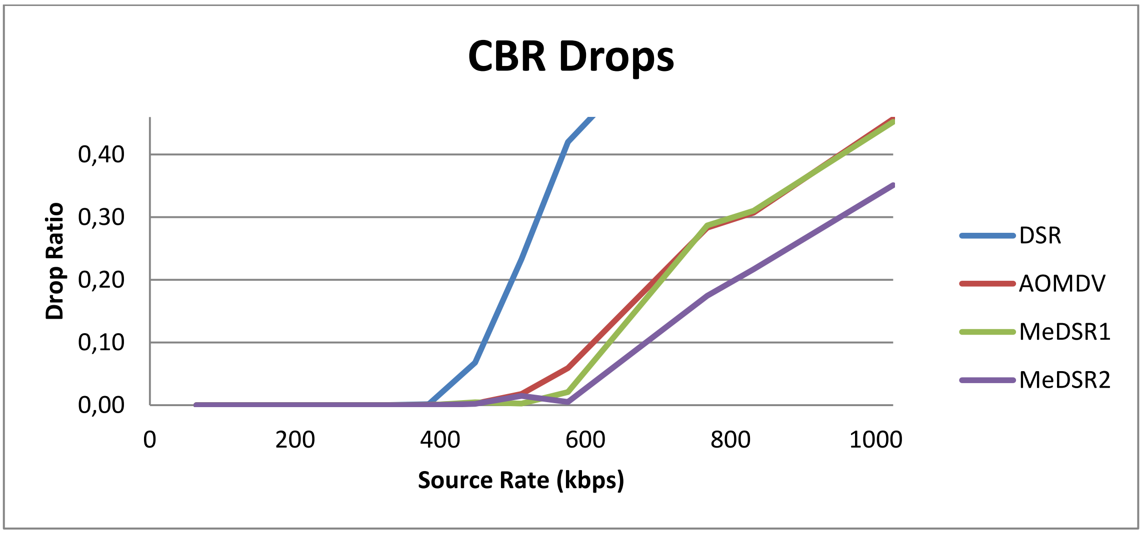


Figure 4.5 Drops comparison between DSR, AOMDV, MeDSR1 and MEDSR2 using CBR traffic.

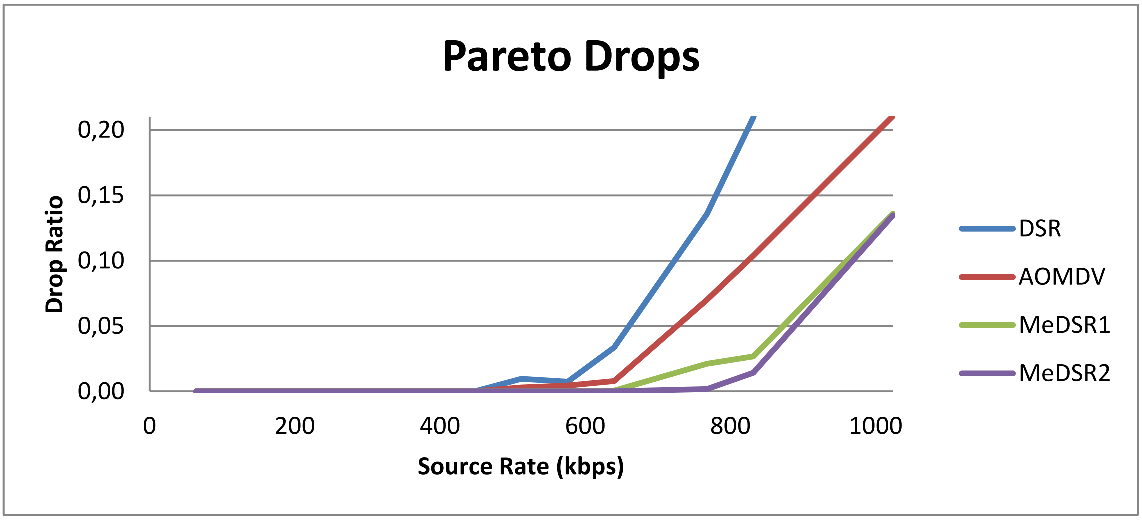


Figure 4.6 Drops comparison between DSR, AOMDV, MeDSR1 and MEDSR2 using Pareto traffic.

Table 4.7 shows the packet drops gain comparison between the routing protocols. As can be seen, MeDSR2 is on average of 29.07% better the MeDSR1 for CBR traffic and 18.35% better for Pareto traffic.

