TECHNISCHE
UNIVERSITÄT
WIEN

VIENNA
UNIVERSITY OF
TECHNOLOGY

DIPLOMARBEIT

Wireless Communication in Home and Building Automation

ausgeführt am Institut für

Rechnergestützte Automation

der Technischen Universität Wien

unter Anleitung von

Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Wolfgang Kastner

und

Univ.-Ass. Dipl.-Ing. Georg Neugschwandtner

durch

Christian Reinisch

Franzensgasse 5/43
1050 Wien

Wien, am 13.02.2007

Abstract

The use of wireless technologies in home and building automation (HBA) systems offers attractive benefits, but also introduces a number of technological challenges. Today, mostly wired automation installations exist. In the future, the challenge has to be seen especially in augmenting these existing installations with wireless technology to hybrid systems.

This thesis gives an overview of requirements and challenges characteristic for the HBA domain that come along with the employment of wireless systems. A comparative discussion of wireless protocols suited for the use in HBA is done.

Next, two case studies showing possible integration approaches of wireless technology into an existing wired automation system are presented. Tunneling devices allow enhancement of a wired KNX network with wireless IEEE 802.15.4/ZigBee nodes. Their design is geared towards zero-configuration and supports easy integration of security mechanisms.

The approaches reveal that communication could be optimized with the use of multicast communication. Hence, different multicast protocols are surveyed and compared afterwards. Additionally, they are classified according to criteria identified during research. Finally, a multicast enhancement for the ZigBee protocol, based on a mapping of the Dynamic Core Multicast Protocol to ZigBee, is proposed.

Kurzfassung

Den Vorteilen des Einsatzes von drahtlosen Netzwerken in der Heim- und Gebudeautomation stehen einige technologische Herausforderungen gegenber. Da heute fast ausschlielich kabelgebundene Systeme eingesetzt werden, wird in nherer Zukunft hauptschlich die Erneuerung und Aufwertung dieser Systeme mit Funktechnologie von Interesse sein. Die aus der Kombination resultierenden Hybrid-Systeme erfordern umfassendes Wissen des Systemdesigners aus beiden zugrundliegenden Technologien.

Die Diplomarbeit fasst die Anforderungen und Probleme des Einsatzes von Funknetzwerken in der Heim- und Gebudeautomation zusammen. Im Anschlu werden wichtige Funkprotokolle diskutiert und gegenbergestellt.

Ausgehend von den vorgestellten Protokollen werden zwei Designkonzepte ausgearbeitet, die eine mgliche Integration des IEEE 802.15.4/ZigBee Protokolls in ein bestehendes KNX Automatisierungssystem zeigen. Das Design der dazu ntigen Tunneling-Gerte ist konfigurationsfrei und erlaubt darberhinaus die schnelle Integration von Security Methoden.

Obwohl voll funktionsfhig, arbeiten die vorgestellten Tunneling-Gerte nicht optimal in Hinblick auf Nachrichten Overhead, der mit Hilfe von Multicast Kommunikationsschemata erheblich reduziert werden kann. Aus diesem Grund werden verschiedene Multicast Protokolle analysiert und verglichen. Zustzlich werden die besprochenen Protokolle in eine neu erstellte Klassifizierung eingeteilt. Zum Abschlu wird ein spezieller Multicast Algorithmus (Dynamic Core Multicast Protocol) fr den Einsatz in einem ZigBee Netzwerk modifiziert.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Home and Building Automation systems are used more and more frequently in our times. On the one hand, they provide increased comfort especially when employed in a private home. On the other hand, automation systems installed in commercial buildings do not only increase comfort, but also allow centralized control of heating, ventilation, air condition and lighting. Hence, they contribute to an overall cost reduction and also to energy saving – which is certainly a main issue today.

Existing, well-established systems are based on wired communication. Examples include BACnet, LonWorks and KNX. Employing a wireless system does not pose a problem as long as the system is planned before and installed during the physical construction of the building. If, however, already existing buildings should be augmented with automation systems, this requires much effort since cabling is necessary. Obviously, wireless systems can come to help here.

In the past few years, wireless technologies reached their breakthrough. Wireless based systems, used everyday and everywhere, range from wireless home networks and mobile phones to garage door openers. In course of this development, also wireless technologies suited for the use in home and building automation have emerged. Various different systems have been proposed, some derived from existing wired protocols, but all targeting different portions of the automation market.

As of today, little comparative research of wireless automation standards has been done, although such knowledge would provide valuable information to everyone looking for the must suitable system for given requirements.
As it is valuable to know the key features of wireless protocols, it is at least equally important to analyze the combination of wireless protocols with "traditional" wired installations. Especially the augmentation of existing wired systems with wireless technology will play an important role in the future. Here, it is of particular interest if and how established features of the wired protocols can be mapped to hybrid (i.e., wired combined with wireless) systems.

This thesis is organized as follows. In Chapter 2, first definitions and basic terms of the home and building automation domain are described. Then, benefits, requirements and solutions of wireless home and building automation systems are outlined. Chapter 3 gives an overview of 5 major wireless protocols and lists their key features. Based on a general description, Chapter 4 features two case studies that show how wireless systems can be used in practice. Chapter 5 deals with multicast communication in general and specific multicast protocols, which can be employed to improve wireless communication considerably. In Chapter 6, the multicast algorithm specified by the ZigBee protocol is discussed and, finally in Chapter 7, an improved multicast algorithm for ZigBee is proposed.

Chapter 2

Wireless Sensor Networks in HBA

Today, wireless sensor networks gradually become more and more common in home and building automation. Applications range from monitoring room temperatures to security critical applications such as access control. The following section gives an overview of important aspects. In literature [9, 12, 11], many different definitions of $mobile\ ad-hoc\ networks$, $wireless\ mesh\ networks$ and $wireless\ sensor\ networks$ exist. However, a sharp separation cannot be upheld, as mostly hybrid approaches exist. For this reason this thesis presents aspects of wireless networks that may be more or less distinct in different protocols.

2.1 Aspects

- Ad-hoc

  A wireless ad-hoc network is a computer network with wireless links between the network nodes. The addition $ad-hoc$ means that connections are established when needed and only as long as needed without relying on a fixed infrastructure. For successful establishment of communication channels, it is necessary that nodes that are part of the network are capable and willing to forward data on behalf of other nodes. Moreover, ad-hoc networks usually get by with minimum configuration and can thus be deployed quickly.

- Mobility

  A mobile ad-hoc network is a dynamic wireless network, where the nodes are assumed to move arbitrarily. In order to keep connections alive, the mobile ad-hoc network has to be self-configuring and self-healing. The topology of a MANET is not specified and can change during operation.

- Mesh Networking

  A wireless mesh network is a wireless network where multiple (redundant) paths between nodes exist. If all nodes are connected directly (without intermediary nodes) to each other, the network is referred to have a *full mesh topology*. In case that not all nodes are directly connected to each other (i.e., some nodes are connected only to some other nodes), the mesh is said to have a *partial mesh topology*. In a wireless mesh network, data is forwarded hop-by-hop in accordance with the employed ad-hoc routing protocol. Since there are multiple paths between sender and receiver, a mesh network can be self-healing.

- Wireless Sensor Network

  Haenselmann [15] gives a definition of sensor networks: "A sensor network is a set of small autonomous systems, called sensor nodes which cooperate to solve at least one common application. Their tasks include some kind of perception of physical parameters."
  Enhancing this definition, *wireless* sensor networks (WSN) describe a sensor network connected via wireless links [16]. It consists of multiple devices that are capable of monitoring environmental conditions. For this purpose, the wireless nodes are equipped with sensors. The origin of wireless sensor networks can be found in military applications,

where these networks have been (and still are) used for battleground monitoring.

Wireless nodes typically consist of a processor (with limited computational power and storage), radio communication equipment and attached sensors. Since the nodes are wireless, power supply is only possible through the use of batteries making energy a scarce resource.

- Home and Building Automation

  The term Home and Building Automation (HBA) describes the combination of automation systems in both private homes and commercial buildings.

  An automation network consists of devices that monitor and control technical systems in a building. The building automation system aims at improving control, monitoring and administration of these systems. The motivation for the deployment can be found in economic benefit and improved control. The core domain of building automation systems is lighting control, and heating, ventilation and air conditioning (referred to as *HVAC*). However, today also all-in-one solutions including security monitoring and safety systems exist [17, 18].

  Considering home automation, the focus is rather on increased comfort than economic benefit. Hence, home automation systems often include control of home entertainment systems and configuration of the living space. Another main difference between home and building automation is the design of the human-computer interface. In home automation the ideal system has to be self-explanatory and easily usable and configurable for all people while building automation systems are mostly operated by specially trained professionals. Nevertheless, also in the latter case usability has to be provided to the prospective end-users of the system.

2.2 Benefits

In recent years, wireless systems like WLAN have become more and more common in home networking. Also in home and building automation systems, the use of wireless technologies offers several advantages that could not be achieved using a wired network only.

- Reduced installation costs

  First and foremost, installation costs are significantly reduced since no cabling is necessary. Wired solutions require cabling, where material as well as the professional laying of cables (e.g. into walls) is expensive.

- Easy placement and coverage

  Wireless nodes can be mounted almost anywhere. In adjacent or remote places, where cabling may not be feasible at all, e.g., a garden house or the patio, connection to the home network is accomplished instantly by simply mounting nodes in the area. Hence, wireless technology also helps to enlarge the covered area.

- Easy extension

  Deploying a wireless network is especially advantageous when, due to new or changed requirements, extension of the network is necessary. In contrast to wired installations, additional nodes do not require additional cabling which makes extension rather trivial. This makes wireless installations a seminal investment.

- Aesthetical benefit

  As mentioned before, placement of wireless nodes is flexible. Apart from covering a larger area, this attribute helps to fulfill aesthetical requirements as well. Examples include representative buildings with all-glass architecture and historical buildings where design or conservatory reasons do not allow laying of cables.

- Integration of mobile devices

  With wireless networks, associating mobile devices such as PDAs and Smartphones with the automation system becomes possible everywhere and at any time, as a device's exact physical location is no longer crucial for a connection (as long as the device is in reach of the network). Typical examples include an engineer who connects to the network, performs a particular management task, and disconnects after having finished the task; or control of blinds using a remote control.

For all these reasons, wireless technology is not only an attractive choice in renovation and refurbishment, but also for new installations.

2.3 Requirements

Home and building automation systems have different requirements on the underlying technology than industrial automation systems. While industrial environments may require shielding against electromagnetic interference, aesthetical concerns will probably prevail in houses and office buildings. Hence, choosing the most appropriate technology goes along with good knowledge of the requirements. The following list gives an overview of the most important requirements of this field.

- Low cost per node / High node count

  Thinking of building automation, hundreds of nodes may be needed to provide automation. However, the market requires competitive performance (compared to wired networks) to be delivered at this low system cost. Additionally, also protocols need to scale to high node count e.g., ensuring message delivery.

- High battery lifetime

  Going wireless adds another constraint. For maximum benefit, all wires have to be cut – including power wires. Due to the high node count in the system, having to change or charge the batteries of each wireless device every few days is not feasible. Of course, in order to reach the goal of energy saving, will also have effects on the protocol design.

- Large area coverage

  Another challenge lies in the fact that devices of a building automation system are dispersed over large areas. Since transceivers must not consume too much power, they cannot be built with a transmission range sufficient for sensors to reach associated controllers or actuators directly. Also, they cannot rely on an infrastructure of access points and a wired backbone network (or particularly sensitive receivers) for reasons of cost.

- Low data throughput and mostly high latency

  Regarding the performance criteria of data throughput and latency, building automation applications have relaxed requirements. Since HVAC control has to deal with high system inertia anyway, the only notable exception regarding latency is open loop lighting control.

2.4 Approaches

As explained above, employing wireless networks comes along with special requirements. This section provides an overview of possible solutions.

In networks with a high node count every single node has to be as cheap as possible to make the investment sensible. To reduce cost, nodes are designed in a way so that they provide just enough storage and computational power to fulfill their purpose. Protocols with lower requirements contribute to cheaper nodes. With increased prevalence of wireless networks, node can also be manufactured in higher quantities, again making single nodes cheaper for end users.

In order to achieve battery lifetimes of at least several months, measures in both hard- and software must be taken. The goal of minimizing power consumption also affects the design of the communication protocol. For example, it has to allow nodes to enter power-saving sleep modes as often as possible. It could even allow sensor nodes entirely without radio receivers. As another approach, the deployment of battery-free nodes has to be considered. Based on so-called energy harvesting technologies, these nodes collect enough energy from the environment (using e.g., solar cells and piezo elements) to be able to communicate with other nodes. This technology could solve the problem of power supply entirely.

The high node count of building automation systems comes to help when talking about large area coverage that has to be achieved with radio-power limited nodes. A high node count allows to employ mesh networking schemes. Using such schemes, nodes that are not in direct reach of their communication partner receive its messages through message forwarding from other nodes. This has two added benefits. First, redundancy is provided, i.e., if a single device fails, communication can be upheld through redundant paths (which do not have to be pre-established at installation time). Second, nodes need no longer have enough transmitting power to reach all other nodes in the network, thus allowing production of cheaper and less power-consuming nodes.

Low latency and high data throughput are no key requirements in home and building automation systems. However, if required by the application, protocols can provide both through the use of fixed transmission slots that guarantee collision free communication.

2.5 Interference

From the current point of view, interference poses the biggest problem for wireless automation installations. Because of the nature of the wireless medium, the communication channel is always open for other users as well, which inevitably leads to problems. Interference of wireless systems can be classified into two main parts:

2.5.1 Unintentional

- Next door installations

  Next-door installations using the same protocol obviously can interfere with each other. While most wireless protocols specify different address spaces for neighboring installations, only some also offer system designers the choice of different frequency bands in order to completely separate the networks. Nevertheless, interference with next-door installations is only a small part of the problem.

- Installations using the same frequency band

  Apart from wireless protocols for home and building automation, a variety of wireless technologies from garage door openers to wireless presenters and WLAN access points are competing for access to the medium, all using different access control strategies. Especially in the license-free ISM (Industrial, Scientific, Medical) frequency bands, devices such as microwave ovens, which create radio frequency (RF) emissions merely as a by-product of their intended use, can pose a problem. Thus, a wireless network node operating in an ISM band is much more likely to find its channel jammed than a wired one. This especially has to be taken into account for safety related applications as operating on an open medium has implications for communications security as well.

2.5.2 Intentional Interference

Using a wireless network, security attacks such as eavesdropping and replaying no longer require access to a medium buried within walls or ceilings. Attackers now can take over unsecured systems without ever having entered the building. As an additional difficulty, protocol security features such as cryptographic algorithms are limited by the requirement of low power consumption in the nodes – a limitation the attacker does not face.

2.5.3 Approaches

To minimize unintentional interference, wireless applications should select a frequency band whose regulations best match their communication characteristics. The maximum allowable transmission power and duty cycle are key parameters here. Also, robust modulation and transmission techniques can for example spread the signals over a larger part of the available frequency spectrum, reducing the effects of narrow band interference. These measures must be complemented by appropriate protocol design on higher layers. This includes methods like acknowledged transmissions or automatic retransmission to increase the reliability of transmissions and the automatic choice of less crowded channels defined by the specification.

To counter intentional interference, especially security critical applications like surveillance, access control, and alarm systems require protocol support for authentication, encryption, message integrity, and replay protection. However, all this must be achieved in parallel to meeting the requirement of low per-node costs.

2.6 WLAN, Bluetooth and HBA

Wireless HBA networks are control networks, consisting of wireless sensors, actuators and controllers. The fact that the amount of data to be transmitted is very small in most applications (for open loop control of lighting or HVAC, the transmission of only a few bytes of control data is sufficient) has to be reflected in the choice of a wireless protocol as well. Considering this, popular contenders, in particular Wireless LAN [19] and Bluetooth [20], are ruled out immediately for several reasons.

First of all, neither WLAN nor Bluetooth can support the required battery lifetime since the protocols do not provide any or only rudimentary methods for power conservation such as sleep modes. Both wireless protocols also do not reach the required area coverage without resorting to wired access points – a drawback not acceptable for completely wireless installations. At first glance, the network stacks also contain some useful features like security methods which could be used also in HBA. But these features, on the other hand, make it hard to reach the goal of low cost per node due to their high complexity. Additionally, WLAN and Bluetooth operate in the 2.4 GHz ISM band which allows them to support data rates required for media streaming. As outlined before, HBA applications get by with low throughput so that the media streaming capability of WLAN and Bluetooth stands in contrast to the desired simple and power-saving wireless protocol for HBA. Moreover, low throughput enables the use of lower frequencies (such as the 868MHz ISM band in the EU), which have the advantage of better radio wave propagation per amount of power spent.

Chapter 3

Wireless protocols for HBA

In the following, a selection of wireless control networking technologies applicable in home and building automation is presented. The selection criteria include frequency bands, data rates, modulation techniques, routing schemes, topologies, interoperability, openness, standardization and special features with the focus set on free availability of (parts of) the protocol specification and general suitability to support HBA applications.

3.1 Z-Wave

The Z-Wave protocol [21, 22] was developed with an explicit focus on home control applications. Z-Wave operates at 908 MHz +/- 12kHz in the US and the ISM band of 868 MHz in Europe, using FSK (frequency shift keying) modulation. The RF data rate is 9.6 kbit/s (with a raise to 40 kbit/s advertised). A single network may contain up to 232 devices. Higher counts can only be obtained by bridging networks.

Z-Wave uses a mesh networking approach with source routing, which means that the whole route is determined already at the creation of the frame in the sender. Therefore, only devices which are aware of the entire network topology can send ad-hoc messages to any destination. Such devices are termed controllers. Another device class, routing slaves, can send unsolicited messages to a number of predefined destinations. The required routes are downloaded by a controller to the routing slave (e.g., a motion sensor) during the association process. Mains powered routing slaves will also use these routes to forward messages on behalf of another node. Finally, nodes which only receive messages to act upon them (e.g., a dimmer) are called (non-routing) slaves.

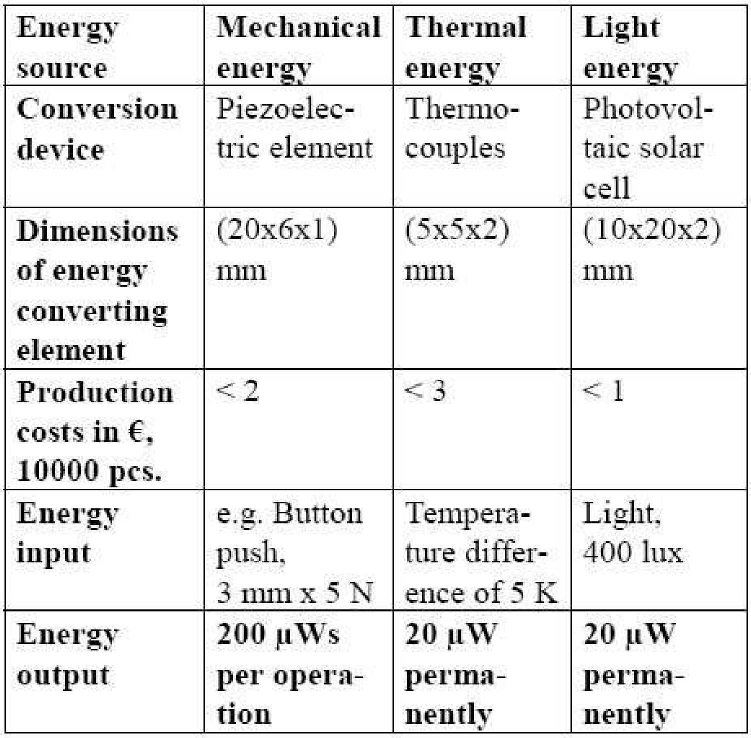
There is always a single controller (primary controller) that holds the authoritative information about the network topology. It is involved every time a device is to be included in or excluded from the network. Routes are automatically found, and defective routes are automatically removed to cope with devices changing their location and RF transmission paths becoming blocked over time. Medium access control involves carrier sensing for collision avoidance with random back-off delays. End-to-end acknowledged unicast and unconfirmed multicast and broadcast communication is supported. Security features previously support by the protocol are no longer part of the specification [23]. Z-Wave Alliance justifies this decision with a 30% smaller stack size and thus lower physical size and lower production costs for the modules.

In order to allow basic interoperability in multi-vendor systems, device class specifications define sets of mandatory, recommended, and optional commands. Self-association based on matching command definitions is advertised. There is currently only a single source for Z-Wave silicon: Zensys' mixed-signal ICs containing the transceiver, an 8051 microcontroller core, a Triac controller with zero crossing detection and an optional 3DES encryption engine. The microcontroller hosts both the Z-Wave protocol and the application software.

The protocol and device class specifications of Z-Wave are not freely available, neither are the IC manuals. Hence, advertised self-healing and self-organization properties of the protocol cannot be fully confirmed.

3.2 EnOcean

The key idea behind EnOcean [1] is to harvest enough energy from the environment to power a wireless sensor node long enough to collect all sensor data and transmit a telegram. This results in a significant reduction in maintenance effort, as there are no more batteries in wireless sensors that need to be replaced. Instead, electricity is provided by piezoelectric elements, thermocouples (not yet implemented) or solar cells, with their specifics shown in Figure 3.1. This concept could be realized thanks to recent technological advances such as efficient energy conversion, low power electronic circuits and reliable yet energy efficient radio transmission.