Table 4.7 Packet Drops Gains Comparison (%)

	CBR		Pareto	
Protocol	**MeDSR1**	**MeDSR2**	**MeDSR1**	**MeDSR2**
DSR	62.81	73.62	81.17	84.63
AOMDV	4.43	32.21	53.81	62.29
MeDSR1		29.07		18.35

4.2.4 Control Overhead

Figure 4.7 and Figure 4.8 show the control overhead comparison between DSR, AOMDV, MeDSR1 and MeDSR2 routing protocols using CBR and Pareto traffic.

For single path routing protocols, as we increase the source rate, congestion and collision increases too. As consequence, we have an increase in control overhead as the protocol becomes unable to deliver packets. AOMDV despite normal control messages used during route discovery operations uses too many hello messages to detect link breakages. For smaller source rates, as the number of sent packets is small, the control overhead is considerable high. By increasing the source rate, the number of control packets becomes negligible in comparison to the number of packets forwarded by the nodes.

Our algorithm uses neighborhood information to find non interfering paths as described in section 3.2. As we do not send periodically control messages, our protocol has less overhead even at higher source rates.

Table 4.8 shows the average overhead 95% confidence intervals of the results obtained during the simulation.

Table 4.8 Average Overhead 95% Confidence Intervals

	DSR	AOMDV	MeDSR1	MeDSR2
CBR	3.521	0.155	0.149	0.175
Pareto	2.331	0.325	0.086	0.108

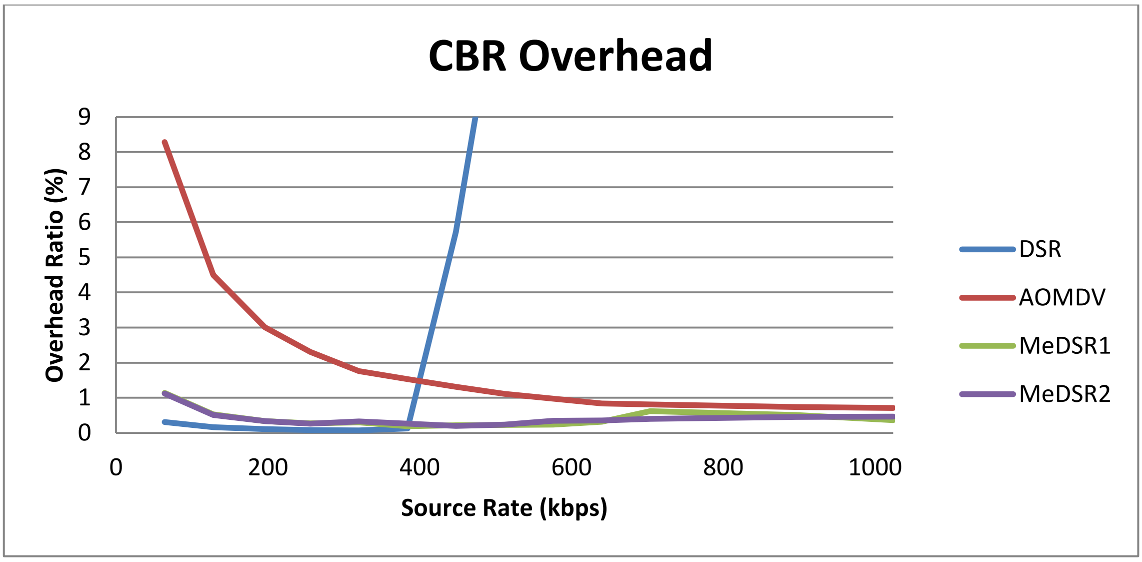


Figure 4.7 Control overhead comparison DSR, AOMDV, MeDSR1 and MEDSR2 using CBR traffic