Energy source	**Mechanical energy**	**Thermal energy**	**Light energy**
Conversion device	Piezoelectric element	Thermocouples	Photovoltaic solar cell
Dimensions of energy converting element	(20x6x1) mm	(5x5x2) mm	(10x20x2) mm
Production costs in €, 10000 pcs.	< 2	< 3	< 1
Energy input	e.g. Button push, 3 mm x 5 N	Temperature difference of 5 K	Light, 400 lux
Energy output	**200 μWs per operation**	**20 μW permanently**	**20 μW permanently**

Figure 3.1: Amounts of energy, supplied by low cost and low size energy converters [1]

All EnOcean building blocks were brought together with a proprietary communication protocol highly optimized for energy saving. Messages are only a couple of bytes long (with a maximum payload of 6 bytes) and are transmitted at the high data rate of 120 kbit/s compared to other wireless protocols. Additionally, strategies such as not transmitting leading zeros are implemented. Thus, transmission takes less than 1 ms.

EnOcean uses ASK (amplitude shift keying) modulation and a novel RF oscillator that can be switched on and off in less than 1 microsecond. Thus, the oscillator can be switched off at every "zero" bit transmission, further reducing energy consumption. The short frame transmission duration in combination with the chosen modulation type results in a low statistical probability for collisions. In addition, frame transmissions are repeated three times. The delay between repetitions is varied at random to reduce the influence of periodic interference signals. The EnOcean protocol cannot increase transmission reliability by means of end-to-end acknowledgments since battery-less transmitter modules do not contain a RF receiver. The low collision probability is also presented as a key argument that the protocol will scale towards networks with a large number of nodes. There are currently four radio telegram types (corresponding to the available transmitter modules) identifying various combinations of boolean and 8-bit integer values, ensuring a basic level of interoperability.
Documentation for EnOcean modules is freely available, but only allows guesses at the radio protocol. Although occasionally advertised, no security mechanisms appear to be included.

3.3 NanoNET

NanoNET [24, 2, 25] operates at 2.45 GHz and supports data rates of up to 2 Mbit/s. The modulation scheme used is called Chirp Spread Spectrum (CSS). Symbols are transmitted as linear chirps, i.e., sinusoidal waveforms whose frequency increases (upchirp) or decreases (downchirp) over time. These chirps have a bandwidth of 80 MHz and a fixed duration of 1 μs. Their broadband nature makes them resistant against disturbances. Moreover, interference can only occur during the short transmission interval. CSS is part of a broader concept called Multi Dimensional Multiple Access (MDMA), a combination of phase, amplitude and frequency modulation. A CSS based physical layer related to nanoNET technology is under consideration as an alternative physical layer for IEEE802.15.4a. The NanoNET company offers the nanoNET transceiver, which is based on an implementation of their MDMA technology.

The portable protocol stack (PPS), as shown in Figure 3.2, complements the nanoNET transceiver. It specifies the Application Interface Layer (AIL), the Data Link Layer (DLL) and the Device Interface Layer (DIL). The AIL works as an interface between the application and the PPS while the DIL provides abstraction for the communication with the nanoNET chip. In between these two layers, the data link layer (DLL) provides methods for (un-) acknowledged, connectionless or connection-oriented communication, frame routing and security services. The MAC part of the DLL supports both unicast and broadcast communication. For medium access, Aloha [26], CSMA-CA and TDMA (Time Division Multiple Access) [27] can be used. It is a key feature of nanoNET that the stack is designed to be highly portable to different microcontrollers by separating hardware dependent and independent code. Regarding security services, the stack offers 128 bit encryption using an undisclosed stream cipher with support of one time pads, and message authentication.

3.4 KNX RF

The standard KNX protocol has initially been specified for the use over twisted-pair and later power line media. In addition to the established trans-

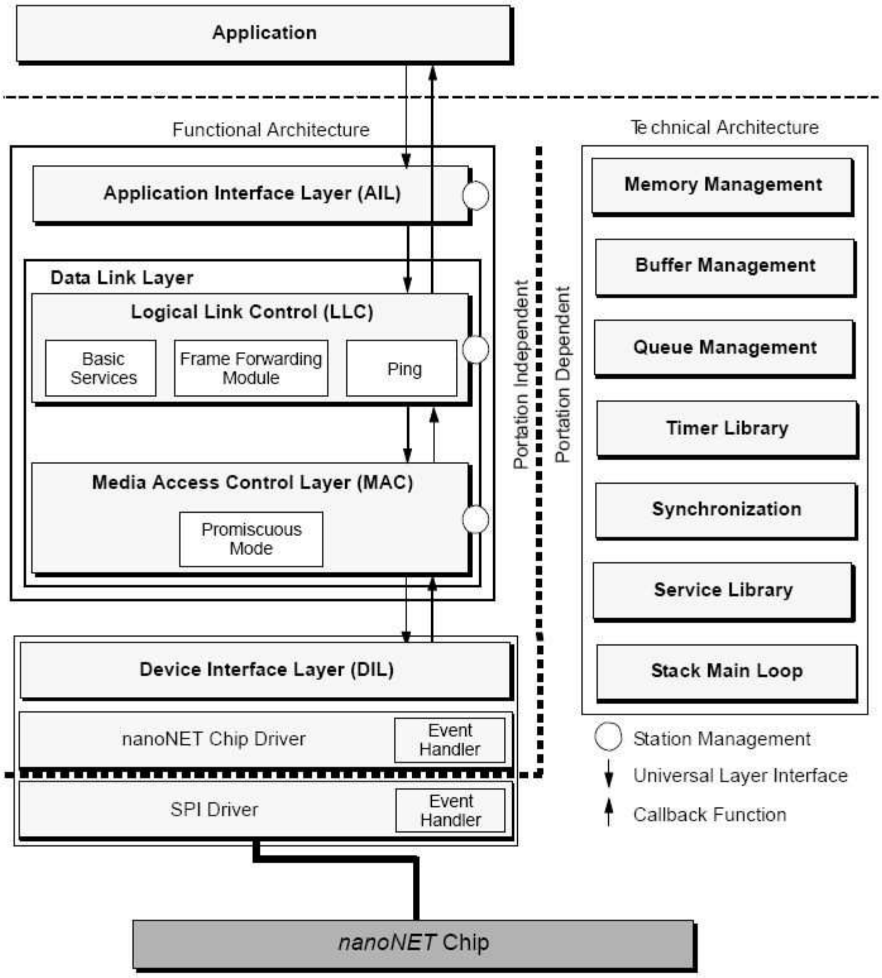


Figure 3.2: NanoNet Portable Protocol Stack [2]

mission media, a wireless alternative called KNX RF has been specified in Supplement 22 of the KNX Specification 1.1 [28]. Hence, KNX is not a protocol tailored for RF communication, but rather a home and building automation standard based on wired media that has been extended to support wireless communication. In the following, only the key specifics of the wireless part and the changes to the existing specification (which were necessary to accommodate the additional functionality in the specification) are being discussed.

KNX RF operates at 868.3 MHz using FSK modulation at a data rate of 16.4 kbit/s. The data link layer uses the FT-3 protocol defined in IEC 870-5-2. The bottom two layers of KNX RF were defined jointly with the wireless meter readout standard EN 13757-4:2005.

As a trade-off between functionality and the goals of low power consumption and low cost, KNX RF allows unidirectional (transmit-only) devices in addition to conventional bidirectional ones. Eliminating the receiver extends the battery lifetime of sensors as well as making them cheaper, also because only a subset of the protocol stack has to be implemented. On the other hand, it has the drawback that these devices cannot be configured via the network. This also excludes the possibility of downloading applications. Application download is however also significantly impaired for bidirectional devices due to the 1% duty cycle limitation which is in effect for the used ISM frequency band. Current KNX RF devices focus on the Easy configuration modes, where this restriction is less relevant.

KNX RF does not use link layer acknowledgments for a couple of reasons. First of all, transmit-only devices would not be able to receive acknowledgments. Also, acknowledgments would have to include a unique identification of their sender to be meaningful. This applies to multicasts in particular, but also in general since on an open medium data frames and acknowledgments of multiple individual transmissions may be mixed up. Instead of adding this overhead, KNX RF suggests implementing end-to-end acknowledgments at the application level where required. To detect and recover from transmission errors, KNX RF frames contain a CRC with hamming distance 6. The repeat flag available in standard KNX is replaced by a 3 bit link layer frame number (LFN). This allows greater flexibility for additional frame repetitions at the data link level. To extend the transmission range, retransmitters can be used. Retransmitters resend all frames they receive. To avoid resending a particular frame multiple times, a history list is used. In this list, the serial number (SN; a 6 octet long unique identifier for each device) and the LFN of each received frame are stored. If the SN and LFN of a received frame are already in the history list, the frame is not relayed but discarded.

Due to the nature of wireless communication and the support of transmit-only devices, KNX RF uses its own addressing scheme which is different from (although similar to) the standard KNX addressing scheme. Since RF is an open medium, the address spaces of neighboring installations would interfere with each other. Therefore it has to be guaranteed that each KNX RF installation has its own address space. For the Powerline medium, this was ensured by adding a 16 bit domain address that identifies the installation. This was not possible for KNX RF, since transmit-only devices cannot automatically receive the domain address via the network and entering it manually would be unfeasible. [1] Instead, extended addresses are used. An extended address is defined as the combination of the traditional KNX address and the serial number (SN) of the device. Since the SN is 6 octets long, an extended address uses 8 octets. Due to the uniqueness of the SN, an extended address of a group (extended group address) or of a particular device (extended individual address) does never interfere with an address from a neighboring installation. Since the SN is already unique, the traditional 16 bit part of extended individual addresses always defaults to 05FF. A drawback of this addressing scheme is that m – n relations are no longer possible. Since the extended group address contains the SN of the sender, two different senders can never send a message to the same extended group addresses. Therefore, only 1 – n relations are possible.

An advantage of the exclusive use of extended addresses can also be found in the fact that it provides an additional barrier for security attacks due to the vastly increased address space. An attacker has to figure out the 48 bit

[1] Moreover, it would be unclear which device should maintain this identifier in a distributed configuration approach.

SN of a device before injecting forged frames, which is impossible by brute force. Nevertheless, an experienced adversary can simply listen in to the packets transmitted via KNX RF and extract the serial number contained in clear in every message.
Because of the different addressing schemes used in KNX and KNX RF, media couplers are not only needed for physical interconnection. They also provide the necessary mapping between the different address spaces which has to be set up during system configuration.
KNX RF does not provide any security mechanisms. Since the transmitted data are neither encrypted nor an integrity check is performed, KNX RF cannot fulfil the high demands of security critical applications. Therefore, alternative technologies have to be used for these kinds of applications.

3.5 IEEE 802.15.4 / ZigBee

The focus of IEEE 802.15.4 [5] and ZigBee [7] is to provide general purpose, easy-to-use and self-organizing wireless communication for low cost and low power embedded devices. These technologies were designed for the use in actuator and sensor networks, including the HBA domain. The used protocol is compact yet flexible and powerful enough to meet relevant demands of HBA applications. A variety of manufacturers provide 802.15.4/ZigBee silicon [29, 30], including systems-on-chip.

As shown in Figure 3.3 IEEE 802.15.4 defines the physical layer and the MAC part of the data link layer according to the ISO/OSI model. ZigBee is specified on top of the IEEE 802.15.4 protocol and makes use of the layers specified therein. The ZigBee specification adds the network (NWK) and application (APL) layers to support more advanced communication functionality.

Strictly speaking, 802.15.4 is therefore an entirely independent protocol. Actually, applications and protocols can be (and are) realized based solely

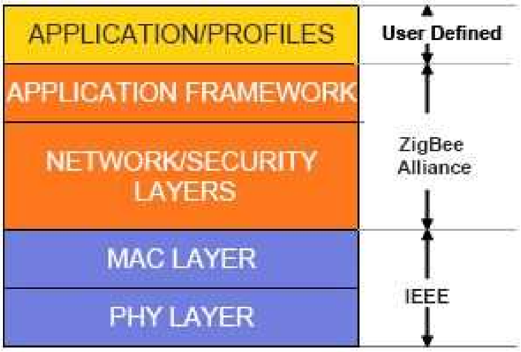


Figure 3.3: IEEE 802.15.4 and ZigBee protocol stack [3]

on IEEE 802.15.4, having nothing to do with ZigBee. Practically, however, the two standards are closely related to each other. They are not only complementary, but have mutually influenced the development of each other.

3.5.1 IEEE 802.15.4

Figure 3.4 shows the IEEE 802.15.4 physical and MAC layer according to the ISO-OSI model.

The physical layer specifies three different frequency bands: 868-868.6 MHz (with 1 channel and a data rate of 20 kb/s), 902-928 MHz (10 channels, 40 kb/s) and 2.40-2.48 GHz (16 channels, 250 kb/s). Different PSK (phase shift keying) modulation types are used for the sub-GHz bands and the 2.4 GHz band, all using DSSS (direct sequence spread spectrum). The availability of three different bands allows system designers to choose the most suitable frequency for the application.

The MAC sublayer on top of the PHY layer is, amongst others, responsible for (dis-)association of the device to/from a PAN, channel access using a CSMA-CA (Carrier Sense Multiple Access - Collision Avoidance) scheme and acknowledged frame delivery. The Logical Link Control (LLC) sublayer can access MAC services through the standardized Service-specific convergence Sublayer (SSCS).

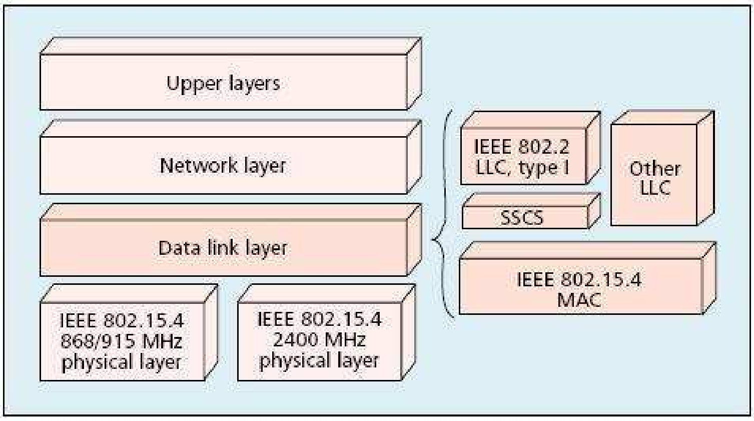


Figure 3.4: IEEE 802.15.4 in the ISO-OSI layered network model [4]

IEEE 802.15.4 classifies devices as Full Function (FFD) and Reduced Function devices (RFD) according to the complexity of the protocol stack that has to be implemented. This differentiation allows building cheaper, less power consuming and at the same time fully integrable devices, that can be used for simpler applications such as lighting control. RFDs are only capable of communicating with FFDs.

As the name implies, full function devices have to implement the whole 802.15.4 protocol stack. These devices can communicate with RFDs as well as other FFDs. A special FFD is the so called PAN coordinator. As shown in Figure 3.5, each network segment which is referred to as Personal Area Network (PAN) has a unique id, the PAN ID, and exactly one PAN coordinator. This coordinator is responsible for network management (e.g., address assignment) as well as for providing information about the network (e.g., definition and broadcasting the PAN ID). Devices are referred to by using a so-called 64 bit extended (or long) address which is unique for every device. Optionally, a PAN may employ 16 bit short addresses, which have to be configured to each device first.

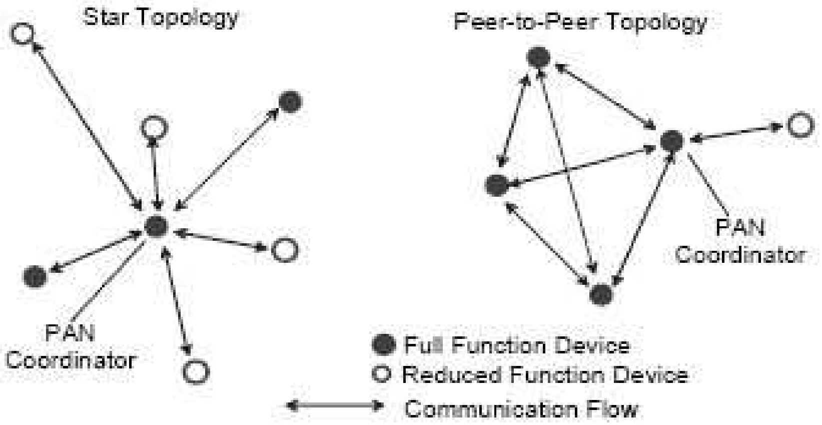


Figure 3.5: Star- and Peer-to-peer topology in IEEE 802.15.4 [5]

IEEE 802.15.4 specifies two different topologies, namely star and peer-to-peer, shown in Figure 3.5). While FFDs can communicate in peer-to-peer fashion, RFDs can only communicate with coordinators, resulting in a star topology. The choice of a topology can be based on the application design. An additional topology is called "clustered stars", where PAN coordinators act as bridges between PANs.

IEEE 802.15.4 differentiates between two different kinds of PANs: beacon enabled and non-beacon enabled networks.

In a beacon enabled network, a superframe structure (see Figures 3.6 and 3.7) is used. A superframe is bounded by so called network beacons which are sent by the PAN coordinator periodically. A beacon includes detailed information about the PAN (e.g., the PAN ID). Between these beacons, the Contention Access Period is located. Divided into slots, the CAP can be used by the PAN members to communicate anytime using a CSMA-CA scheme.

Additionally, as shown in figure 3.7, the PAN coordinator can dedicate a portion of the CAP known as Contention Free Period (CFP) to single de-

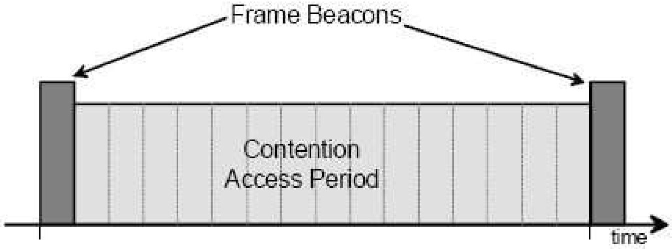


Figure 3.6: Superframe structure without GTS [5]

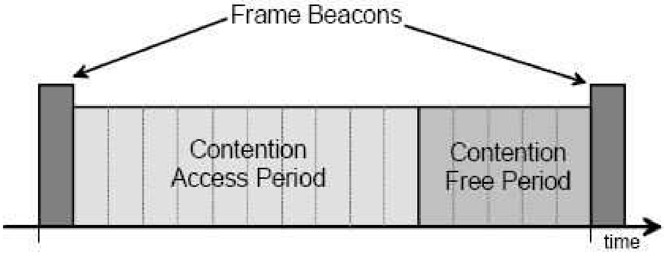


Figure 3.7: Superframe structure with GTS [5]

vices. The CFP consists of a maximum of 7 guaranteed time slots (GTS) and appears always at the end of a superframe. One or more of these guaranteed time slots can be assigned to one or more applications, hence ensuring exclusive access to the medium. This is especially advantageous for low latency applications, since no CSMA-CA is needed. Moreover, the use of a superframe structure helps to save energy. Between the periodic beacons, devices can enter sleep modes and only have to wake up for the next beacon.

In contrast to a beacon-enabled network, the coordinator does not send a beacon in a non-beacon enabled network. Therefore, all PAN members can communicate at any time using CSMA-CA.

IEEE 802.15.4 uses indirect data transmission, meaning that a transfer from a coordinator to a associated device is always instigated by the device. Figures 3.8 and 3.9 schematically show how data transfer works in beacon

and non-beacon enabled networks.
If beacons are used, the coordinator indicates in its beacon a list of devices for which data is pending. In order to retrieve this data, the devices poll the coordinator using CSMA-CA during the CAP or their GTS if assigned.

If no beacons are used, pending data cannot be indicated by the coordinator. The coordinator stores data for the devices, which have to poll the coordinator for data at an application defined rate. Communication to the coordinator works using a CSMA-CA scheme.

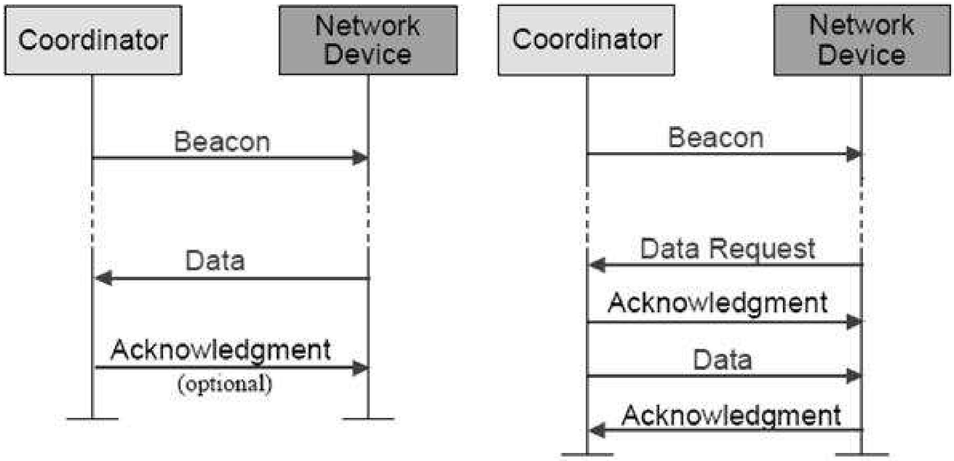


Figure 3.8: Communication to (left) and from (right) a coordinator in a beacon-enabled network [5]

In contrast to other wireless technologies, IEEE 802.15.4 already specifies different security services at the data link layer which rely on the AES [31] algorithm. These are access control, message confidentiality, message integrity and replay protection.