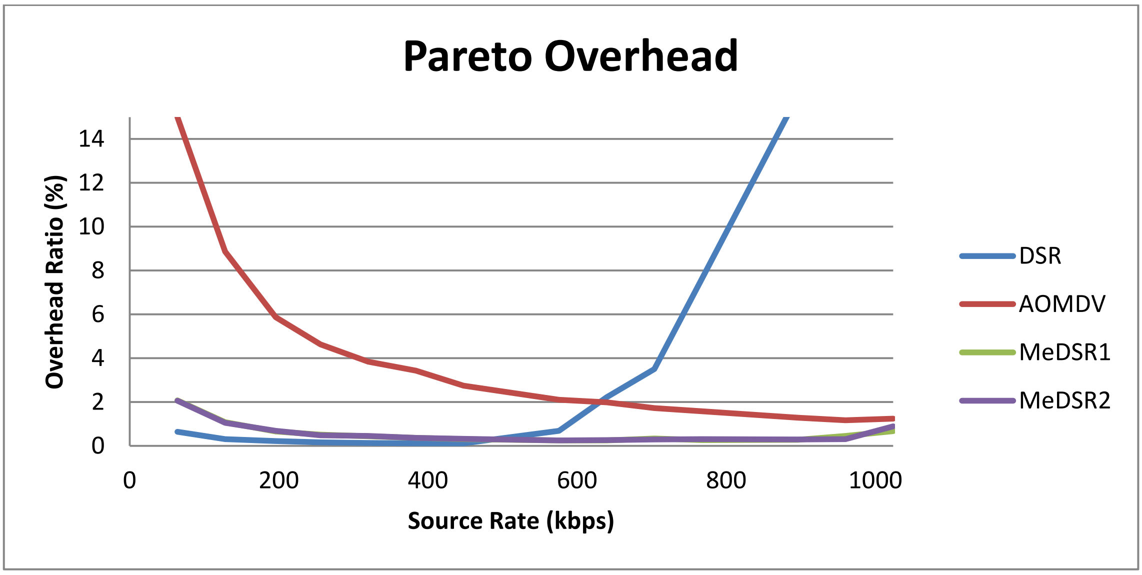


Figure 4.8 Control overhead comparison between DSR, AOMDV, MeDSR1 and MEDSR2 using Pareto traffic

Table 4.9 shows the overhead gain comparison between the routing protocols. As can be seen, MeDSR2 is on average of 0.81% better the MeDSR1 for CBR traffic and 0.99% better for Pareto traffic.

Table 4.9 Overhead Gains Comparison (%)

	CBR		Pareto	
Protocol	**MeDSR1**	**MeDSR2**	**MeDSR1**	**MeDSR2**
DSR	96.98	81.30	90.29	90.19
AOMDV	96.96	81.14	85.74	85.60
MeDSR1		0.81		0.99

4.2.5 Energy Consumption

Figure 4.9 and Figure 4.10 show the energy consumed by all nodes during the entire simulation comparison between DSR, AOMDV, MeDSR1 and MeDSR2 routing protocols using CBR and Pareto traffic.

Due to a higher control overhead, AOMDV consumes more energy than all others. A single path routing protocol, for smaller source rates, consumes less energy, as it deals less with interference than multiple paths routing protocols. As the source rate increases, and with the protocol overhead, more energy is spent with new route discovery processes. In certain scenarios, in order to use non interfering paths, the protocol selects paths with more hops. Using more hops has the disadvantage of spending more energy from the nodes and increasing the end to end delay.

Table 4.10 shows the average energy consumption 95% confidence intervals of the results obtained during the simulation.

Table 4.10 Average Energy Consumption 95% Confidence Intervals

	DSR	AOMDV	MeDSR1	MeDSR2
CBR	0.760	0.403	1.014	1.031
Pareto	0.536	0.338	0.381	0.329