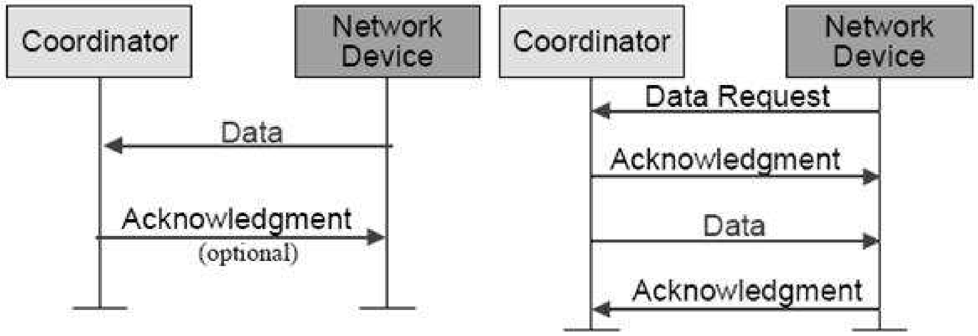


Figure 3.9: Communication to (left) and from (right) a coordinator in a nonbeacon-enabled network [5]

3.5.2 ZigBee

Figure 3.10 shows that the ZigBee specification is divided into three parts: network layer, application layer, and security services. The network layer (NWK) is responsible for enabling a self-forming and self-healing network by providing appropriate routing services including route discovery and maintenance. Based on the topologies specified in IEEE 802.15.4, ZigBee enhances peer-to-peer networking with support for a mesh topology providing increased reliability and scalability. With the cluster tree topology (see Figure 3.11), ZigBee offers a combination of star and mesh topology to support both high reliability and support for battery-powered nodes. The variety of available topologies brings along that system designers are free to choose the most appropriate topology for the given requirements.

ZigBee also includes mechanisms for joining and leaving a network. In addition, the NWK of a ZigBee coordinator can start a network and assign addresses to new participants following a distributed scheme.

The ZigBee application layer (APL) consists of the application support sub-layer (APS), the ZigBee device object (ZDO), and the application framework (AF) hosting the application objects (AO). The manufacturer-defined application objects incorporate the actual functionality of the device. Each

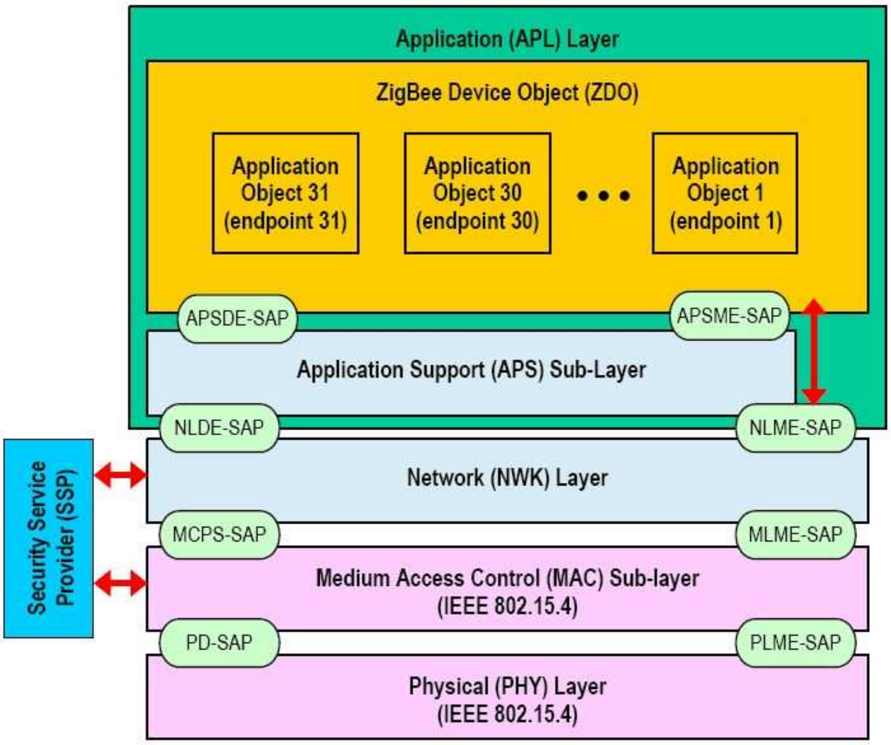

Application (APL) Layer
ZigBee Device Object (ZDO)
Application Object 31 (endpoint 31)
Application Object 30 (endpoint 30)
Application Object 1 (endpoint 1)
APSDE-SAP
APSME-SAP
Application Support (APS) Sub-Layer
NLDE-SAP
NLME-SAP
Security Service Provider (SSP)
Network (NWK) Layer
MCPS-SAP
MLME-SAP
Medium Access Control (MAC) Sub-layer (IEEE 802.15.4)
PD-SAP
PLME-SAP
Physical (PHY) Layer (IEEE 802.15.4)

Figure 3.10: ZigBee in the ISO-OSI layered network model [6]

AO forms an independent functional sub-unit and can be addressed via its endpoint number. AOs communicate via free form messages or by manipulating each other's state variables. For the latter purpose, the AF provides the key value pair (KVP) service with acknowledged and unacknowledged get, set and event notification interactions. Standard data types are also defined. KVP allows tagged data structures using compressed XML (which a gateway can expand to textual representation for use by other systems). The semantics of a free form message or a whole set of key-value pairs are encapsulated in its numeric cluster identifier. Cluster IDs thus allow accessing specific functionality within an AO.

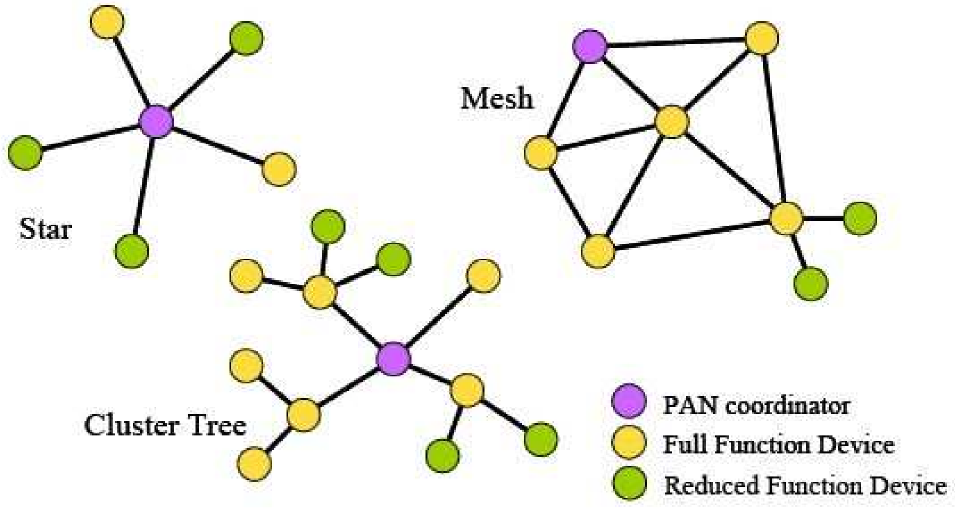


Figure 3.11: Possible ZigBee network topologies [3]

The APS provides an interface between the NWK and the device and application objects. It is responsible for delivering messages to their destination endpoint and cluster. The APS of a coordinator maintains a binding table (which maps a source address/endpoint/cluster combination to one or more destination addresses and endpoints, keeping the cluster ID) and forwards messages accordingly. This mapping is also used as basis for multicast group relationsships.

The ZDO is a special application (residing at endpoint 0) that encapsulates management operations concerning APS, NWK, and other parts of the stack. These include discovering and joining a network, establishing bindings, and configuring security services (e.g., key establishment and authentication). The ZDO also handles device and service discovery. The services of the ZDO are available to the AOs via public interfaces. Security mechanisms are integrated into all layers. A security service provider (SSP) handles tasks such as encryption which are common to all of them.

While the ZigBee specification is openly available to the public, the ZigBee device profiles are not available free of charge.

3.5.3 IEEE 802.15.4 / ZigBee security

IEEE 802.15.4 specifies its security mechanisms in the data link layer.

Access control and message integrity are provided by means of adding a message authentication code (termed MIC, message integrity code) to outgoing frames. The MIC is a secure checksum of the message and is computed with the help of a secret key shared by the devices involved in the particular message exchange. Only if the MIC is correct an incoming frame will be accepted.

Replay protection relies on adding a (typically monotonically increasing) sequence number to each frame. Incoming frames are only accepted if the sequence number is greater than the last one received.

Finally confidentiality between sender and receiver is established by data encryption with the AES algorithm. Again, the symmetric key has to be shared between the communication partners.

802.15.4 radio ICs maintain an access control list (ACL) that allows to specify the combination of security mechanisms (called “suite”) and key to be used separately for every communication partner. In practice, however, a single key is typically shared by all devices in the network. The use of shared (symmetric) keys is clearly a drawback of IEEE 802.15.4 security mechanisms. It poses problems when thinking of topics such as key distribution over unsecured networks and supporting the temporary association of mobile devices. Moreover, acknowledgement frames are always sent unencrypted and unauthenticated so that system designers cannot rely on them as a security measure.

ZigBee security leverages the mechanisms provided by IEEE 802.15.4 and complements them with essential administrative aspects such as key generation, distribution and administration. ZigBee introduces different keys for network or end-to-end security as well as the concept of a Trust Center, a node which is trusted by others to handle security related operations. In a ZigBee network, the Trust Center authenticates devices wanting to join, provides them with keys and offers functions for establishing network-wide and peer-to-peer secure connections. Normally, the role of the trust center is assumed by the ZigBee coordinator, but mobile devices can take it over as well.

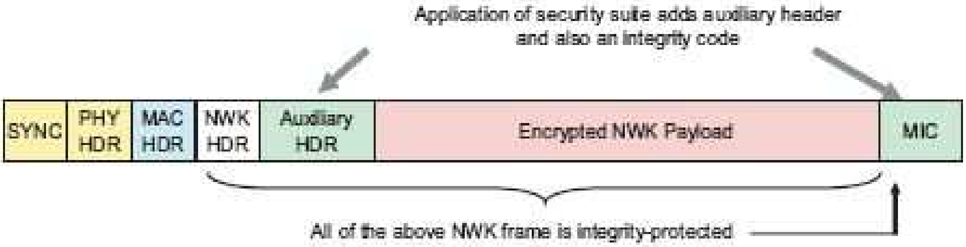

Application of security suite adds auxiliary header
and also an integrity code
SYNC
PHY HDR
MAC HDR
NWK HDR
Auxiliary HDR
Encrypted NWK Payload
MIC
All of the above NWK frame is integrity-protected

Figure 3.12: ZigBee frame with security on the NWK layer [7]

3.6 Protocol summary

Tables 3.1 and 3.2 provide an overview of common features of the protocols discussed in chapter 3.

Regarding the listed frequency bands, all protocols, except KNX RF, that are capable of using the 868MHz band in the EU are also specified for operation in the 908MHz ISM band used in the US. For better legibility, the 908MHz (US) band was therefore omitted from the table.

A protocol is considered “published” if the protocol specification is available to the general public for a “non-discriminating” fee – comparable to publication by official standard bodies. Incidentally, transceivers for the unpublished protocols in this overview are available from a single source only, while multiple alternatives exist for the published protocols.

All protocols employ at least some kind of mesh networking scheme. In KNX RF, packet forwarding is optional and typically performed by dedicated retransmitters.

Table 3.1: Summary of protocol features I

	$Frequencyband$	$Datarate$	$Security$
$Z - Wave$	868 MHz (EU)	9.6 kbit/s	advertised
$EnOcean$	868 MHz (EU)	120 kbit/s	no
$nanoNET$	2.4 GHz	2 mbit/s	yes
$KNX\ RF$	868 MHz	16.4 kbit/s	no
$IEEE802.15.4/ZigBee$	868 MHz (EU), 2.4 GHz	20, 250 kbit/s	AES

Table 3.2: Summary of protocol features II

	$Published$	$max.Nodecount$	$Modulation$
$Z - Wave$	no	232 per network	FSK
$EnOcean$	no	2^{32}	ASK
$nanoNET$	no	2^{48}	CSS
$KNX\ RF$	yes	256 per line	FSK
$IEEE802.15.4/ZigBee$	yes	2^{16}	PSK

Chapter 4

Examples of Wireless Communication

This chapter presents possible approaches how wireless technology can be combined with traditional wired technologies. The examples are set in a Home and Building Automation context where systems with long life cycles are employed. There, the installation and maintenance of solely wireless systems (i.e., systems without connection to any other wired control network) is only a fraction of the whole automation effort. Rather, existing wired automation networks will be combined and enhanced with wireless nodes – often with the goal in mind to extend the system's lifetime. However, with increased spreading of the rather new wireless technology in HBA, also component prices will drop thus resulting in higher market penetration of purely wireless systems.

The following sections describe how wireless networks may be used as an enhancement [32, 33]. Possible problems are identified and discussed. Additionally, two case studies of implementations are presented.

4.1 Wireless tunneling bridge

An obvious application for a wireless technology is to substitute cables where appropriate. Formerly not connected (or connected using wires) network segments are bridged using the wireless medium. This solution follows a tunneling approach, as illustrated in Figure 4.1.

4.1.1 General description

In a tunneling approach, the sender receives frames from one control network (CN) and wraps them into tunneling packets. These packets are transmitted over the tunneling medium (TM, host network) to the receiver where they are unwrapped and forwarded to the other CN segment. A major benefit of this solution is that it is completely transparent to the control network as control network frames remain unchanged.

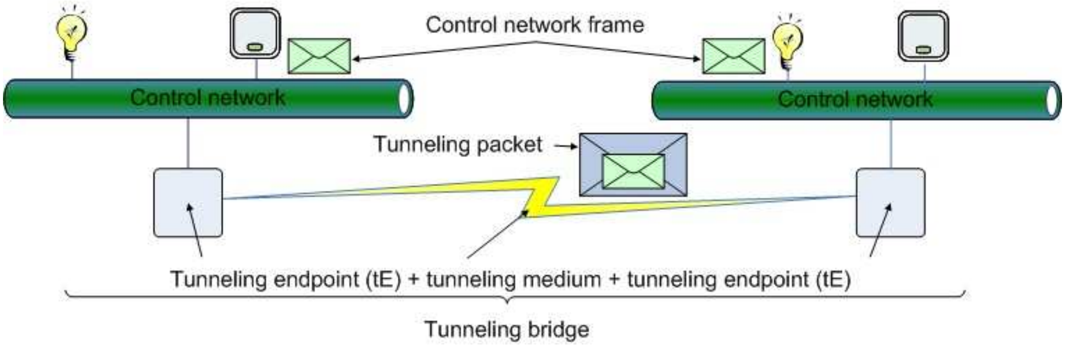


Figure 4.1: Connecting control network segments via tunneling

In the simplest case, every tunneling endpoint (tE) always has a fixed association with another single tE. In this setup, tunneling devices always come in pairs.[1] Such a tunneling bridge can be likened to cutting the network cable and splicing it back together via the bridge. Operating on the

[1] Although there may be multiple pairs, a member of one pair will never communicate with a member of another.

MAC layer of the control network, it merely wraps the received frames into tunneling packets and passes them to the other side. It requires almost no configuration effort. Setting the network address of the associated tunneling end-point is sufficient.

This approach is very suitable for providing remote access to a HBA installation (e.g., via the Internet). However, it is of limited use in practice when a short range wireless network is targeted as tunneling medium. One possible application would be connecting two parts of a low-traffic segment over a public street. Another would be easy connection of a mobile device (i.e., laptop or PDA with engineering software). For convenient use in practice, however, the latter application would already require some sort of discovery protocol. This discovery protocol would automatically provide the user with a list of currently available connection points (i.e., tEs) at a specific location. Without this assistance, connecting the mobile device to network segments – all with separate tEs – would be cumbersome.

4.1.2 Tunneling device classes

As various applications in a wireless network exist, also single nodes may be tailored to their function. The software running on the nodes may be built less complex which contributes to cheaper nodes. According to the functionality they provide, different classes of wireless devices can be identified.

Common for all device classes is a tunneling endpoint implementation layer (tEil). Situated at layer 3, it is the task of the tEil to mediate between the control network and the tunneling medium service access points (SAPs).

- Tunneling application device

  Tunneling application devices (tADs) are devices that implement only the application layer of the control network and an interface to the tunneling medium. tADs can be used when it is not reasonable or useful to implement all layers of the control network, i.e., if only one device is connected to a tunneling endpoint. For example, a wireless light switch or a mobile device that is connected to a tunneling endpoint just has to generate protocol conform message frames but needs not take care of any routing issues. Hence, in case of a single device connected, implementing all layers down to the physical layer of the control network would result in unnecessary overhead and make nodes more complicated and expensive.

- Tunnel access points

  Tunnel access points (tAPs) are devices that implement both the control network and the tunneling medium down to the physical layer. This approach is necessary, if not only a single device is connected to the wireless medium via a tunneling endpoint, but a whole network segment. In contrast to a CN consisting of a tAD only, a tunnel access point has to provide mechanism for medium access and routing as it has to deal with several devices communicating simultaneously. Additionally, a tAP provides the same services of the tunneling medium as a tAD.

  An example of a tunnel access point is shown in Figure 4.1.

- Tunneling interfaces

  Tunneling interfaces (tIFs) are a special kind of tAP, that do not implement any higher layer of the control network. For example, PC-based nodes are typically connected to the CN via an adapter (the tunneling interface) which does not implement any higher layers of the control network. For communication with the PC, the adapter has to implement a second tunneling connection (which is typically point-to-point; e.g., USB).

Compared to the classification presented in [34], this classification focuses on network protocol aspects rather than functional points of view. It is illustrated in Figure 4.2.

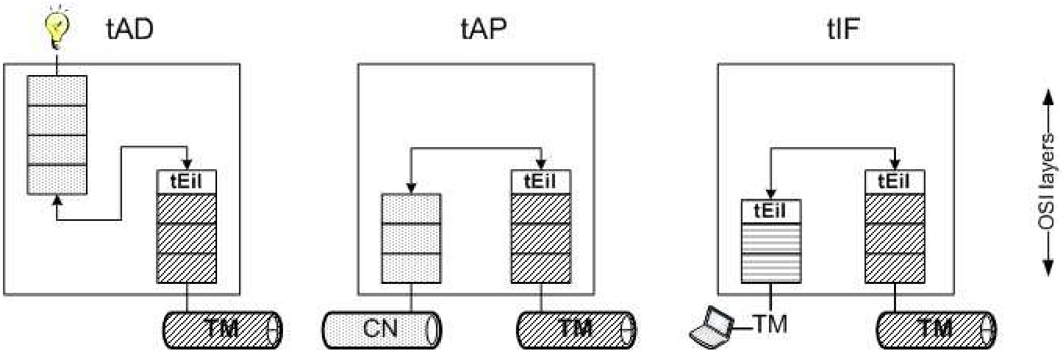


Figure 4.2: Tunneling device classes

Using the simple point-to-point scheme a tunneling bridge provides, the only possibility to integrate a tAD (or tIF) is to pair it with a tAP. Otherwise, the network would be limited to only two devices – even if there are other tAPs in the network. This means that two wireless tADs (sensor and actuator) would need two tAPs to communicate even if they are within transmission range of each other. This is obviously inefficient since each wireless end device would need its own tAP.

4.1.3 Case study

An example of an enhancement provided by wireless technology is the KNX – IEEE 802.15.4 tunneling bridge shown in Figure 4.3. Two identical bridges are used to connect two wired control network segments implementing the KNX/EIB automation standard. For the wireless interconnection, an IEEE 802.15.4 network was chosen because its specification is openly available and provides published security features.

At first glance, it may not be obvious why a KNX network should not use the native wireless extension KNX RF. But when taking a closer look, it shows that KNX RF leaves ample room for improvement. By replacing KNX RF with IEEE 802.15.4, security support as well as a potential cost reduction is obtained easily since IEEE 802.15.4 transceivers are already being produced in large numbers. Moreover, it is an open, well proven technology. An IEEE 802.15.4 link also easily accommodates the data rate on the KNX/EIB twisted pair medium, which operates at 9.6 kbit/s. As a particular improvement over KNX RF, zero configuration is possible.

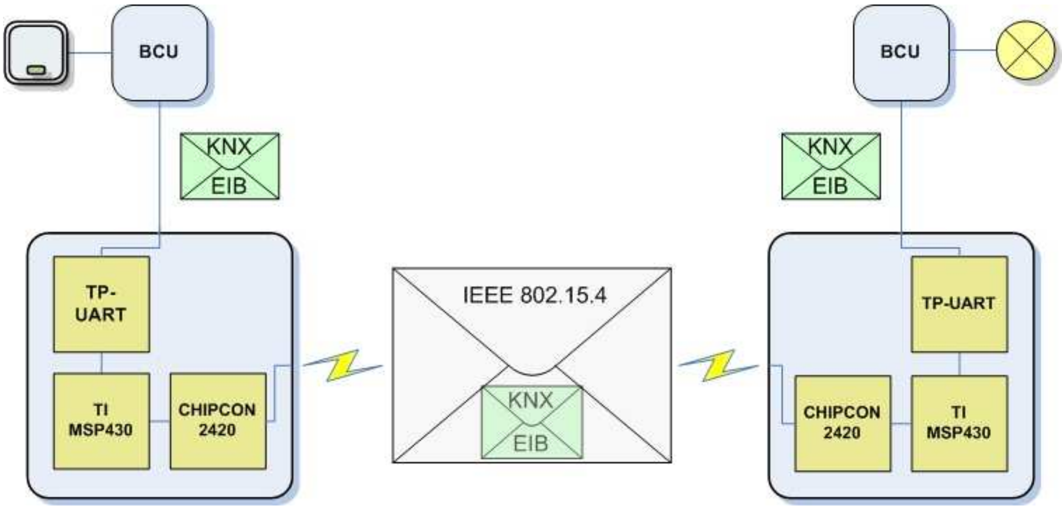


Figure 4.3: KNX – IEEE Tunneling bridge

The KNX – IEEE 802.15.4 tunneling bridge is comprised of three major parts. The TP-UART works as an interface between the KNX/EIB installation and a Texas Instruments MSP430 series microcontroller. The Chipcon 2420 RF transceiver is used for sending IEEE 802.15.4 frames in the 2.4 GHz band using a peer-to-peer, non-beacon network configuration. The TI MSP430x149 is equipped with 2 USART interfaces (one is required for communication with the TP-UART, one for the Chipcon 2420) and supports different low power modes. The MSP430 application configures the Chipcon 2420 for IEEE 802.15.4 communication with the required parameters and enables its RF transceiver.

KNX/EIB frames are received via the TP-UART and handed over to the TI MSP430. The MSP430 application configures the Chipcon 2420 for IEEE 802.15.4 communication with the required parameters and enables its RF transceiver. IEEE 802.15.4 frames containing the unmodified KNX/EIB frame as payload are sent via the RF connection, received by the second (identical) tunneling bridge and are acknowledged by an ACK frame. The microcontroller at the receiving side extracts the KNX/EIB message and forwards it to the TP-UART that places it onto the second KNX/EIB segment. Simultaneous communication in both directions is possible. Although the current implementation does not make use of any security mechanism, it establishes an excellent basis for extensions in that direction. First, only the tunneling connection could be secured by means of 802.15.4 security mechanisms. Such a solution would remain entirely transparent to the KNX/EIB devices. However, it does not provide protection against attacks on the KNX/EIB wired network. Such protection could for example be achieved by deploying EIBsec [17], which would be perfectly possible on this hardware platform.

The setup presented above only allows communication between two (paired) devices. The main advantage of this wireless solution is its complete transparency to the wired automation network. Both the wireless and the wired protocol could be substituted without affecting each others configuration. Only the respective interface of the tunneling bridge has to be adapted.
In practice, however, home and building automation applications often require simultaneous communication with multiple devices. Unlike in Figure 4.3, a light switch will probably be used to control more than one light at

different locations. Such an application cannot be realized using this setup (except all recipients are attached to a single wired backbone). As the tunneling bridge relies on peer-to-peer communication, multiple peer-to-peer connections would be needed to reflect the multicast communication used on the wired segment.

The most obvious solution to the problems imposed by the peer-to-peer communication scheme is to exchange unicast messages with broadcast messages. The resulting device, a wireless tunneling router, is discussed in the next section.

4.2 Wireless tunneling router

In the general case, it is desirable to model networks as shown in Figure 4.4. In order to be able to fully replace a wired control network with a wireless one, it shall be possible to integrate wireless sensors, actuators and controllers and wireless management devices (e.g., light switch, PDA) using tunneling application devices. Furthermore with the use of tunneling interfaces the integration of PC-based configuration and management devices is also possible. Tunnel access points can still provide interconnections to wired control network segments.

4.2.1 General description

As outlined before, peer-to-peer communication cannot be used to completely substitute a wired network. To overcome the drawbacks of the simple point-to-point scheme, the tE would need to be able to communicate with other tEs on the tunneling medium. More precisely, it would have to communicate with the proper tE to reach the destination node in the control network. This cannot be achieved with a plain tunneling bridge, which does not have knowledge about the addressing scheme used on the CN.

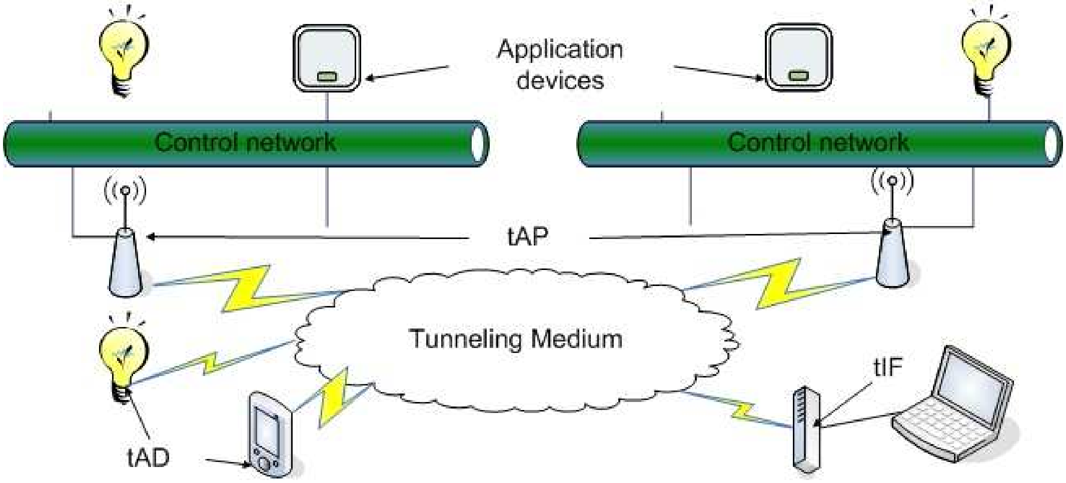


Figure 4.4: Wireless enabled CN using tunneling

A tunneling router must understand the routing scheme of the control network, and provide an appropriate routing scheme (e.g., mesh networking) on the tunneling medium. The required routing tables would have to be configured manually given the resource limitations of low-power wireless nodes.

In home and building automation applications, process data are exchanged mostly in communication groups. The corresponding messages are referred to as group messages. Multiple senders are able to send process data to multiple receivers according to a producer-consumer scheme based on group addresses where senders and receivers are not aware of each other. This communication scheme is called multicast communication and has to be supported by the protocol. Section 5 provides more information on multicast in general and different protocols.

Since many wireless protocols do not specify any support for multicast communication, another solution has to be found. One opportunity would be to simulate group communication by sending a tunneling packet (including the group message) to each member of the group using point-to-point communication.

Obviously, this approach is not applicable. Since multicast groups can be large, this approach will lead to high network traffic as well as to a significant delay of group communication as a single message at the sender has to be sent to every group member sequentially. Furthermore, the configuration and maintenance effort will increase rapidly since elaborate routing and group membership tables are required in each node. Hence, employing such a communication scheme has the significant drawback that nodes wanting to communicate with a group have to be aware of all group members and furthermore must constantly be informed of changes in the group structure.

To overcome these limitations, a broadcast scheme, which is provided by most wireless protocols, can be applied. Using broadcast messages, a node wanting to communicate with a single device or a group simply sends the according packet (containing the recipient or group address) encapsulated in a wireless message to all nodes in the network. Upon reception, each node can determine itself, based on the destination address of the encapsulated packet, whether it is a recipient or not.

Clearly, the advantage of a broadcast scheme is its easy implementation and the fact that is does not require a priori knowledge of group members. Unfortunately, using a broadcast message to communicate with only a subset of all nodes implies (unnecessary) overhead. Additionally, it is in the nature of broadcast algorithms specified for wireless networks, that a single broadcast frame may be received multiple times by a device. This circumstance is known as duplicate message reception and demands special precautions. For example, devices have to be aware of messages already received to be able to detect duplicate frames. This increased storage demand further decreases communication efficiency.

4.2.2 Case study

This section presents the concept of a wireless KNX/EIB-IEEE 802.15.4 tunneling router using a broadcast scheme for communication. The broadcast itself is accomplished using a simple flooding algorithm.

Every KNX/EIB group message is encapsulated unchanged into an IEEE 802.15.4 broadcast telegram. The destination address is set to the IEEE 802.15.4 broadcast address and the PANID to a predefined value. Each wireless device which is within the transmission range of the sender receives this broadcast message and resends it. The IEEE 802.15.4 frame header and trailer are discarded and the KNX/EIB message is extracted. A tAD or tIF checks the group address. If it is configured to be a member of that group, it processes the message. Otherwise it discards it. In case that the receiving device is a tAP, it simply inserts the message to the wired KNX/EIB segment where it is transmitted using the usual KNX/EIB multicast mechanism. If a tAP receives a message on the wired segment (i.e., the message has originated in the wired control network segment), it broadcasts it.

So far, the retransmission scheme of all devices simply relaying the message only ensures that every KNX/EIB frame reaches every device as long as the network graph is connected.[2] However, the scheme necessarily causes message duplications and cycles.

In order to solve this problem, it must be ensured that every node repeats a received message exactly once. For this purpose, every message needs to be tagged with a message ID (mID). This mID consists of a local sequence number (sNR) and the IEEE 802.15.4 long address of the sender node (sAD). Because the long address is unique, also the combination with the sequence number is.

The sNR is initialized with zero at power up and increases monotonically with every broadcast sent. Since the mID is globally unique, it is guaranteed

[2] An edge in the network graph corresponds to a wired or wireless link between two devices (nodes).

that each broadcast message can be uniquely identified in the whole network as long as its transmission is fully completed before the local sequence number is reused.

The mID of each incoming broadcast message is stored in a local broadcast table (BCT). If another message with the same sAD is received, its sNR is compared against the one found in the BCT. If the incoming sNR is lower or equal, the message is discarded, as this means that the device has already received and relayed this message. Otherwise the message is re-broadcasted and its mID replaces the old one in the BCT. This is the case if one sender has sent multiple, independent messages.

An entry in the BCT is only removed when it can be guaranteed that no messages with this mID are present in the entire network. This is the case when the message corresponding to this entry has been resent by every node.

This timeout is $time\_out = t_{hold_{max}} \cdot hop\_count_{max}$ with $t_{hold_{max}}$ being the maximum time that the node will hold and try to resend an incoming message until it is discarded, and hop_count_{max} being the longest path a packet can take through the network. If the sAD of an incoming message cannot be found in the BCT and the BCT can hold no more entries, the message is discarded.

This algorithm guarantees *source FIFO ordering* [35] and *at-most-once semantics* [36] for delivery. Note that the BCT size and sNR range are not critical to these properties. Both parameters only influence the probability for successful message delivery.[3]

Assuring these two properties is of particular importance since reordering and duplication cannot occur on the KNX/EIB wired medium. Thus, the higher stack layers cannot deal with these cases. Necessarily, it comes at the price of possibly losing some message or the other which could otherwise

[3] Even if the sNR of a particular sender wraps before $time\_out \cdot sNR\_range$ has elapsed, the consequence is only that the new message is ignored by those nodes which still hold the re-used mID in their BCT.

have been relayed. However, this is no restriction since the KNX/EIB network layer does not support reliable transmission anyway. Applications that require end-to-end reliable transmission have to handle this on their own.

While the BCT size and sNR range are not critical to these properties, the timeout is. Given that the maximum message hold time cannot be modified, we can only make better use of a given table capacity by limiting hop_count_{max}. In general, its lowest upper bound is given by the number of (wireless) nodes minus one. However, it can be lowered further by introducing a broadcast time to live (BCTTL) parameter in every message. Its value determines the maximum number of times a broadcast packet is retransmitted. This field is initialized by the originator (the initial value is common for all devices). Every time before a tunneling device resends a broadcast frame, the value of its BCTTL is decremented by one. If the result is zero, the frame is discarded. Otherwise, the result is set as new BCTTL and the frame is relayed.

The algorithm works as long as each wired network segment has no more than one tAP assigned, as it only allows devices to determine if they have received duplicate frames. If multiple tAPs per segment are to be allowed, it is not sufficient to detect duplicates at a tAP only. The reason is that also a wireless link (direct or via intermediate nodes) between these tAPs can exist in addition to the wired connection. Hence, some arbitration scheme between these tAPs has to be found that prevents duplicate frame creation on both the control network and the tunneling medium.

First, we regard the case of a tunneling packet originating outside this wired segment. Since a wireless path exists between the tAPs, both tAPs will forward the encapsulated KNX/EIB frame to the wired network segment. Thus, the group message will be duplicated on the KNX/EIB network. Even worse, since the CN frame cannot be identified as having been inserted by a tAP (because it is a standard KNX/EIB message), the other tAP will rebroadcast it, creating a loop. To avoid this, the tAPs must also create

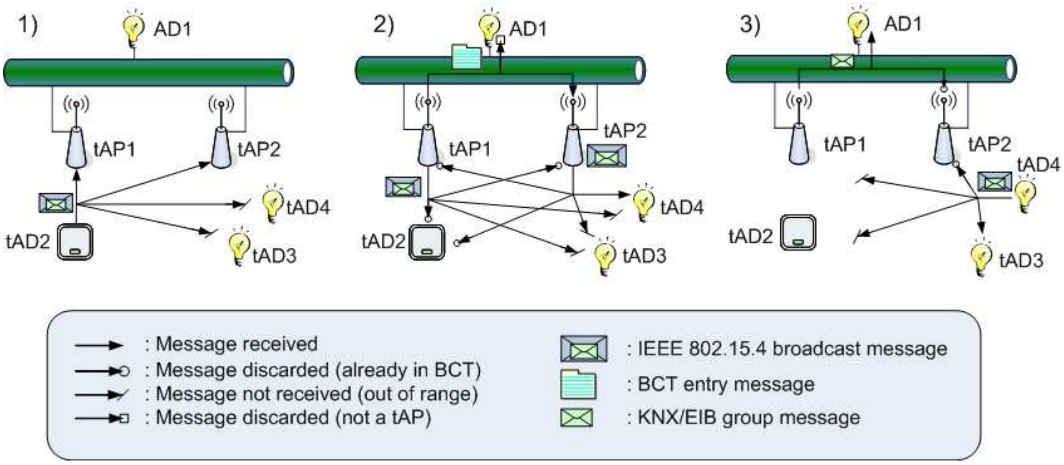

1)
AD1
tAP1
tAP2
tAD4
tAD2
tAD3
2)
AD1
tAP1
tAP2
tAD4
tAD2
tAD3
3)
AD1
tAP1
tAP2
tAD4
tAD2
tAD3
: Message received
: Message discarded (already in BCT)
: Message not received (out of range)
: Message discarded (not a tAP)
: IEEE 802.15.4 broadcast message
: BCT entry message
: KNX/EIB group message

Figure 4.5: Duplication prevention

a BCT entry for such frames, which then provides enough information to prevent or filter duplicates.

For this purpose, the tAP must not simply discard the IEEE 802.15.4 header – and with it, the necessary information to create a BCT entry –, but rather transmit this information over the wired segment together with the CN frame. This is done by sending a BCT entry message before the control network frame. This message is a KNX/EIB extended frame containing the mID and BCTTL.[4] All BCT entry messages are sent to a group address predefined for this purpose. Figure 4.5 illustrates this concept: tAD2 and tAD3 can communicate via tAP1 and tAP2 and the wired segment – without the installer having to take any special precautions even if tAP1 and tAP2 are within wireless communication range of each other.

In Figure 4.5, a frame originating at tAD2 is received by both tAP1 and tAP2. tAP1 first receives the message and puts an according BCT entry

[4] Since another node could transmit a higher-priority CN frame between the BCT entry message and the CN frame, the BCT entry message also contains a hash value computed over its associated CN frame. This hash value allows the receiving tAP to correctly associate the BCT entry message.

message on the control network segment. Moreover it creates a BCT entry and re-broadcasts the message. In the meantime, also tAP2 has received the message over the wireless link. In the given example, it is assumed that tAP2 has not yet received the BCT entry message from tAP1. Hence, tAP2 creates a BCT entry and also relays the message. According to the proposed algorithm, devices with a corresponding BCT entry simply discard the message. In the rightmost image, tAP1 now inserts the received message in the wired KNX segment. However, tAP2 ignores it, because of the already existing BCT entry. Finally, also tAD4 re-broadcasts the message. Upon reception at tAD3 all devices have received the frame and the algorithm terminates.

The case of a "native" CN frame originating in the wired segment is almost symmetric. Again, the existence of a wireless path between the tAPs will lead to message duplication and loops. To retain at-most-once semantics, the tunneling packets must be sent out with synchronized mIDs by all tAPs on the segment. This is achieved by every tAP transmitting an initialization message on the wired segment at power-up. This message (sent to the predefined, reserved group address) contains the sAD of the tAP. Every tAP receiving this message then uses this sAD instead of its own for the tunneling packets it generates in response to incoming CN frames (and only for these tunneling packets), starting with a sNR of zero. For added robustness, the current sNR for such packets can also be included in the BCT entry messages to allow resynchronization.

4.3 Summary

The main challenge today is not deployment of "pure" wireless systems, but rather enhancing established wired automation systems with wireless technology. The use of wireless technology ranges from substitution of a (single) wire, i.e., interconnecting two network segments via tunneling, to more complex hybrid systems. While simple tunneling bridges are of limited use in practice, hybrid wired/wireless systems will become more and more popular in the future. However, especially the latter systems still require future development. For example, the storage demand for the BCT and the increased bandwidth consumption of the duplication prevention algorithm employed in the tunneling router presented above somehow contrasts the requirements of low node cost and high battery lifetime for wireless systems. This reveals that there obviously is still room for improvements.

One main flaw of the tunneling concepts discussed is their dependency on broadcast communication. Although very robust, also the transmission overhead is large. A possible advancement could come along with the use of multicast algorithms instead of broadcast. The following chapters of this thesis present different multicast algorithms suited for wireless systems and outlines a multicast-enhanced ZigBee protocol.

Chapter 5

Multicast communication

Multicast communication has become a widely used feature in the Internet today. Applications such as IP TV and e-learning rely on this more efficient mode of transmission. Of course, group communication can also be employed in the home and building automation field to improve communication.

5.1 Definitions

- Unicast

  Unicast defines the transmission of information to a single destination in a network. Hence, a unicast message is addressed to exactly one recipient at a time. For the delivery of a unicast message, it will be required that nodes participate in forwarding the message to the recipient.

  Figure 5.1 shows two examples of unicast communication. On the right hand side, unicast routing is necessary for message delivery.

- Broadcast

  Broadcast defines the transmission of information to all destinations in a network. Hence, broadcast is the complete opposite of unicast.

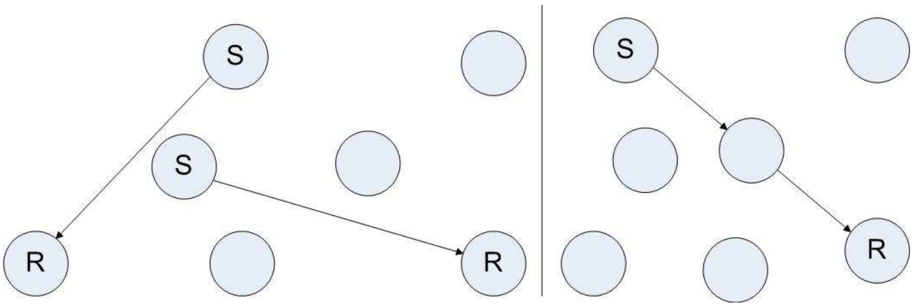


Figure 5.1: Examples of unicast communication

To reach all network members, it may be necessary, that intermediary nodes participate in forwarding and relaying the broadcast message. Broadcasting can also be limited to so-called broadcast zones, a subset of nodes contained in the network. In wireless ad-hoc networks, broadcast messages are sent to reserved network addresses such as $0xffff$ in IEEE 802.15.4. Other technologies such as Token Ring use flags in the message header to indicate a broadcast transmission.

Especially in wireless ad-hoc networks, a (broadcast) algorithm is needed to ensure delivery to all nodes and at the same time prevent duplicate frames. Figure 5.2 shows a typical broadcast scheme in wireless ad-hoc networks, where recipient nodes also relay the broadcast message.

- Multicast

  Multicast defines the simultaneous transmission of information contained in packets to a defined subset of destinations in a network in the most efficient way. The subset of nodes is called a multicast group.

  Using multicast, the transmission has to be accomplished in the most efficient way, meaning that the message count necessary to reach all members of the multicast group has to be kept as low as possible. As shown in Figure 5.3, this requirement can only be fulfilled if just one copy of the multicast message is sent along the path until the links to

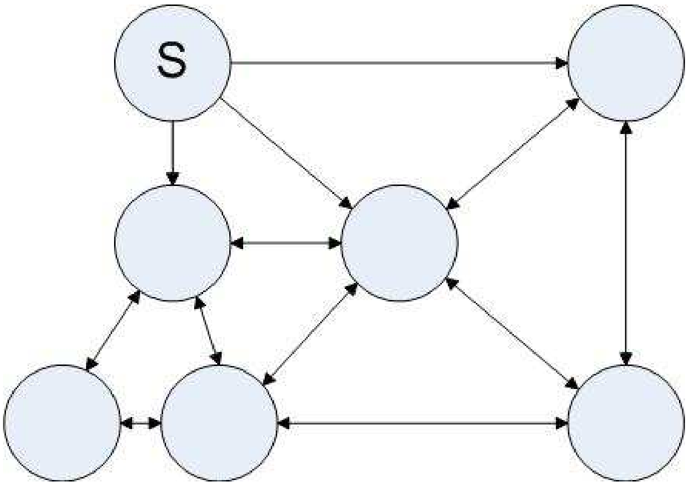


Figure 5.2: Example of broadcast communication

different multicast receivers split. Only then, the multicast message is copied and sent to all recipients independently. The exact process of delivery is specified in the multicast (routing) algorithm.

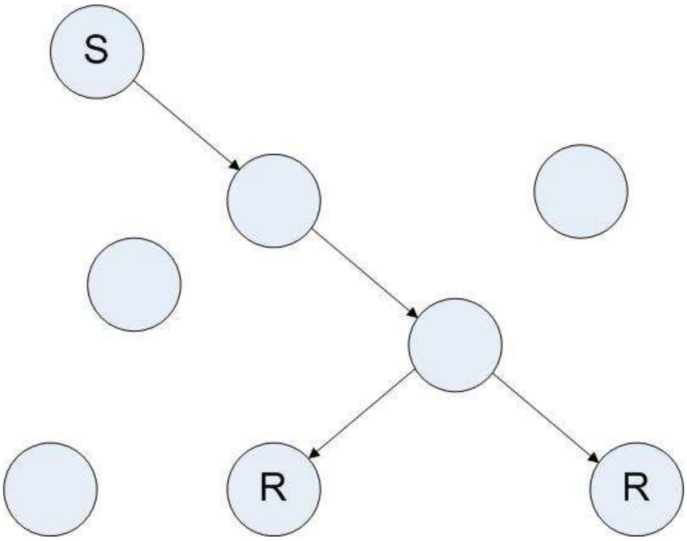


Figure 5.3: Example of multicast communication

Using multicast, it is necessary to define *multicast groups*. These groups are formed by a set of nodes that are registered to the group. Each group has assigned a single multicast address which is used by senders to communicate with all group nodes at once. Hence, a single node has to join a multicast group first before it receives these particular messages [13].
Multicasting is advantageous when the same information has to be sent to more than one destination. Since only one message is sent and routed (in contrast to multiple unicast messages) the communication costs can be significantly reduced.