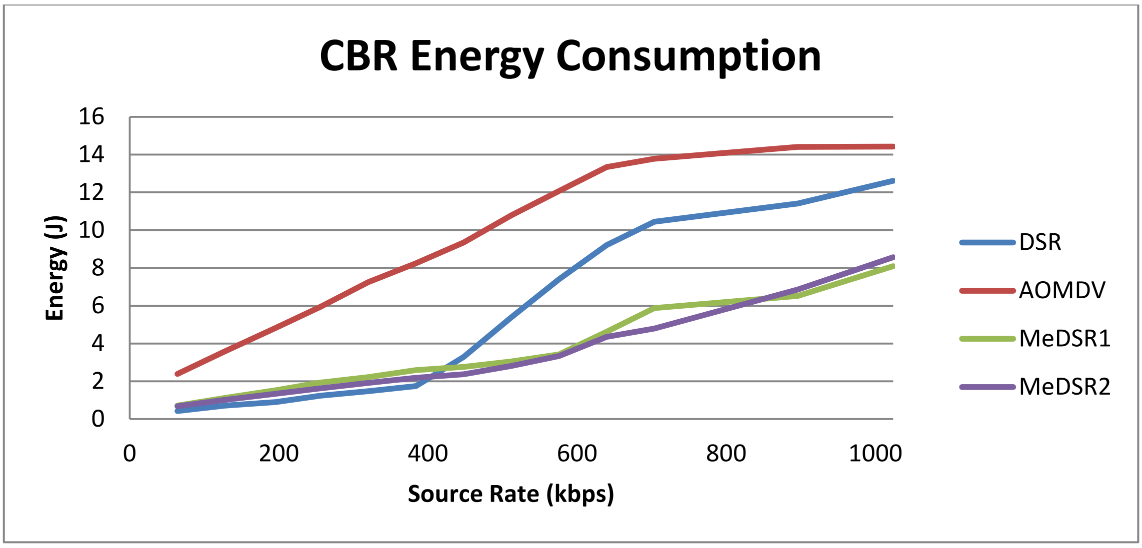


Figure 4.9 Energy consumption comparison DSR, AOMDV, MeDSR1 and MEDSR2 using CBR traffic.

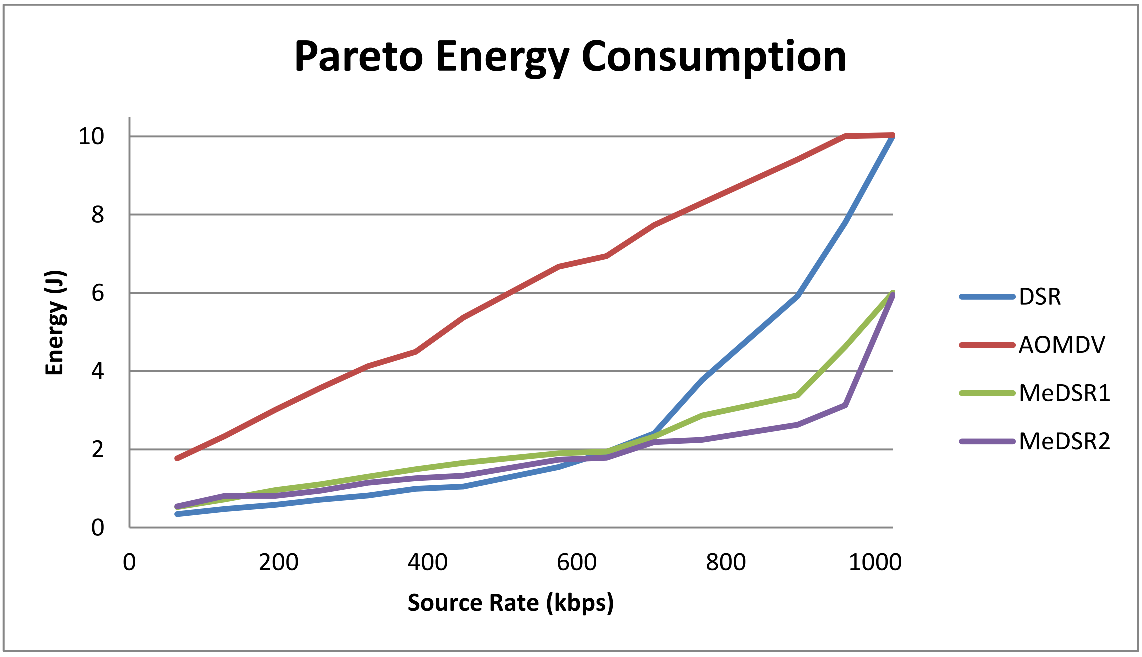


Figure 4.10 Energy consumption comparison between DSR, AOMDV, MeDSR1 and MEDSR2 using Pareto traffic.

Table 4.11 shows the energy consumption gains comparison between the routing protocols. As can be seen, MeDSR2 is on average of 5.79% better the MeDSR1 for CBR traffic and 13.99% better for Pareto traffic.

Table 4.11 Energy Consumption Gains Comparison (%)

	CBR		Pareto	
Protocol	**MeDSR1**	**MeDSR2**	**MeDSR1**	**MeDSR2**
DSR	32.92	36.80	19.73	30.96
AOMDV	63.09	65.22	63.24	68.38
MeDSR1		5.79		13.99

Summary

In this chapter we have experimentally evaluated the operation of MeDSR using simulations. We have compared its performance against a single path (DSR) and a multipath (AOMDV) routing protocol. MeDSR proved to offer an interesting tradeoff between the delay and packet drop ratio as we increase the source rate.