- Routing

  Routing is defined as the selection of a path in a network in order to transmit packets along this path. Routing is the basis for message forwarding. The routing algorithm tries to find an optimal delivery path from a source node to a recipient. This path is an ordered list of nodes part of the network, that are in between the sender and recipient. Each of these intermediary nodes is called a router. Upon reception, these routers forward the message to the next node on the routing path until the message reaches its destination. Each router stores information on routing paths to destinations in the network in a so called routing table. Hence, routing algorithms specify the construction and maintenance of routing tables in the network nodes.

  Routing algorithms have to take into account, that nodes may fail and paths containing these nodes may become unavailable. To counter this problem, two different strategies are used in traditional routing protocols (i.e., routing protocols designed for wired networks).

  a) Distance vector algorithms

  Distance vector algorithms are based on a pairwise assignment of link costs between all nodes in the network. The cost metrics that can be applied are based on physical distance between the nodes, intermediary hop-count, time delay and link quality. According to the path costs, each node computes the best path to each other in the network. If nodes fail, each router has to verify and probably adapt its paths. Distance vector algorithms can be implemented easily and are very efficient for small networks. Unfortunately, the effort for computing the path increases polynomially with the node count. Moreover, upon route changes, all paths are re-computed thus making distance vector algorithms less suited for larger networks. Also the count-to-infinity problem [37] can occur using distance vector algorithms. An example of a distance vector algorithm is the Routing Information Protocol [38], which is widely used for Internet routing.

b) Link-state algorithms

The basis of link state algorithms is a map of the network represented as a graph. This map contains all nodes and their interconnections with each other. All nodes in the network receive this map and calculate the next-hop from them to each possible destination in the network. The best next-hops are stored and form the basis for message forwarding.

Link-state algorithms need not communicate with neighbors apart from the stage of creation of the network map. Additionally, only the next-hop has to be calculated, which makes the link state protocol's performance independent from the node count in the network. As a drawback, the map of the network has to be recomputed upon node failure and the implementation is more complex and requires more storage capacity than the distance vector approach. Nevertheless, network map recomputing time is bounded and generally faster than in distance vector implementations.

Routing protocols are necessary for packet delivery in uni-, multi- and broadcast communication if not all recipient nodes are directly connected to the sender.

5.2 Requirements

Multicast is used to deliver identical messages to multiple recipients at the same time. To alleviate the overhead of multiple unicast transmissions between the sender and each recipient, multicast protocols are used. Here, a (single) copy of the message is sent along a delivery path and replicated only when necessary (i.e. when branches part).

Many requirements for an optimal multicast routing protocol exist in literature. This section lists identified design goals of an ideal multicast routing protocol.

- Efficiency

  Efficiency is the key motivation for the use of multicasting. A multicast message should be delivered as single message as long as possible. Especially in ad-hoc networks bandwidth is scarce thus making high efficiency even more valuable.

- Reliability

  A multicast protocol should guarantee reliable transmission of packets. This means that messages should not only arrive at one destination, but at all intended recipients.

  Another aspect of reliability is the at-most-once semantic. Due to the nature of the wireless communication medium, it is probable that single nodes receive identical messages more than just once. These duplicate messages must be detected and dealt with by the protocol to ensure predictable system behavior. Other problems that have to be addressed include the hidden-node problem [39] and the detection and prevention of routing loops. A metric often used for measuring reliability is the packet delivery ratio.

- Robustness

  An ideal multicast protocol (especially one designed for MANETs) has to be robust. In Chapter 2, interference was presented as one of the biggest challenges to be solved. Regarding multicasting, interference can lead to line breaks thus rendering delivery paths useless. Protocols should be prepared to cope with such problems ideally fast and efficiently. Apart from interference, mobility of the nodes and node failure lead to similar problems. Node failure also adds another requirement. Protocols must not have single points of failure, for example relying on the same node in all delivery paths.

- Minimal control protocol overhead

  Multicast protocols rely on control messages to manage group membership, establish delivery paths and discover neighbors. The amount (including message count and message size) of these messages is called overhead. Additionally, also the payload that is sent to nodes without being strictly necessary, has to be counted as overhead. An ideal protocol should get by with few control messages (always compared to the number of data packets sent) but yet establish and maintain reliable delivery paths.

  For example, if broadcast is used, duplication prevention (e.g. estimation of a suitable TTL value) can be implemented to reduce unnecessary overhead.

- Minimized routing costs

Routing cost is a characteristic comparable to both quality of service and control overhead. However, routing costs rather are an estimation based on information about network links.

Several metrics can be assigned to network links then referred to as link costs, e.g. link quality or distance between nodes. The total sum of the costs of all links used in the delivery path are the routing costs. As multiple, redundant delivery paths for a multicast transmission may exist, a routing protocol can be tailored to prefer links with less costs.

It cannot be said that optimization regarding one parameter is also favorable regarding other metrics. For example, minimizing the physical distance a packet has to travel may come at the price of using connections with worse link quality due to punctiform interference.

- Resource management

  Resource management is of particular interest in wireless ad-hoc networks. A multicast routing protocol influences resource usage of the nodes significantly. The implementation of an ideal protocol should get by with a small code size. In the best case, it should be even sufficient to run the protocol code on a single device. Moreover, routing tables should be kept as small as possible, for example by limiting the discovery of redundant paths. Thinking of battery-powered nodes, the code execution time should be kept low to allow nodes to enter power-saving or sleeping modes.

- Independence from unicast algorithm

  A multicast routing protocol should be operable without being dependent on any unicast routing algorithm. This requirement derives from the goal of compatibility. A multicast protocol should be situated on top of any underlying network protocol and in the best case be completely independent of it. This can go as far as a multicast protocol specifying its own unicast algorithm just for the purpose of routing multicast control data. This approach guarantees that the underlying protocol (i.e. either the whole protocol or parts of it like the unicast algorithm) can change without affecting the multicast routing protocol. On the other hand, a multicast routing protocol becomes applicable for all different ad-hoc network protocols. In times of rapid development, this requirement ensures technological advancement.

- Loop free

  A multicast protocol should guarantee that constructed routing paths never contain loops. A loop in the routing path would lead to multicast messages circulating through the network without ever reaching the intended destination. A possible reason for routing loops are inconsistent views of the current network state: using distributed routing tables, some nodes may have detected a link failure and updated their databases while others still rely on the old state. A multicast frame could now be forwarded incorrectly by the second node, and thus be routed in a loop as long as the old state information persists. Such a routing loop is referred to as *transient loop*.

- Further Quality of Service aspects

  Quality of Service (QoS) in networks is a kind of collective term that summarizes requirements affecting performance. In this thesis the term QoS is discussed for the sake of completeness and can be understood as a criterion to further differentiate protocols according to the level of performance they offer. One criteria, i.e., delay, describes the time it takes a packet to be routed from sender to receiver. Apparently, the multicast routing protocol has a great influence on the delay as it computes the exact path and the influences the hop count. Also the time that this computation takes matters. Routes can be predefined or may be established just upon packet creation. Finally, the maximum throughput that can be achieved when employing a specific protocol is of interest. As the amount and size of control messages vary, also the maximum throughput achievable changes.

It has to be said that all of the criteria listed above describe the ideal multicast routing protocol and can, in practice, not be fulfilled simultaneously. For example, high reliability will always be connected with higher control overhead, and so most protocols are balanced (or can be configured) either way. Here, it is the task of a system engineer to choose the most appropriate protocol for the given problem.

5.3 Overview of Multicast Protocols

This section presents different multicast protocols that have been proposed for wireless networks. The goal of identifying characteristics and establishing a classification already influenced the selection of the protocols. To cover as much of the multicast protocol spectrum as possible, especially protocols employing contrasting technologies (e.g., tree and mesh based, source and receiver initiated) are featured. This is done with regard to discovery of the most suitable multicast protocol that can be mapped to enhance the ZigBee specification. However, this selection approach may bring along that also protocols not perfectly suited for wireless sensor networks are discussed here.

5.3.1 Flooding

The most trivial multicast protocol is flooding. Using flooding, a sender always broadcasts the multicast packet. Since it is the definition of a broadcast that it reaches all nodes in the network, nodes simply have to check whether they are part of the corresponding multicast group or not. For this purpose, nodes have to keep track of all multicast groups they belong to.

Flooding although simplistic and inefficient, is very robust and easy to implement. Nevertheless, the resulting overhead is not acceptable in networks consisting of resource-limited nodes.

5.3.2 Multicast Routing Extension for Open Shortest Path First

Open Shortest Path First (OSPF) [40] has a long tradition in TCP/IP unicast routing. Additionally, the Multicast Routing Extension for OSPF (MOSPF) [41, 42] has been specified. MOSPF is capable of both unicast and multicast routing.

OSPF and hence also MOSPF keeps a distributed, replicated database of all interconnections in the network called link state database. In order to synchronize the database, reliable flooding is used. In order to forward multicast packets, MOSPF calculates a tree based on the link state database information. In the tree, the sender acts as root node. For each multicast group a tree is created upon reception of the multicast frame. The tree's branches end at the group members. Packets are replicated only when branches of a path diverge. The taken route is based on both the source and destination address and calculated using Dijkstra's algorithm [43] thus guaranteeing the shortest delivery path with least costs. Group management in MOSPF is accomplished using the Internet Group Management Protocol (IGMP) [44]. Using IGMP messages, member location is acquired and then distributed throughout the network using flooding. Delivery trees are recalculated if necessary.

Through the creation of a delivery tree, MOSPF can reduce the amount of network traffic significantly compared to flooding. MOSPF was designed for TCP/IP networks and hence is focused on quite static networks. In dynamically changing networks (like wireless ad-hoc networks) MOSPF can be used but does not achieve an optimal performance. As delivery trees are created for all sender - receiver combinations the protocol does not scale well to higher node counts.

5.3.3 AMRIS

The Ad hoc Multicast Routing protocol utilizing Increasing id-numberS (AMRIS) [9] has been developed at the National University of Singapore and is specifically tailored to meet the requirement of ad hoc networks. AMRIS is a multicast protocol that supports multiple senders and receivers (i.e., m–n relations). The protocol tries to construct and maintain a delivery tree for multicast messages and is independent of any underlying unicast protocol, since routing information need not be exchanged with other nodes.

The key idea behind AMRIS is the assignment of non-consecutive id-numbers to the nodes, that increase with the distance from the sender. Hence, these numbers can be used to reflect the logical height of nodes in the delivery tree. Because the numbers are non-consecutive, local route repair and (re-) joining the multicast group is facilitated. It is a requirement, that a single root node exists and that every node except the root node has one parent. The parent-child relationship is defined using session-specific multicast session member ids (msm-ids). (Child) Nodes can choose their parent from all nodes in reach that have a smaller msm-id.

AMRIS operates in two phases, tree initialization and tree maintenance.

- Tree initialization

  During this phase, multicast sessions are created and advertised throughout the network. Nodes can primarily choose whether they want to participate in this multicast session, but non-participating nodes may be forced to join if they are necessary as forwarding nodes for the delivery tree.

  At first, a root node, which functions as special initialization node (Sid), is elected among the group of senders. The Sid starts the initialization phase by broadcasting a NEW-SESSION message to the neighbors. The NEW-SESSION message contains the Sid msm-id, a multicast session id and the value of some routing metric. Nodes that receive the NEW-SESSION message compute their own, larger and non-consecutive msm-id and then themselves broadcast a NEW-SESSION message with their own msm-id. Moreover, each node keeps a Neighbor-status table containing multicast session id, the according msm-id and the routing metric.

If a node receives more than one NEW-SESSION message, the message with best metrics is used as basis for msm-id computation, but information from all messages is stored in the neighbor table. In order to prevent broadcast storms, the rebroadcast of NEW-SESSION messages is delayed by a jitter.

Until now, nodes only collect information on neighboring nodes and possible connections to them. As shown in Figure 5.4 a node that wants to join a multicast session sends a JOIN-REQ message to a potential parent (i.e., a neighboring node with a smaller msm-id). The set of potential parents can be derived from the information contained in the neighbor status table. The parent which receives the join request first checks if it is already part of the multicast session. If yes, then the join request is accepted by replying a JOIN-ACK packet to the child. If the parent itself has not joined yet, it requests a join using one of its potential parents. This process is repeated, until one node finally accepts the join request. The JOIN-ACK now recursively propagates back the path until all nodes have joined. The multicast delivery tree is now fully constructed, but has to be maintained constantly using the tree maintenance mechanism.

- Tree maintenance

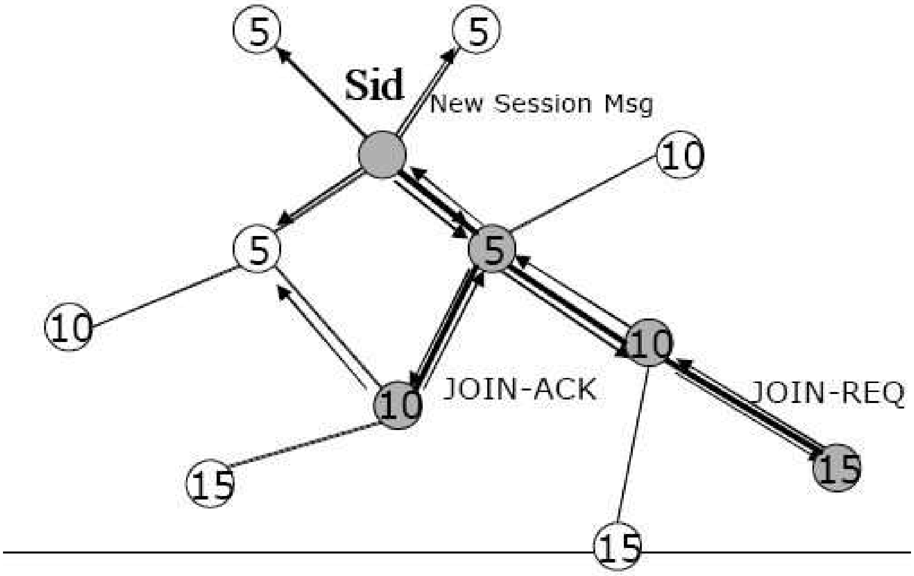


Figure 5.4: Initialization phase of the AMRIS protocol [8]

Tree maintenance is necessary to keep (mobile) nodes connected to the multicast delivery tree. In case of a broken link, the protocol defines that always the node with the larger msm-id (i.e., the child node) is responsible for re-joining. The beacon interval specifies the time interval between the periodic connection updates of the nodes.

AMRIS specifies a branch reconstruction (BR) mechanism which consists of two different methods BR1 and BR2. BR1 is used when at least one neighboring node is a potential parent for the disconnected node X. X can then request a join using JOIN-REQ. The same procedure as in the initialization phase starts. If the chosen parent node is not yet a member of the multicast tree and also cannot join the tree, it sends JOIN-NACK to X. X now tries to rejoin via all other potential parents in its neighbor status table. If this does not succeed either, the second branch reconstruction algorithm (BR2) is executed.

> Using BR2, a disconnected node X broadcasts JOIN-REQ messages to all its neighbors in reach. A special range field hereby states how often the broadcast can be relayed by the neighbors and thus how far it propagates through the network. All nodes that receive the broadcast check if they can fulfill the JOIN-REQ and answer with JOIN-ACK if so. Thus, node X may receive more than one JOIN-ACK and has to choose one parent. It sends a JOIN-CONF to this selected parent and is now part of the multicast delivery tree.
>
> Tree maintenance is also executed, if a new device wants to join a multicast delivery tree. This new device first listens to the traffic of neighboring devices and then computes an msm-id based on the msm-ids of its neighbors. In order to join, the node then uses the branch reconstruction algorithms.

The simulations in Figure 5.5 show, that the packet delivery ratio is highly dependent on the beacon interval. If the beacon interval is higher (e.g. 1000ms), broken links cannot be detected and countered quickly enough to guarantee optimal packet delivery. On the other hand, a short beacon interval (e.g. 500ms) does not automatically mean a better delivery ratio, but rather worse performance. Since some nodes in mobile ad-hoc networks move quickly, it may happen that nodes located at the border of the network lose connection for a short period of time only before the are connected again. If the beacon interval is short, these disconnections are detected before the node is in range again. This results in the execution of the branch reconstruction algorithm even if the node is already in reach again. Because of the additional management packets sent while executing the branch reconstruction, packet collisions occur that deteriorate the overall delivery ratio.

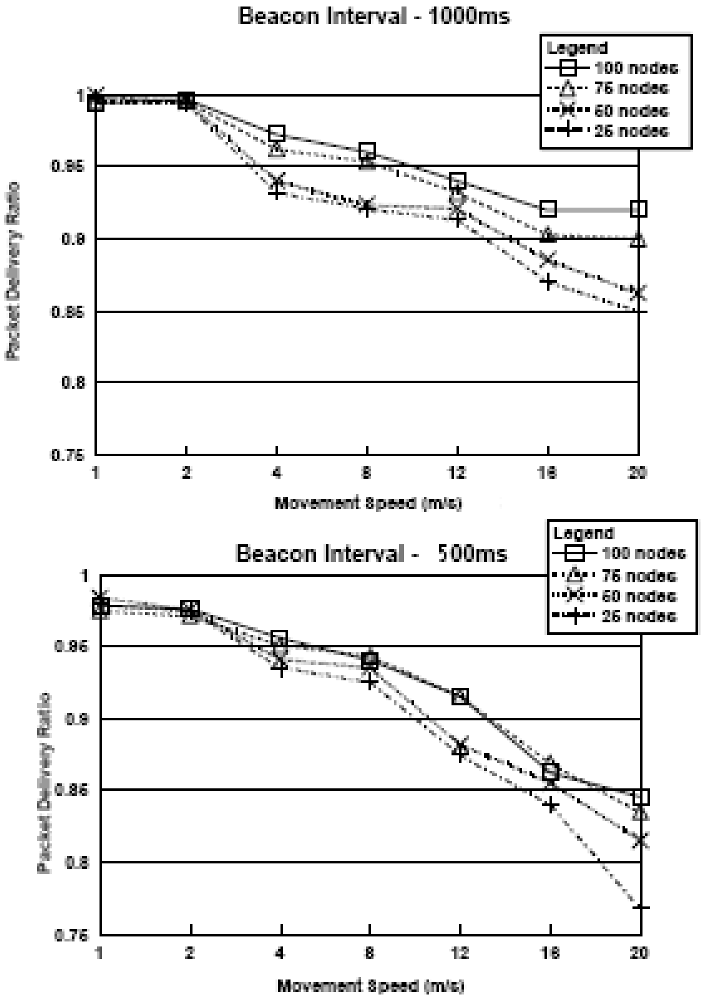


Figure 5.5: Simulation results of packet delivery ratio vs. node movement speed at different beacon intervals using the AMRIS protocol [9]

5.3.4 On-Demand Multicast Routing Protocol

The On-Demand Multicast Routing Protocol (ODMRP) [10] was developed at University of California, Los Angeles. It is a mesh based routing protocol supporting m–n relations and is also a draft of the MANET working group of IETF [45].

In ODMRP, the multicast mesh is created by a source node on demand. As shown in Figure 5.6 at startup the sender S floods the network with JOIN-QUERY packets. Nodes that receive this packet check if they are part of the intended multicast group. If the receiving node is not part of the multicast group, it stores the sender's node ID in its multicast routing table and rebroadcasts the packet. If the packet reaches a member of the multicast group, this node creates and broadcasts a JOIN-REPLY packet which contains multicast sender and all found next node IDs as payload.

The JOIN-QUERY packets are also used by the sender to periodically update the multicast paths.

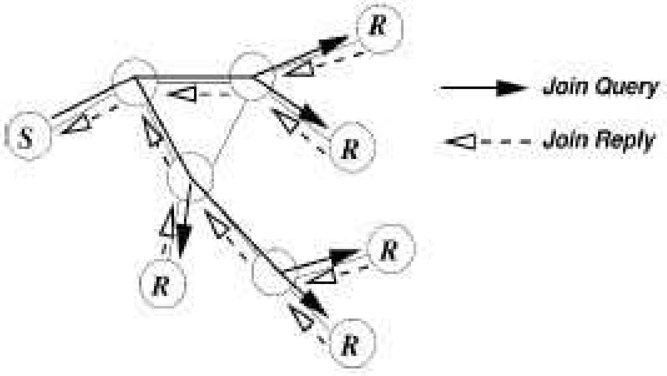


Figure 5.6: On-demand procedure for membership setup and maintenance [10]

This JOIN-REPLY packet is received by the neighboring nodes, which check if one of the next node IDs contained in the packet matches its own node ID. If this is the case, the node knows, that it is on the multicast path between sender and receiver. Hence, the node is part of the forwarding group for this multicast and accordingly sets the forwarding group flag (FG_FLAG) before broadcasting its own JOIN-REPLY packet. When the JOIN-REPLY reaches the source node, it has been propagated via all nodes on the shortest path between sender and receiver. DCMP calls this mechanism *backward learning*. It has been published as *reverse path forwarding* by Dalal et al. in [46].

The group of internediary nodes is called forwarding group. Figure 5.7 illustrates the forwarding group concept. As the name implies, forwarding group nodes participate in forwarding the multicast packets from the sender to the receiver, where data is always sent using the shortest path between any two members. It is important to note that also multicast receivers can become forwarding nodes if this is required.

The algorithm described above is not only used during setup but also for route maintenance.

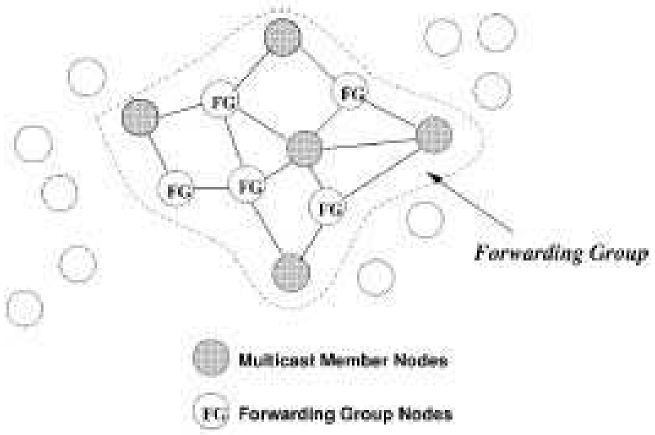


Figure 5.7: Forwarding group concept in ODMRP [10]

In contrast to tree based routing protocols, a mesh approach guarantees increased robustness. As shown in Figure 5.7, there are multiple paths between the nodes. This redundancy comes into play when single links fail, e.g. due to node movement. Figure 5.8 compares the packet delivery ratio of ODMRP with competing technologies. It is evident, that the mesh net approach of ODMRP contributes to the high delivery ratio.

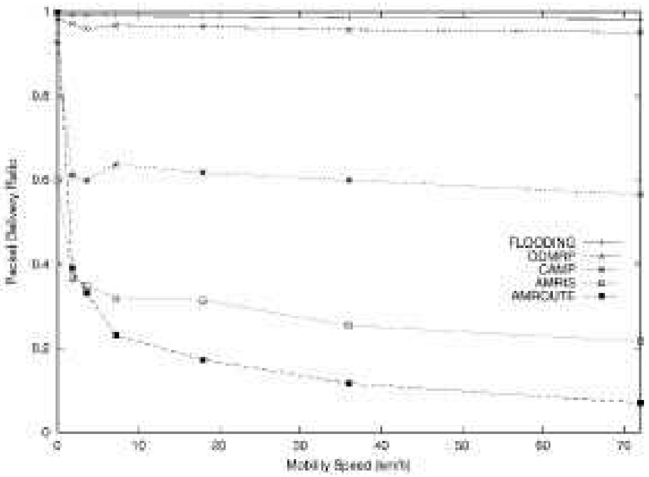


Figure 5.8: Packet delivery ratio as a function of node mobility [10]

One of the major advantages of the on-demand multicast routing protocol is its unicast capability. While some other protocols depend on an already existing, underlying unicast algorithm to function, ODMRP can operate as unicast protocol itself. [47] evaluated the performance of ODMRP working as unicast routing protocol. Their study showed, that unicast ODMRP suffers from high packet loss even in rather static networks. They attributed this to environment noise and interference. Hence, ODMRP should be used as unicast protocol if necessary, but it cannot completely replace a specially tailored unicast protocol.

Regarding multicast groups, the use of ODMRP comes along with easier management. Multicast receiver nodes can leave a (particular) multicast group by simple stop sending JOIN-REPLY packets for this group. This means, that no path to this node is established and hence multicast message are no longer forwarded. Even more trivial, a multicast source (ODRMP allows m–n relations) simply leaves a multicast group by no longer broadcasting JOIN-QUERY packets.

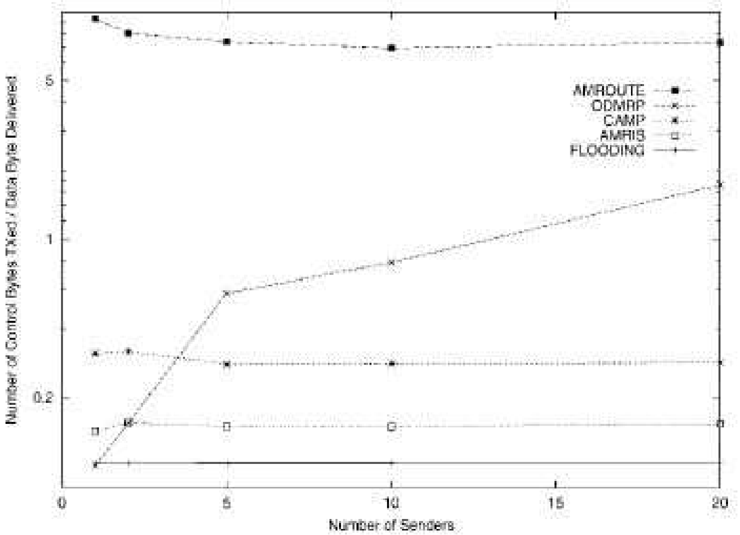


Figure 5.9: Control overhead in ODMRP as a function of number of senders [10]

5.3.5 Dynamic Core Based Multicast Routing Protocol

The Dynamic Core Based Multicast Routing Protocol for ad hoc wireless networks (DCMP) [11] proposes a shared-mesh based approach for multicast communication. It is based on ODMRP and expands ODMRP's mesh net with a tree routing concept.

While the On-Demand Multicast Routing Protocol (cf. Section 5.3.4) has a high packet delivery ratio even at higher mobility of the nodes, Figure 5.9 shows that the protocol does not scale with higher node counts as the control overhead increases drastically. DCMP counters this problem by classifying source nodes in *Active* and *Passive* nodes, with the goal to decrease the control overhead as passive nodes need not flood JoinReq packets.

DCMP builds a mesh formed of multiple core based trees [48]. Core based trees, rather than flooding the data everywhere, map the multicast group address to a particular unicast address of a so-called *core router*. This core then builds explicit distribution trees centered around itself.

In order to reduce control overhead, DCMP classifies senders into three categories:

- Active Sources

  Active sources keep their multicast relations up to date by regularly flooding control packets. Hence, active sources can be compared to *sources* in ODMRP.

- Passive Sources

  Passive nodes never actively participate in multicast delivery path creation. These nodes are associated with an core active node that forwards data packets for them.

- Core Active Sources

  Core active sources are active sources that additionally act as cores for passive sources. Core active nodes have assigned passive nodes to them and are responsible for creating the shared mesh. The number of associated passive nodes is limited by $MaxPassSize$. The maximum distance (in hops) between a core active source and a passive node is bounded by $MaxHop$.

Limitation of passive sources is necessary to prevent too many active sources to change into passive mode as this would lead to decreased robustness of the mesh. Less nodes would participate in multicast path setup and maintenance thus unbalancing the shared mesh approach towards a tree routing algorithm.

As in ODMRP, sources initiate setup of multicast paths by flooding JOIN-REQUEST packets. Each packet additionally contains the *CoreAcceptance* flag, indicating whether the source node can support more passive sources. In general, DCMP multicast route setup works as in ODMRP using reverse path creation with JOIN-REPLY packets. The difference lies in the fact that when an active source in a multicast group receives a JOIN-REQUEST packet, it requests a change of its status from active to passive if the following conditions are satisfied.

1. *CoreAcceptance* is set.

2. *MaxHop* has not been exceeded. This can be detected using the hop counter of the JOIN-REQUEST packet.

3. The receiving active source (also called *ToBePassive*) has a node ID that is less than the node ID from the sending source (also called *ToBeCore*).

The ToBePassive source requests passive status by sending a *PassReq* packet with the flag *CoreReq* to the ToBeCore source. After sending *PassReq* the node starts a timer and waits for the *Confirm* packet. The passive request is forwarded to the *ToBeCore* node. The source then checks if it still can support passive nodes (it is possible that in the meantime other sources request passive status too). It then replies using a *Confirm* packet which is in turn forwarded to *ToBePassive*. The formerly active source has now become a core active source meaning that it forwards all packets coming from *ToBePassive*. Upon reception of the *Confirm* packet, *ToBePassive* changes its status from active to passive. A passive node has to confirm its passive status periodically or else its associated core node presume that the passive node's status changed to active again thus deleting it from its passive list.

Figure 5.10 shows an example of a mesh topology in DCMP. It can be seen, that the passive source S3 uses its core active source S4 to communicate with the receivers. Unlike in ODMRP, there are no redundant paths between S3 and its receivers, but only a routing tree between S4 and S3. Since

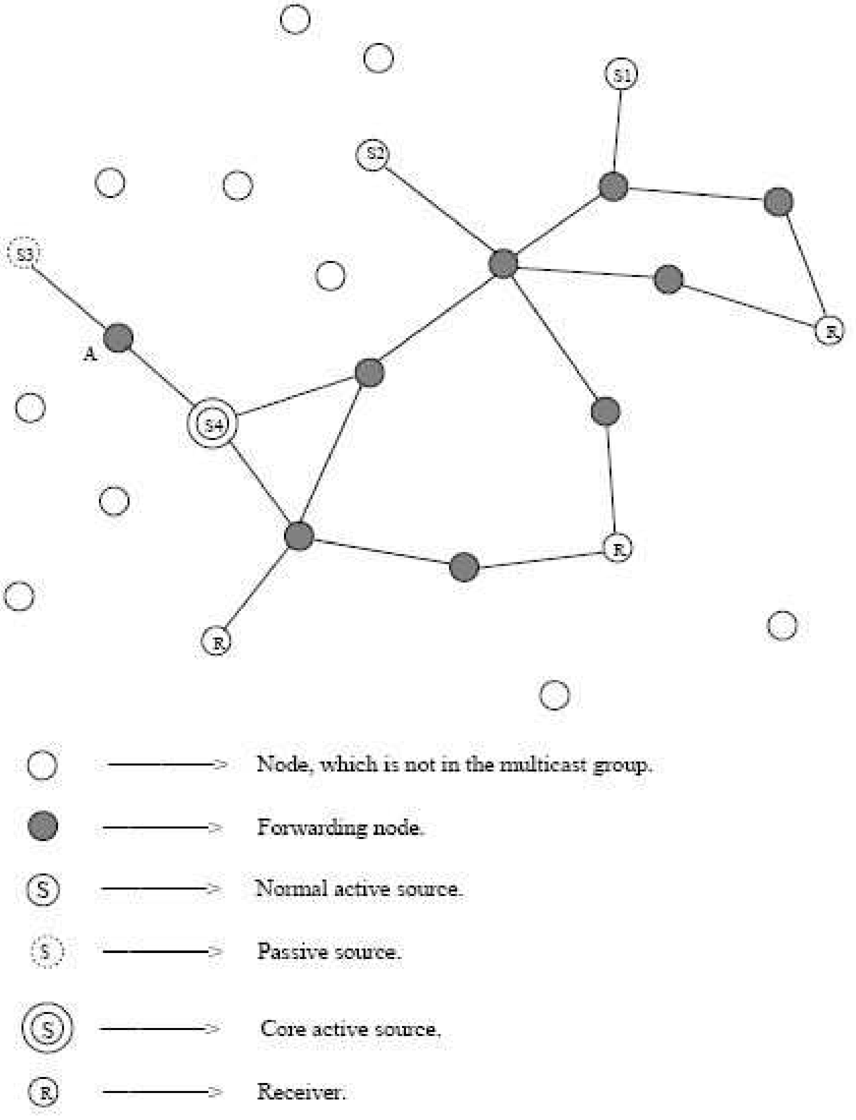


Figure 5.10: Shared Mesh Topology of DCMP[11]

passive nodes do not send out JOIN-REQUEST periodically, there are less forwarding nodes in the mesh thus reducing network load.

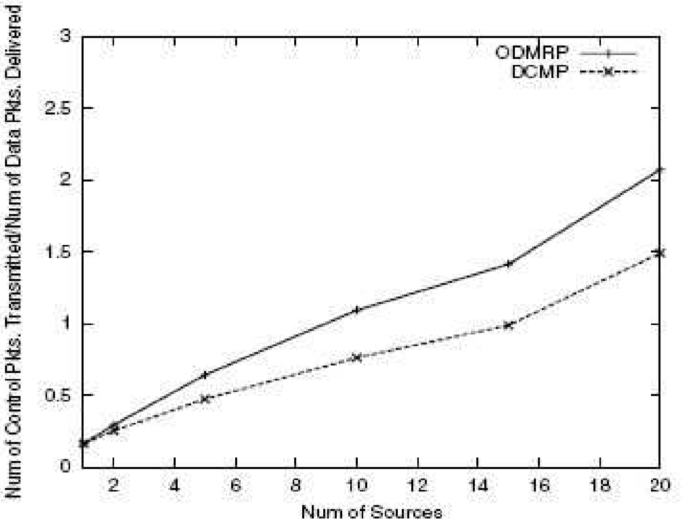


Figure 5.11: Control overhead in DCMP as a function of number of senders [11]

The simulation [11] in Figure 5.11 shows that the control overhead in DCMP can be significantly reduced in comparison to ODMRP. Even with an increasing number of sources, DCMP allows multiple sources to become passive thus decreasing the control overhead. Concerning mobility, the study revealed that the packet delivery ratio for small multicast groups deteriorated slightly. This can be explained by the fact that in a small network the use of a multicast routing tree (i.e., a source becomes passive) affects the transmission robustness, since no or less alternative routes are present. Furthermore, in a larger multicast group a single broken link to a passive source has considerably less impact on the statistic.

The main advantage of DCMP is its scalability. Control overhead does not increase uniformly with an increased number of nodes since a higher node count also leads to more passive sources. The price for the use of passive sources is a decrease in protocol robustness.

5.3.6 AMRoute

The AMRoute protocol [12] tries to combine the key advantages of multicast trees and multicast meshes in a hybrid approach. The main goal is to design a protocol that is at the same time efficient (i.e., it has low control overhead) and robust even at a higher mobility rate.

The key part of AMRoute is the creation of a so-called $user - multicast$ *tree*. All senders and receivers of a multicast session are part of this virtual tree. Between pairs of multicast group nodes (i.e., only multicast senders or receivers) that are located close together, AMRoute establishes a bidirectional unicast tunnel. These unicast tunnels use a subset of the available mesh links in order to forward multicast data along the virtual tree.

Moreover, each multicast group has a *logical core* node which is responsible for membership management and creation and maintenance of the multicast tree. In contrast to its predecessor protocol CBT [48], the logical core node is not preset, meaning it can change dynamically, e.g. due to changes in topology or multicast groups. Neither need the logical core be involved in data forwarding like this is the case in CBTs. These definitions help AMRoute to overcome the problem of a single point of failure in the network.

Figure 5.12 shows an example of 6 multicast group nodes connected by a user-multicast tree. Between nearby group nodes, virtual multicast tree links are established. These links represent a virtual connection of the two nodes but abstract the underlying physical interconnection. Data exchanged between two neighboring multicast group members may be forwarded over multiple physically intermediary nodes. Hence, the physical path of a multicast packet sent via the unicast tunnel can change without affecting the multicast delivery tree as long as there are intact, backup mesh links. AMRoute assumes that the underlying unicast protocol is capable of according route repair mechanisms.

An additional advantage comes along with the decision to connect only group nodes using unicast tunnels. Nodes that are not part of the multicast

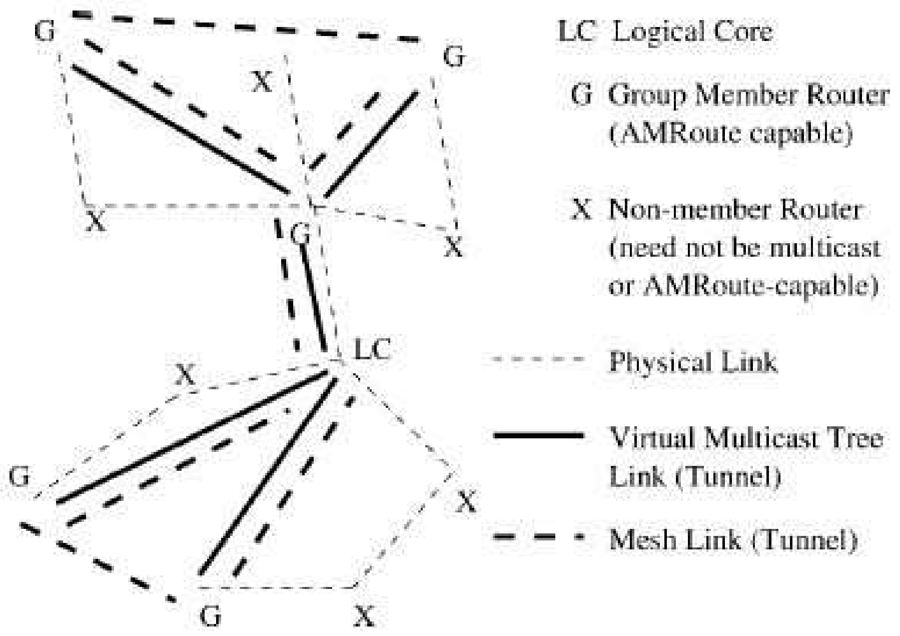

G
G
X
X
G
X
LC
X
G
X
G
X
LC Logical Core
G Group Member Router (AMRoute capable)
X Non-member Router (need not be multicast or AMRoute-capable)
Physical Link
Virtual Multicast Tree Link (Tunnel)
Mesh Link (Tunnel)

Figure 5.12: AMRoute virtual multicast tree [12]

group need not support any multicast protocol nor replicate packets but only need to provide unicast capabilities.

The AMRoute protocol classifies two steps of the user-multicast tree creation.

- Mesh creation

  At the beginning, AMRoute creates a mesh containing all multicast group nodes. This mesh provides redundant connections between the nodes but does not specify data forwarding paths. All members start as core nodes (i.e., each of them forms a one node mesh itself) thus broadcasting JOIN-REQ packets with an increasing time-to-live counter. Eventually, other core nodes of the multicast group will receive these

JOIN-REQ packets and reply using JOIN-ACK packets. Upon reception, a bidirectional unicast tunnel is created between the two nodes. Due to the mesh merger, two cores would exist in one mesh. To overcome this problem, AMRoute specifies a deterministic core resolution algorithm which is used to decide on a single core node in the newly formed mesh.

- Tree creation

  After the mesh has been established, a tree along which the multicast data is forwarded has to be found. This tree consists of a subset of the unicast tunnels found in the mesh creation step.

  It is the task of the single remaining logical core node to initiate the tree creation process. This is done by periodically sending TREE-CREATE messages along the mesh links (i.e., along the unicast tunnels). Other group members receive the message and forward it on all (outgoing) mesh links except the incoming one. Furthermore, all members mark the incoming and outgoing links as tree links. In order to convert tree links to (backup) mesh links, group nodes reply to TREE-CREATE on the incoming link with a TREE-CREATE-NAK message. Upon reception, the tree link is marked as mesh link. Hence, members consider all links on which they do not receive a NAK as tree links.

  As an alternative to the NAK scheme, also an ACK scheme is proposed where data is only forwarded along links if these link are acknowledged. This has the advantage of using only verified links for data forwarding but results in more control overhead.

A inherent problem of AMRoute lies in the tree creation algorithm. Trees are created upon reception of periodically sent TREE-CREATE packet. Due to mobility of nodes and link or node failures, trees may change. As not all member nodes receive the packets simultaneously, some may forward data according to older trees while others rely on the newly created trees. This inconsistent network state can result in routing loops and data loss.

In order to solve at least the problem of packet duplication, sequence numbers for each multicast group and sender are added. Hence duplicate frames are recognized and can be discarded. However, this feature results in additional overhead.

5.3.7 Location Based Multicast

A protocol for Location Based Multicast (LBM) [13] was developed at Texas University.

The main difference of Location Based Multicast to other mutlicast algorithms is the definition of the multicast group. For this purpose, LBM defines a $location - based\ multicast\ group$ which contains a set of nodes currently residing at a defined geographical location. Thus nodes in a particular area (called *multicast region*) at a given time will automatically be part of the location-based multicast group corresponding to this area. So unlike other protocols, LBM manages group membership automatically, that is without any explicit control messages. It is evident, that for location-based multicasting all nodes must be aware of their physical location. Thus, nodes need to be equipped with Global Positioning Systems (GPS). Figure 5.13 shows an example of a multicast region.

The motivation for the use of LBM is the fact that in some cases there is a high correlation between location of the nodes and specific information flow. Thinking of HBA, LBM could be used to turn off all connected devices in a particular room including mobile devices that just happen to be there at a specific time (e.g., turning off all entertainment devices including the portable mp3 player).

The most simplistic implementation of LBM could be done using flooding. A sender S would broadcast a multicast packet that contains the region of the intended multicast group, for example as GPS coordinates representing

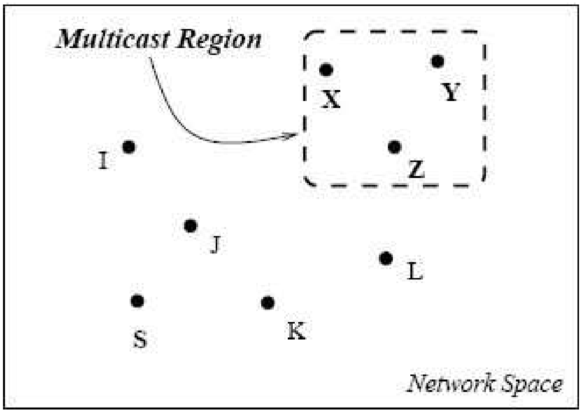


Figure 5.13: Multicast Region in LBM [13]

a closed polygon. Upon reception, location aware nodes know if they are part of the multicast group or not according to their own physical location. Flooding would imply that all nodes in the network receive the message making it highly inefficient. Hence, LBM tries to reduce the number of nodes outside the location-based multicast group that receive the message by definition of *forwarding zones*.