5 Conclusions

5.1 Conclusions

Multipath routing protocols have been used to enhance the performance of Wireless Multimedia Sensor Networks in different ways. Gains can be achieved in throughput, delay, drop ratio, load distribution and energy consumption.

We have developed a new multipath extension to DSR (MeDSR) and studied its performance through simulation. In MeDSR, nodes use the neighbor information they have to help choosing paths with minimal interference between them. This mechanism also avoids some overhead caused by Hello packets as in AOMDV.

As we can see by the experimental evaluation, multipath routing protocols perform better than single path ones over extreme situations, where the network load is high, as for multimedia application scenarios, since they explore the parallel use of transmission resources throughout multiple paths in the network. The average gains of MeDSR as compared with DSR are 31.83% in throughput, 93.78% in delay, 79.12% in packet drops, 85.74% in overhead and 33.88% in energy consumption.

The route coupling issue was also addressed in our work. MeDSR tries to choose the two most disjoint paths with minimal hop length. Adding a delay on the source to interleave packet transmissions, as in the MeDSR2 variant, resulted in a reduction of collisions and an improvement in terms of throughput, and a reduction on delay and packet drop ratio. The experimental evaluation has shown that MeDSR2 outperforms the normal MeDSR by an average of 3.24% in throughput, 10.84% in delay, 23.71% in packet drops, 0.90% in overhead and 9.89% in energy consumption.

The experimental evaluation has shown that MeDSR outperforms AOMDV, as we increase the network load. The average gains of MeDSR2 as compared with AOMDV are 4.85% in throughput, 18.85% in delay, 47.25% in packet drops, 83.37% in overhead and 66.8% in energy consumption.

5.2 Future Work

The current solution is targeted to networks with low mobility, which is the most common situation in Wireless Multimedia Sensor Networks. As future work, we plan on evaluating and improving the stability of routes and protocol overhead in face of failures and node mobility.

Another topic is using Hello messages to keep an updated view of the neighbors and study its overhead increase. By having more updated neighbor information, nodes can decide better which routes to use.

A third topic is using link quality information to choose the best routes. It is expected that Received Signal Strength Indication (RSSI) values may be used to select links with higher data rates in the physical layer, at the cost of reduced radio reach. Naturally, the interference range is the same, the routes will have more hops with more interference between them, which may require and improved packet interleaving mechanism at the source.

A forth topic is using real multimedia traffic to evaluate the protocol performance. By using EvalVid tools-set [Klaue 2003] together with the NS-2[2], perceived quality and objective measure like PSNR calculation can be obtained after network simulation.

A fifth topic is using network coding. By using network coding, the nodes of a network instead of simply relaying the packets of information they receive will take several packets and combine them together for transmission. Redundant information can be sent in alternative paths and used to recover from packet loss without the need for retransmission.

Finally, it would also be interesting to evaluate protocol scalability for larger networks.

[2] The interfacing code between EvalVid and NS-2 is suggested in [Ke 2008]

6 Bibliography

Akkaya, K., Younis, M. "A survey on routing protocols for wireless sensor networks." *Elsevier Ad Hoc Network Journal* 3 [2005]: 325-349.

Akyildiz, I. F., Melodia, T., Chowdhury, K. R. "A survey on wireless multimedia sensor networks." *Computer Networks 51*, 2007: 921–960.

Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E. "Wireless sensor networks: a survey." *Computer Networks 38*, 2002: 393–422.

Al-Karaki, J. N., Kamal, A.E. "Routing techniques in wireless sensor networks: a survey." *IEEE Wireless Communications*, 2004: 6 - 28.

Ayanoglu, E., Chih-Lin, I., Gitlin, R. D., Mazo, J. E. "Diversity coding for transparent self-healing and fault-tolerant communication networks." *IEEE Transactions on Communications* 41 [1993]: 1677-1686.

Baumann, R., Heimlicher, S., Strasser, M., Weibel, A. “A Survey on Routing Metrics.” TIK Report 262, Switzerland, 2007.

Bougard, B., Catthoor, F., Daly, D., Chandrakasan, A., Dehaene, W. “Energy Efficiency of the IEEE 802.15.4 Standard in Dense Wireless Microsensor Networks: Modeling and Improvement Perspectives.” *Design, Autom., and Test in Euro. Conf. and Exhib. (DATE’05).* 2005. 196-201.

Broch, J., Maltz, D., Johnson, D., Hu, Y., Jetcheva, J. “A performance comparison of multi-hop wireless ad hoc network routing protocols.” *ACM MobiCom* . 1998. 85-97.

Chen, Q., Schmidt-Eisenlohr, F., Jiang, D., Torrent-Moreno, M., Delgrossi, L., Hartenstein, H. “Overhaul of IEEE 802.11 Modeling and Simulation in NS-2.” *10th ACM Symposium on Modeling, Analysis, and Simulation of Wireless and Mobile Systems.* NY, USA, 2007. 159-167.

Chen, Y., Nasser, N. “ Enabling QoS Multipath Routing Protocol for Wireless Sensor Networks.” *IEEE International Conference on Communications*, 2008: 2421 – 2425.

Dulman, S., Wu, J., Havinga, P. “An energy efficient multipath routing algorithm for wireless sensor networks.” *6th international symposium on autonomous decentralized systems with an emphasis on advanced distributed transportation systems.* Pisa, Italy, 2003.

Fall, K., Varadhan, K. "The ns Manual (formerly ns Notes and Documentation)." *The Network Simulator ns-2: Documentation.* May 9, 2010. http://www.isi.edu/nsnam/ns/ns-documentation.html [accessed December 2010].

FCC. “Revision of Part 15 of the Commission’s Rules Regarding Ultra-Wideband Transmission Systems. First note and Order.” ET-Docket 98-153, Federal Communications Commission, 2002.

Girod, B., Aaron, A., Rane, S., Rebollo-Monedero, D. “Distributed video coding.” *IEEE.* 2005. 71–83.

Harish, A. R., Garigala, S., Raman , B., Gupta, P. “Feasibility study of spatial reuse in an 802.11 access network.” *XXVIII URSI General Assembly.* New Delhi, India, 2005.

Hedetniemi, S., Liestman, A. “A Survey of Gossiping and broadcasting in Communication Networks.” *IEEE Network* 18, n.º 4 [1988]: 319–349.

Heinzelman, W., Kulik J., Balakrishnan, H. "Adaptive Protocols for Information Dissemination in Wireless Sensor Networks." *5th ACM/IEEE Mobicom.* Seattle, WA, 1999. 174–85.

Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H. “Energy-efficient communication protocol for wireless microsensor networks.” *33rd Annual Hawaii International Conference on System Sciences.* Cambridge, MA, 2000. 1–10.

Intanagonwiwat, C., Govindan, R., Estrin, D. “Directed diffusion: a scalable and robust communication paradigm for sensor networks.” *ACM Mobi-Com’00.* Boston, MA, 2000. 56–67.

Johnson, D., Maltz, D., Broch, J. *The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4.* RFC 4728 (Experimental), IETF, 2007.

Ke, C.-H.,Shieh, C.-K., Hwang, W.-S., Ziviani, A. “An Evaluation Framework for More Realistic Simulations of MPEG Video Transmission.” *Journal Of Information Science And Engineering* 24 [2008]: 425-440.

Klaue, J.,Rathke, B. , Wolisz, A. “EvalVid - A Framework for Video Transmission and Quality Evaluation.” *13th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation.* Urbana, Illinois, USA, 2003.

Krishnan, R., Silvester, J. A. “Choice of allocation granularity in multipath source routing schemes.” *INFOCOM '93.* 1993. 322 - 329.

Lee, S.-J., Gerla, M. “Split Multipath Routing with Maximally Disjoint Paths in Ad Hoc Networks.” *IEEE International Conference on Communications.* Helsinki, 2001. 3201–3205.

Liaw, Y. S., Dadej, A., Jayasuriya, A. “Throughput Performance of Multiple Independent Paths in Wireless Multihop Network.” *IEEE International Conference on Communications.* SA, Australia, 2004. 4157 - 4161.

Liu, C., Kaiser, J. *A Survey of Mobile Ad Hoc network Routing Protocols.* Magdeburg, Germany: University of Ulm Tech. Report Nr. 2008-08, 2005.

Mao, S., Lin, S., Panwar, S. S., Wang, Y., Celebi, E. “Video Transport over Ad Hoc Networks: Multistream Coding with Multipath Transport.” *IEEE Journal on Selected Areas in Communications* 21, n.º 10 [2003]: 1721–37.