Forwarding zones, as shown in Figure 5.14 are defined areas of the network space, which span at least the multicast region or comprise up to the whole network space. As an enhanced definition to flooding, nodes only forward packets if they are part of the corresponding forwarding zone. The key task of LBM is to specify algorithms that allow nodes to decide whether they are part of the forwarding group or not. A forwarding zone does not mean, that only nodes located within will receive multicast messages, but rather that only these nodes will forward the packet.

- Algorithm 1

  The forwarding zone is defined as the smallest rectangle that includes the multicast region and the sender. In case of the sender being phys-

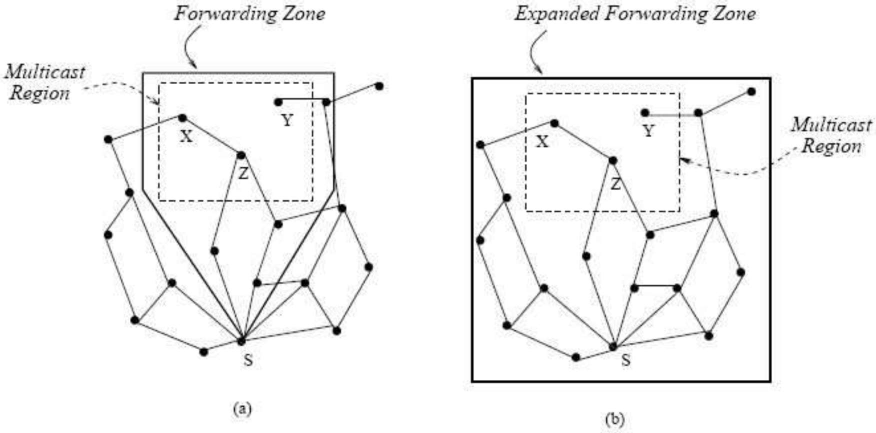


Figure 5.14: (Expanded) Forwarding Zone in LBM[13]

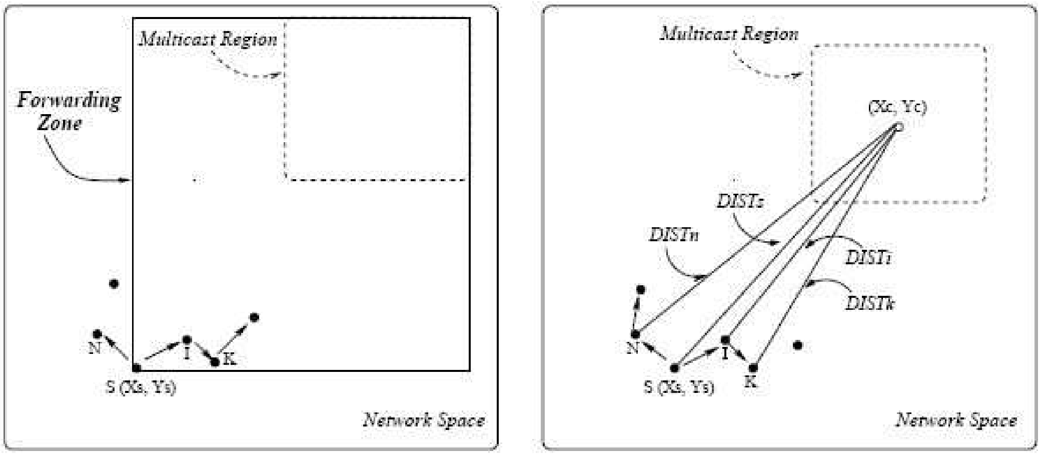
ically located within the multicast region, the forwarding zone corresponds to the multicast region. Otherwise, the forwarding zone is larger in size. At the beginning of a multicast transmission, the sender includes the coordinates of the four corners of the forwarding zone in its message. Hence, nodes are able to decide if they are currently located within the forwarding zone.
In order to improve performance, LBM specifies a parameter δ that is used to extend the forwarding zone by the given amount.

- Algorithm 2

  Here, unlike in algorithm 1, the sender does not include the coordinates of the forwarding zone in its message, but other information. That is coordinates of the multicast region, the coordinates of the geographical center of the multicast region and the coordinates of the sender. If a node receives a multicast packet it determines if it is part of the forwarding group using the following procedure.

1. If the node is part of the multicast region, it accepts the packet. Furthermore, it calculates its distance from the geographical center of the multicast region. If its distance from the sender it greater than the distance from the geographical center, the packet is relayed (the node then replaces the coordinates of the originating sender with its own coordinates). Otherwise the node now knows if the sender is located within the multicast region. If so, the packet is relayed, else it is discarded.
2. The next receiving node on the path only forwards the message, if it is at most δ farther away from the geographical center than the originating node.

Figure 5.15 compares the two multicast schemes. It can be seen that the forwarding path of a multicast message is different using the two schemes.



(a) Location-based Multicast Scheme 1

(b) Location-based Multicast Scheme 2

Figure 5.15: Different Multicast Schemes in LBM[13]

A key characteristic of the location based multicast is accuracy. Accuracy is defined as the ratio of number of multicast members that receive the packet and the total number of multicast members at a specific time. Studies conducted by Ko et al. [13] show, that the accuracy of LBM is comparable to flooding, with LBM having less delivery overhead.

The drawback of LBM is the requirement for GPS in all nodes, making nodes large in size and also more expensive. Also the additional information (coordinates of the region) that has to be transmitted and the GPS based position sensing that has to be done by the (battery-powered) nodes stand in contrast to several design goals of the ideal multicast algorithm.

5.4 Classification

One goal of this diploma thesis is the identification of universal multicast protocol characteristics that can be used to distinguish different design approaches. These characteristics can be used by system designers to choose an algorithm that is most appropriate for the current system.

The classification was done on the basis of the protocols discussed above and literature study [11, 49, 50, 51, 52].

At the beginning, this section explains all features in detail. Afterwards, a mapping of the protocols into the according categories is done.

- Tree based versus Mesh based

  A main criterion which differentiates the multicast protocols is the topology of the delivery structure that is created by the algorithm.

  First, a tree-based structure can be used. Between each sender and its receivers a tree is created. Multicast packets are routed along the tree. Using trees, only a single path between a sender and a receiver exists. This choice increases multicast efficiency but comes at the price of less reliability as there is only one single path at a time. In case of higher node mobility, the consequence is frequent tree reconfiguration. Moreover, in cases of high traffic the single path can become congested more easily.

Second, a multicast routing protocol may create a mesh of nodes to be able to deliver multicast messages. Apparently, the control overhead for creation of the mesh increases. Yet, mesh networking may (and mostly does) offer multiple paths between sender and receiver thus also increasing reliability and robustness especially against node mobility. Another advantage of a mesh is that it can accommodate different senders simultaneously. This implies, that in contrast to trees, construction overhead can be reduced. In other words, one global mesh may be established for all possible multicast groups at once.

- Soft state versus Hard state

  The state approach denotes the scheme according to which the multicast groups are maintained. State maintenance is necessary to detect nodes that want to become part of a multicast group and nodes that left the multicast group.

  Using a soft state approach, a protocol periodically has to update the multicast groups and all routes belonging to the groups as they time out automatically. Refreshment is done by periodically flooding the network with control packets. In other words, a current image of the multicast state (i.e., all multicast members and associated paths) is captured. This image is valid only until the next flooding. Nodes (and according paths) that do not answer are then automatically removed from the state. Clearly, the control overhead is increased through control packet flooding. Also the multicast state can become outdated between two refreshments. However, if the time span between the control packet flooding is small (the value usually can be defined in the protocol implementation), multicast group maintenance is done very accurately.

  Using a hard state approach, changes in group membership have to be reported explicitly. A state is regarded as correct as long as no other information is received. In wireless ad-hoc networks, this has the advantage of less control overhead. Problems can arise if messages (e.g. a group-leave message) get lost as this leads to orphaned nodes. Thus, a hard state approach is mostly used in connection with reliable multicast protocols. Ji et al. [51] propose mechanisms to enhance hard state protocols.

The advantage of a soft state over a hard state approach is the configurable granularity of group maintenance. Application designers can determine a suitable value for the flooding interval at system deployment and thus guarantee a predictable system behavior (e.g. a mean packet delivery ratio of a certain percentage) under certain conditions.

- Table-driven (proactive) versus On-demand (reactive)

  Multicast protocols can be distinguished by the way they store routing information.

  In a table-driven approach, each node sets up and maintains a routing table itself. This behavior is also known as a proactive scheme. In this table, a path to all destination nodes is stored. Hence, a node a priori knows the path to each other node and therefore can forward frames immediately upon reception. Problems arise if the topology changes. The tables in all nodes need to be refreshed to reflect these changes. This can be done either globally (distribution of the table is necessary) or locally. As it is no trivial task to synchronize all nodes to locally update their tables simultaneously, inconsistent network states can occur. This can lead to routing loops and also packet loss. Another drawback is that this approach requires sufficient space in all nodes to store the routing tables. These tables moreover contain routes that may never be used (and nevertheless are kept up to date).

  Using an on-demand scheme, a multicast routing protocol establishes a path only if a sender has to send a multicast message. The on-demand scheme is also referred to as reactive protocol acting upon creation of a message. The on-demand approach offers some major advantages over table-driven protocols. First, the additional (unnecessary) effort to store tables in all nodes can be omitted. This choice reduces both control traffic in the network and relieves nodes of storage concerns (which of course allows production of cheaper nodes). Secondly, on-demand based protocols scale better to an increased node count as route maintenance is no longer necessary. A drawback can be seen in the fact that on-demand establishment of routing paths results in a delay of message delivery time.

- Global versus Local

  In literature, global and local is used to characterize the amount of information available to nodes in the network. If all nodes are aware of the whole network topology (i.e., all other nodes and interconnections) this is referred to as global approach. Hence, in a local approach, a single node is only aware of part of the network topology.

- Distributed vs. Centralized

  Operation of a multicast routing protocol is based on the availability of information on multicast members, senders and the network itself. Hence, if all information concerning a multicast group is stored in just one node, the protocol has a centralized approach.

  A centralized approach has the advantage, that upon changes, only one information base exists that has to be updated. However, a single authorative source for routing purposes also implies a single point of failure and in case of high traffic bottlenecks can occur. Additionally, higher node counts in the network may be impossible because of limited storage in the control node.

  In a distributed approach, all nodes have to store at least some information on a multicast group that is necessary for successful operation of the multicast routing protocol. Decentralization of the information helps to increase performance and leverages storage concerns. However, special precautions must be taken to ensure a synchronous state in the whole network even after changes.

- Source initiated versus Receiver initiated

  The characteristic of source and receiver initiated multicast communication refers to group management.

  A protocol that specifies source initiated (also sender initiated) construction is based on the premise that the sender of a multicast group has to discover its recipients. This can be done by flooding the network with control packets looking for paths to the receiver.

  In contrast, receiver initiated multicast protocols require the recipients to join multicast sessions themselves. For example, the receiver-initiated soft-state probabilistic multicasting protocol [53] specifies that receivers have to subscribe to multicast sessions. Available multicast sessions are advertised by the senders using beacon telegrams.

  Although similar to sender initiated flooding, the control overhead in terms of data bytes sent can be reduced since the beacon telegrams are small in size. Additionally, only recipients interested in a particular multicast session have to join.

Apart from the characteristics mentioned above, it is possible to map multicast protocols according to their topology into a tree structure. This tree depicts oppositional design choices as well as connections and dependencies between the approaches. However, this classification does not contain all identified criteria but is discussed here for better understanding.
At the beginning of this Chapter the differences between a mesh and a tree topology were explained. Based on this, a further differentiation can be made. Figure 5.16 shows this classification.

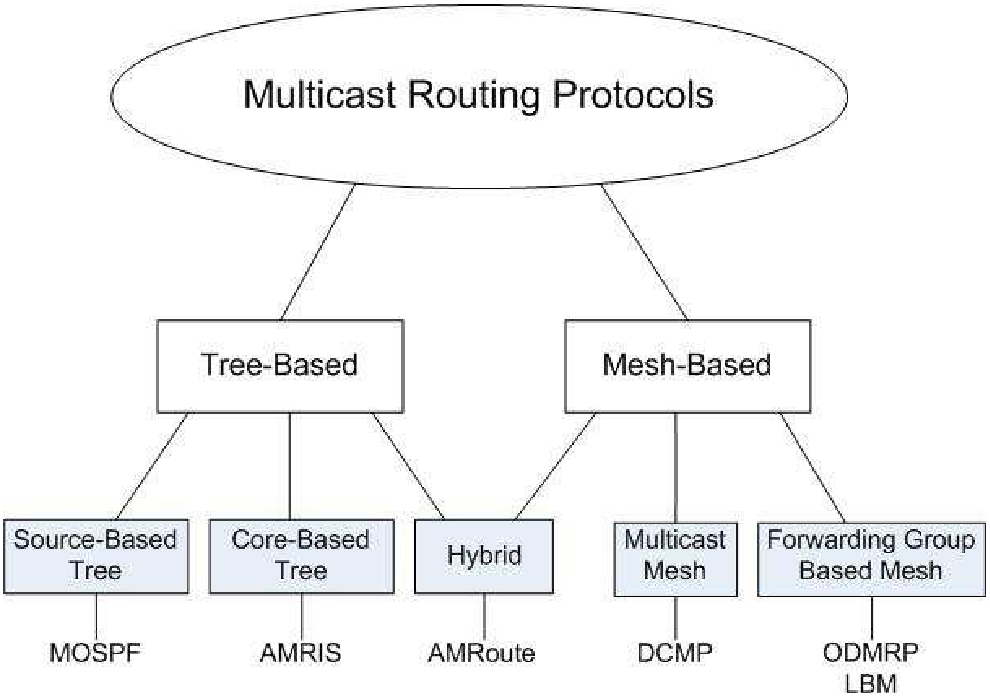


Figure 5.16: Classification of Multicast Routing Protocols

Multicast trees can be $source - based$ or $core - based$. In a source-based approach, each node keeps a routing table with the shortest paths to each other node in the network. In a core-based multicast protocol, for each multicast group a particular core node exist. This core node creates the routing tree containing all multicast members.

Multicast protocols can also be mesh based. A protocol that establishes a mesh between all nodes without any further features, is called *multicast mesh*. Based on the mesh topology, $forwarding$ *group*-based multicast protocols exist. These define a set of nodes called forwarding group which is responsible for forwarding the multicast message.

Additionally, there are also multicast routing protocols that try to combine the advantages of tree-based and mesh-based protocols into so-called *hybrid* protocols.

5.5 Summary

This section summarizes the features and characteristics of the multicast protocols discussed in this chapter.

It is noticeable, that the summary table does not contain any table driven multicast algorithm. The reason for that is, that table driven algorithms are not suited for the use in wireless ad-hoc networks as they require a lot of storage space (for routing tables and routes) and frequent updates. Using resource limited nodes, employing table driven protocols would make nodes more expensive (due to the increased storage demand) and severely influence battery lifetime. Hence, table driven protocols are being used in static networks and in combination with nodes less dependent on resource saving. A typical example of a table driven routing protocol is the Destination-Sequenced Distance-Vector (DSDV) protocol [54].
Also no hard state protocol has been discussed as these protocols are also not well suited for wireless applications. An example of such a protocol would be the Revised Internet Stream Protocol [55].
The characteristics *dependency on unicast algorithm* and $loop\ free$ have to be seen as special features. Nevertheless, for the sake of completeness, the summary table also lists both of them as they are valuable information when having to choose a multicast protocol.

In summary, all presented multicast algorithms can deliver good performance when used in the right situation.

Flooding comes along with high overhead but at the same time provides high reliability and can be easily implemented.

MOSPF computes shortest path trees routed in the multicast sender. The tree branches terminate at multicast group members. Through the creation of multicast delivery trees, the overhead can be significantly reduced compared to flooding. Because MOSPF was initially designed for (static) IP networks, it does not perform well in networks with mobile nodes.

The AMRIS protocol is based on a similar approach as MOSPF, namely construction of a multicast delivery tree. But unlike MOSPF, it is designed to perform better under node mobility. This is achieved by assignment of numbers to the nodes that reflect their depth in the delivery tree. These numbers are used to repair the delivery tree locally instead of completely recomputing it.

ODMRP is based on the creation of a mesh as multicast forwarding structure. The mesh based approach comes along with increased robustness even at high node mobility, but requires more control overhead than a tree based protocol. Furthermore, delivery paths of multicast messages in a mesh may not be optimal. ODMRP has been designed as a multicast protocol but can also be used for unicast communication thus making it interesting for resource limited nodes.

The dynamic core multicast protocol (DCMP) expands the idea of multicast meshes by connecting multiple core based trees through mesh links. This novel approach supports local route repair that does not necessarily affect the whole network. DCMP allows some source nodes to be passive network members. The main advantage of this design choice is the reduced overhead, as these nodes do not send out control packets periodically. This makes DCMP perform better even at higher node counts.

AMRoute is a completely hybrid protocol. It creates a virtual multicast tree between the source nodes and all receivers, where all delivery paths use mesh links. In other words, an abstraction of the underlying mesh links towards a tree based delivery structure is done. While at the first glance very promising, studies have shown that the direction taken by AMRoute can lead to

serious problems. Node mobility may require tree reconfiguration, which is not done in a synchronized way. The resulting inconsistencies in the forwarding paths lead to routing loops and probably data loss.
Location based multicast is – although not perfectly applicable to home and building automation networks – an interesting concept. In LBM, group membership is not handled explicitly using control messages, but defined at the time of multicast messaging using the physical location of the nodes. Clearly, the drawback is that all nodes have to be equipped with GPS hardware.

Table 5.1: Summary of multicast protocol features I

	Topology	**State**	**Table driven vs. On-demand**
Flooding	Mesh	Stateless	On-demand
MOSPF	Tree	Soft state	On-demand
AMRIS	Tree	Soft state	On-demand
ODMRP	Mesh with forwarding groups	Soft state	On-demand
DCMP	Mesh	Soft state	On-demand
AMRoute	Hybrid	Soft state	On-demand
LBM	Mesh with forwarding groups	Stateless	On-demand
ZigBee	Tree	Soft state	On-demand

Table 5.2: Summary of multicast protocol features II

	Source vs. Receiver initiated	**Dependent on unicast**	**Loop free**
Flooding	—	No	Yes
MOSPF	Source	No	Yes
AMRIS	Source	No	Yes
ODMRP	Source	No	Yes
DCMP	Source	No	Yes
AMRoute	Receiver	Yes	No
LBM	Source	No	Yes
ZigBee	Source	Yes	Yes

Chapter 6

Multicast in ZigBee

Although specified for both industrial and HBA applications, ZigBee is a wireless protocol that fulfills all requirements of home and building automation almost perfectly. Especially in home and building automation many scenarios exist in which group communication is essential. Possible applications reach from collectively turning off all lights in a room, floor or building to activation of locally stored user-based scenarios.

All of these scenarios can also be accomplished using multiple unicast frames. But as explained in the previous chapters, efficiency is important in resource-limited networks.

The ZigBee specification released in 2004 [7] does not specify any satisfying methods for group communication or multicast except the coordinator reflection "multicast" which is based on the coordinator's binding table. However, a broadcast algorithm exists, which can be used to build a flooding based multicast protocol.

For better understanding, the ZigBee broadcast algorithm is explained here.

6.1 ZigBee Broadcast algorithm

In ZigBee, different broadcast types exist. According to a predefined value in the message control header, a broadcast may be intended for all devices part of the PAN, only coordinators and routers or all devices that have their receivers enabled constantly (ZigBee end devices may opt to turn off their receivers when idle to reduce power consumption). Hence, a device does only accept incoming broadcast frames if they match its device class.

Any device that is part of the ZigBee network can start a broadcast transmission. To do so, the frame is addressed to the broadcast short address $0xffff$. In order to keep track of broadcast transmissions, the ZigBee coordinator and ZigBee routers store a so-called Broadcast Transaction Record (BTR) for each broadcast (either initiated locally or received from a neighboring device) in the Broadcast Transaction Table (BTT). The broadcast transaction record itself contains at least the broadcast sequence number and the source address of the broadcast. A BTR is valid for a limited time only and expires afterwards.
Upon reception of a broadcast frame from a neighboring device, a device checks its BTT for a matching entry. If an entry is found (i.e., the device has already received this particular broadcast frame before), the device marks its neighbor as having relayed the broadcast frame and discards the message. For this purpose, all nodes keep a neighbor table which contains all other nodes that are in their direct wave reach (i.e., located within one hop of the node). If no entry can be found in the BTT, a new entry is created and the neighboring device again is marked as having relayed the frame. The received broadcast frame is then passed to the next-higher layer. Furthermore, if the radius field (which specifies the maximum number of hops a frame can travel) contains a value greater than 0, the broadcast frame is relayed a specified number of times.

Figure 6.1 shows a typical broadcast transmission sequence chart. As can be seen, the devices keep track of their neighbors relaying the broadcast message. If single neighbors do not relay the broadcast within a specified amount of time, a device again rebroadcasts the frame to ensure delivery ratio. This scheme is called *passive acknowledgement* and improves packet delivery ratio. However, the cost of higher overhead for storing and updating the neighbor table has to be considered.