Mao, S., Lin, S., Wang , Y., Panwar, SS., Li, Y. “Multipath video transport over ad hoc networks.” *IEEE Wireless Communications* 12 [2005]: 42 - 49.

Marina, M. K., Das, S.R. “On-demand multipath distance vector routing in ad hoc networks.” *Ninth International Conference on Network Protocols.* 2001. 14 - 23.

Montenegro, G., Kushalnagar, N., Hui, J., Culler, D. *Transmission of IPv6 Packets over IEEE 802.15.4 Networks.* RFC 4944 (Standards Track), IETF, 2007.

Mueller, S., Ghosal, D. “Multipath Routing in Mobile Ad Hoc Networks: Issues and Challenges.” *Lecture Notes in Computer Science* 2965 [2004]: 209 – 234.

Nasipuri, A., Das, S. “On-demand multipath routing for mobile ad hoc networks.” *Eight International Conference on Computer Communications and Networks.* Boston, MA, 1999. 64 - 70.

Neame, T.D., Zukerman, M., Addie, R.G. “Application of the M/Pareto Process to modeling broadband traffic streams.” *IEEE International Conference on Networks.* Brisbane, 1999. 53-58.

Pearlman, M. R., Haas, Z. J., Sholander, P., Tabrizi, S. S. "On the impact of alternate path routing for load balancing in mobile ad hoc networks." *1st ACM international symposium on Mobile ad hoc networking & computing.* Boston, MA: IEEE Press, 2000. 3 - 10.

Perkins, C., Belding-Royer, E., Das, S. *Ad hoc On-Demand Distance Vector (AODV) Routing.* RFC 3561 (Experimental), IETF, 2003.

Shu, L., Zhou, Z.B., Hauswirth, M., Phuoc, D.L., Yu, P., Zhang, L. “Transmitting Streaming Data in Wireless Multimedia Sensor Networks with Holes.” Oulu, Finland, 2007.

Singh, C.P., Vyas, O.P., Tiwari, M.K. “A Survey of Simulation in Sensor Networks.” *International Conference on Computational Intelligence for Modeling Control & Automation.* Vienna, 2008. 867 – 872.

Slepian, D., Wolf, J. “Noiseless coding of correlated information sources.” *IEEE Transactions on Information Theory* 19, n.º 4 [1973]: 471–480.

Takai, M., Martin, J., Bagrodia, R. “Effects of wireless physical layer modeling in mobile ad hoc networks.” *2nd ACM international symposium on Mobile ad hoc networking & computing.* NY, USA, 2001.

Tarique, M., Tepe, K. E.,Adibi, S., Erfani, S. "Review: Survey of multipath routing protocols for mobile ad hoc networks." *Network and Computer Applications 32*, 2009: 1125 - 1143.

Tsai, J., Moors, T. "A Review of Multipath Routing Protocols: From Wireless Ad Hoc to Mesh Networks." *ACoRN Early Career Researcher Workshop on Wireless Multihop Networking.* 2006. 17-18.

Tsirigos, A., Haas, Z. J. "Multipath routing in the presence of frequent topological changes." *IEEE Communications Magazine* 39 [2001]: 132-138.

Wei, W., Zakhor, A. "Robust multipath source routing protocol (RMPSR) for video communication over wireless ad hoc networks." *IEEE International Conference on Multimedia and Expo.* Taipei, 2004. 1379 - 1382.

Wu, K.,Harms, J. "On-Demand Multipath Routing for Mobile Ad Hoc Networks." *European Personal Mobile Communications Conference EPMCC.* Vienna, 2001.

—. "On-Demand Multipath Routing for Mobile Ad Hoc Networks." *European Personal Mobile Communications Conference EPMCC.* Vienna, 2001.

Wyner, A., Ziv, J. "The rate-distortion function for source coding with side information at the decoder." *IEEE Transactions on Information Theory* 22 [1976]: 1–10.

Xiong, Z., Liveris, A.D., Cheng, S. "Distributed source coding for sensor networks." *IEEE Signal Processing Magazine* 21 , n.º 5 [2004]: 80–94.

Yao, Z., J. Jiang, P. Fan, Z. Cao, e V.O.K. Li. "A neighbor-table-based multipath routing in ad hoc networks." *The 57th IEEE Semiannual Vehicular Technology Conference.* 2003. 1739 - 1743.

Zuniga, M., Krishnamachari, B. *Integrating future large scale sensor networks with the Internet.* USC Computer Science Technical Report CS 03-792, 2003.