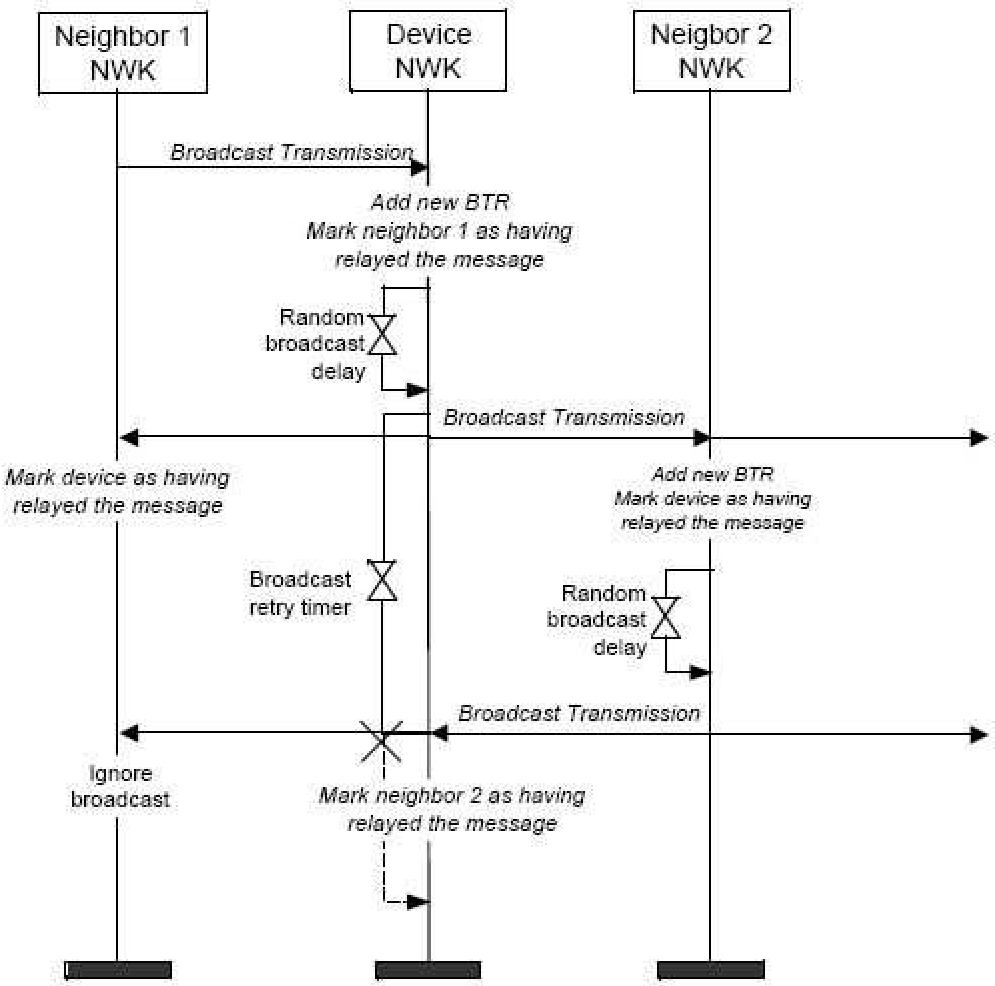


Figure 6.1: Broadcast Transmission Message Sequence Chart[14]

6.2 ZigBee Multicast algorithm

In late 2006, a revised ZigBee protocol specification [14] was made available to the public. Nodes built according to this specification now provide support for multicast communication. The ZigBee 2006 multicast fundamentals are summarized in this section.

According to the ZigBee specification, all devices keep a multicast (also called group id) table. This table contains a list of all multicast groups a device is part of. Currently, only data frames can be sent as multicast messages. Unlike other protocols, ZigBee does not require a sender to be part of any multicast group. In other words, any node can send multicast messages to any group address without requiring any special precautions.

The ZigBee protocol differentiates between 2 modes of a multicast transmission. Each mode is indicated in the message using a flag.

- Member mode multicast

  A member mode multicast is initiated if the sender is part of the multicast group. For this purpose, each sending device checks its multicast table for a matching entry. Multicast messages that are sent in non-member mode (i.e., the multicast message originates at a node that is not part of the recipient multicast group), change to member-mode once they are received and forwarded by a member of the multicast group. Hence, the member mode is used for delivery of multicast messages within the multicast group.

  Upon creation, a member mode multicast is also recorded in the broadcast transaction table of the sender as if it was a broadcast. This means, a broadcast transaction record is created containing the local (originating) node's address as source and a new multicast message sequence number.

  Upon reception of a member-mode multicast, a device (this receiving device may not be a multicast group member) first has to check its

BTT. If a BTR with the same sequence number and source address exists, the incoming frame is discarded. Otherwise, a new BTR is created. Furthermore, it is required to check whether the device is part of the intended multicast group. If so, the received message is passed to the next higher layer and also set to member mode. Additionally, the multicast message is broadcasted multiple times to the broadcast address $0xffff$.

If the recipient node is not a member of the multicast group, it is necessary to limit the (broadcast) propagation of the multicast frame. For this purpose, each multicast frame contains a $NonMemberRadius$ value. This value specifies the number of times a multicast frame will be relayed by non-member nodes. If the value is zero, the frame is discarded. It is important to note that member-mode multicast frames are never changed to non-member mode even if received and forwarded by nodes that are not part of the multicast group.

- Non-member mode multicast

  A multicast message has the status non-member, if the sending device is not a member of the multicast group. The term sending device is used here to either refer to the node where the multicast has originated or the last forwarding node of the multicast message. A non-member mode multicast is initiated if a device wants to send data to a multicast group that it is not part of. In case of an existing routing table entry for the destination address, the frame is sent as unicast to the next-hop address. If no matching entry is found in the routing table, the (unicast) route discovery procedure is invoked. The pending multicast frame may be buffered until route discovery is completed. If route discovery is disabled, the multicast transmission is reported as failed. Upon reception of a non-member multicast frame, a device has to verify if it is part of the multicast group. If so, the mode is changed to member mode and the frame is passed to the next higher layer. The frame is then processed as a member-mode frame. If no matching entry is found in the multicast member table (i.e., the device is not part of the multicast group), the device checks its routing table for a matching entry. If no entry exists, the frame is discarded. Otherwise, the frame is sent as acknowledged unicast to the next-hop indicated in the routing table.

6.3 ZigBee Routing algorithm

In ZigBee, routing is performed only by the network coordinator and ZigBee routers, while end devices never participate in routing. In order to find optimal routes (i.e., routes with lowest path costs), ZigBee employs a path cost metric based on the link quality index specified in IEEE 802.15.4.

Routing capable devices may keep a routing table in which routes to specific destination addresses are stored. Each route has an assigned status to indicate whether the route is active, inactive or still under discovery. In addition to the routing table, devices should also keep a route discovery table. While the routing table stores information that is rather long lived, the route discovery table holds information only as long as necessary for route discovery.
Multicast route discovery is performed with regard to a source address of a device and a multicast (destination) group address. At the beginning of route discovery, the initiating device creates a routing table entry with the status $DISCOVERY\_UNDERWAY$, or overwrites the status of an already existing entry. Furthermore, a route discovery table entry with a new route request counter value is created.

To initiate route discovery, a *route request command frame* is assembled and broadcasted. At the initializing device, the broadcast is repeated a specified number of times, each separated by a specified time interval.

Only devices with routing capacity accept a multicast route request command frame, all others discard these incoming frames. Upon reception, a device checks if it is part of the requested multicast group. If so, a route discovery table entry is created and the link cost to the previous node is calculated. If such an entry already exists, the one with less link costs is kept. The device then creates a *route reply command frame* directed to the source of the route request.

If the receiving device is not part of the multicast group, a route discovery table entry is created or updated. This new entry gets assigned a specified route request timeout, which is used to remove entries when expired. The route request is then rebroadcasted.

In multicast operation, it is possible (and probable) that a device receives more than one route reply that would cause a change in the routing table (i.e., an additional link in the device's multicast delivery tree would be added making the tree more complex). To avoid this sub-optimal behavior, a device waits a specified amount of time and collects all route replies. After a timeout, only the best route is kept and added to the routing table.

The route reply command frame is finally received (after having been relayed by intermediary nodes) by the request initiator. The forwarding path has then been built using *reverse route establishment*.

Routing itself is done on the basis of the information stored in the routing table. Devices that receive a frame check their routing table for an entry corresponding to the destination address of the received frame. If a matching, valid (i.e., not in status inactive or failed) entry is found, the frame is relayed to the corresponding next-hop address. Here, the relaying device substitutes the originating source address with its own address, and the destination address with the found next-hop address. ZigBee calls this routing scheme *source routing*.

If relaying did not succeed (e.g., because of failed route discovery or broken links), a ZigBee network may employ *hierarchical routing*. This scheme is used only if source routing is not possible for any reason (also if devices have no routing capability). Using hierarchical routing, devices route frames along the tree to descendant devices. ZigBee specifies a logical expression devices use to determine their descendants.

6.4 Discussion

Merged together, the ZigBee routing algorithm, the multicast algorithm and the broadcast algorithm described above form a multicast routing protocol that can be compared to the protocols presented in Chapter 4.

In fact, ZigBee uses the Ad-hoc On-demand Distance Vector Routing protocol (AODV) published in [56, 57] for unicast routing. While a multicast capable version of AODV has been proposed in [58, 59], ZigBee specifies its own multicast algorithm.

This route discovery phase of the ZigBee multicast algorithm is the same as described in AODV. According to the multicast algorithm, multicast data frames are forwarded along the same path as a unicast frame would have been (i.e., the frame is routed along the shortest path between sender and one multicast member node using the standard unicast routing algorithm). In other words, a multicast frame is sent in the same way as an unicast until it reaches a member of the multicast group (also called non-member mode multicast). The recipient multicast-member node then replicates the message and broadcasts it (member-mode multicast) so that it reaches all multicast member nodes. To avoid broadcast propagation throughout the network, nodes that do not belong to the multicast group can only forward the path a limited number of times.

A classification of the ZigBee multicast algorithm is contained in Table 5.1 and Table 5.2, respectively. ZigBee constructs multicast delivery trees. The branches of the trees correspond to the wireless links along which the frames are forwarded. Multicast routing is done on-demand only and has to be initiated by the source node. Multicast routes are updated using a soft-state approach. ZigBee routing highly depends on the underlying unicast routing protocol, AODV, which guarantees loop free routing paths. A drawback of the ZigBee multicast algorithm is, apart from its dependency on the unicast algorithm, the localized flooding approach. At the time a multicast packet reaches a group member it is broadcasted in order to reach the other members. This approach is only advantageous (i.e., efficient) if the group members of a specific multicast group are located physically close together.

In case of distant group members, many non-member nodes will be required to rebroadcast the multicast message, thus leading to considerable overhead. In the worst case, the limited non-member multicast propagation scheme could even not be capable of successfully bridging the distance between intended recipients, thus resulting in data loss.

However, in a couple of typical home and building automation applications such as lighting control, the assumption that only closely located nodes will be interested in the same information can be true. In contrast to that, an HVAC application (which also is a typical example of ZigBee building automation) will probably need to communicate with nodes that are distributed over the whole building. In this case, the ZigBee multicast scheme exhibits increased overhead. Hence, using multicast in such network configurations requires alternative solutions.

Chapter 7

Alternatives to ZigBee Multicasting

As pointed out in the previous, the multicast protocol specified in the latest ZigBee specification is not suited for all purposes. This chapter identifies one possible alternative and discusses advantages and drawbacks of this scheme.

7.1 Aspects

As outlined in Chapter 2, a main challenge for wireless networks used in home and building automation is interference. Mesh technologies cannot reduce interference but alleviate its effect through providing redundant communication paths.

The packet delivery ratio in mobile ad-hoc networks is degraded due to high node mobility which requires frequent path updates. In HBA, wireless sensor networks are employed, in which nodes exhibit little or no mobility. However, wireless links may become unavailable because of interference. Tree based protocols offer only a single communication path between sender and receiver. The links on this path are prone to break, resulting in data loss.

This justifies the use of more reliable mesh-based multicast communication technologies.
The identified requirement of a mesh based multicast protocol narrows the possibilities considerably as the MOSPF, AMRIS and AMRoute protocol are all tree based. Of the remaining protocols, flooding comes along with unnecessary high overhead. Location based multicast is a tempting approach in HBA systems. But LBM requires all nodes to be equipped with GPS systems for position sensing thus making nodes more expensive and power consuming. An alternative approach would be pre-programming the physical location into each device at setup time. Apart from being cumbersome and time-consuming, this would not solve the second problem either. In LBM multicast groups are nodes that are located physically close. There is no possibility of excluding single nodes that are physically located in the multicast region from the multicast group. Thinking of distributed applications such as HVAC, it is not possible to specify the intended multicast group exactly.

The two remaining protocols, ODMRP and DCMP, are closely related as DCMP is an advancement of ODMRP. ODMRP and DCMP both use a soft state approach which requires periodic flooding of control packets. In ODMRP this control overhead increases steadily with the number of sources. As HBA networks can have a high node count, this characteristic of ODMRP is not acceptable. DCMP reduces the control overhead through the use of passive sources, and thus also scales better. Since less control traffic and passive nodes also lead to less resource usage, DCMP is the best choice among the discussed protocols for the use in home and building automation networks.

7.2 DCMP and ZigBee

When using the dynamic core multicast protocol in combination with ZigBee, several issues have to be taken into account. First of all, an underlying mesh topology of the ZigBee network is assumed as this will be the most common one in HBA installations. Also pure tree based installations may exist, but this topology is predestined for a hierarchical routing scheme. Additionally, a mesh based multicast protocol may only be applicable to tree or peer-to-peer structures with considerable limitations. By limiting the topology to mesh, beacon-oriented communication need no longer be considered as it must not be used according to the ZigBee specification. Nevertheless, power saving modes of the ZigBee end devices can still be used. The end devices are still associated with a ZigBee router or coordinator and enter sleep modes when possible. Using indirect data transfer, the devices do not wake up periodically for the beacons but at an application defined rate.

DCMP requires nodes to have assigned a unique node ID. This node ID is required for path creation and transition from active to passive source. In ZigBee the unique serial number of each device can function as node identifier. This choice also contributes to shorter frames because the (unique) extended address is contained in every ZigBee frame.

Setup of group membership is neither explicitly specified in DCMP nor in ZigBee. The ZigBee multicast scheme states, that all devices are required to keep a multicast table containing a list of multicast groups the device is part of. This presumption can also be applied in this mapping.

Employing any multicast scheme in ZigBee always has to be done under the premise of resource conservation. This is especially important if resource limited nodes (i.e., *end devices* in ZigBee nomenclature) are part of the network. Hence, this mapping tries to reduce resource consumption as far as possible. For consistency, the same nomenclature of the control packets as in DCMP is used here.

At the beginning, DCMP requires a source to flood the network with JOIN-REQUEST packets in order to discover the multicast receivers. In ZigBee, this can be accomplished using the provided broadcast scheme described in Section 6.1.

End devices in ZigBee may not be able to receive this broadcast immediately because their receivers are not enabled permanently. As in standard unicast communication, these end devices are associated with a ZigBee router. This router has to buffer all incoming data for its children. The end devices wake up at an application defined rate and poll for new messages. This indirect communication scheme results in an increased setup time for the multicast path as the multicast initiator has to wait longer for Reply packets from these end devices. However, the time span can be bounded exactly by specifying a certain poll rate in the application.

DCMP creates multicast routing paths using a reverse path creation algorithm. In DCMP all devices except end devices have to keep a routing table containing a list of source addresses and the corresponding next-hop address for routing purposes. This is the same scheme as ZigBee uses for unicast routing and also proposes for its own multicast communication scheme. Hence, standard ZigBee routing tables are simply expanded to also store multicast routing information.

Intermediary nodes that may be used for data forwarding form forwarding groups. In ZigBee, only mains powered routers or coordinators will assume the role of a forwarding group node because end devices need not participate in routing. These routers have to keep a forwarding group table which contains entries for each multicast group the node is part of and a timestamp when the node was refreshed last. Clearly, this table implies additional overhead and resource consumption. Nevertheless it is necessary for maintaining the multicast mesh which provides increased reliability.

Until now, DCMP was mapped only to routers or coordinators. But ZigBee also specifies end devices, which have to be associated with a router or coordinator, thus forming a parent-child relationship. In a typical HBA network, also end devices will be multicast sources and receivers (e.g., a switch controlling multiple lights on a floor). In principle, end devices that are sources, can employ the same multicast algorithm as proposed for routers and coordinators. However, they would be required to keep their receivers enabled until all JOIN-REPLY packets from the group members have been received. Additionally, periodic updates of the multicast paths using flooding would be necessary. Here, the choice of DCMP instead of the more simple ODMRP starts to pay. DCMP allows sources to change into passive mode which relieves them from periodically refreshing their multicast group. Periodic JOIN-REQUEST flooding is then done only by the core active node. Multicast data originating from passive sources is forwarded by their associated core active node.

In other words, a change to passive mode is very attractive for end devices, as they only need to confirm their passive status regularly but need not wait for any replies thus prolonging their stand-by time. The confirmation packet is at the same time used to setup and confirm the forwarding route to the core active node. It is a requirement of the ZigBee protocol that end devices are associated with a router or coordinator. As shown in Figure 7.1, the (parent) router is at the same time the intermediary node (i.e., forwarding group node) on the path to the core active source. The algorithm a source node executes to change into passive mode is the same as in standard DCMP.

As stated before, end devices can also be multicast receivers. To ensure multicast delivery, it is necessary that the parent router is aware of all multicast groups its child nodes are part of. For this purpose it is required that the parent device keeps a table containing a list of its end devices and their multicast memberships. Upon reception of multicast data frames addressed to one of its child nodes, it stores them until being polled by the child node.

Figure 7.1 shows an example of a ZigBee network using the DCMP multicast protocol. At position 1 a ZigBee end device (ZED) which is a DCMP

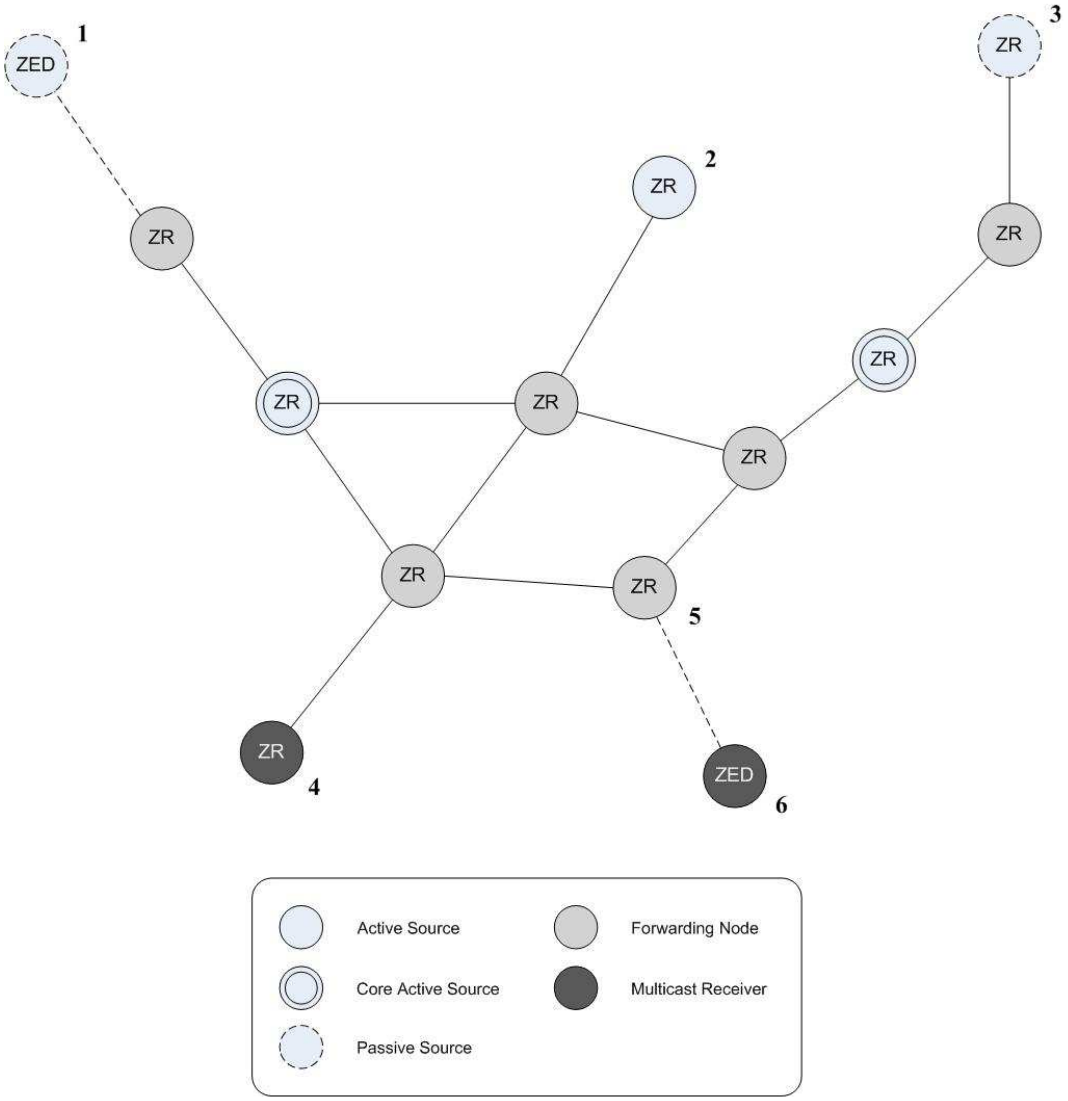


Figure 7.1: DCMP Multicast Mesh in a ZigBee Network

passive source is located. The end device is a child node of the neighboring ZigBee Router (ZR). Another ZigBee router assumes the role of a core active node. The ZED is connected with the core active node via its parent node. This parent node is part of the DCMP forwarding group for this particular multicast source. Communication between the ZED and its parent is based on periodic polling where the child requests transfer of pending data. The advantage of this particular multicast scheme is that the active core ZigBee router maintains multicast links also on behalf of the associated ZED. Hence, the ZED can enter a sleep mode and wake up only to check for pending data and refresh its passive status. The fact that the ZED acts as multicast source also means that it has to receive all JOIN-REQUESTs sent out by other nodes. These JOIN-REQUEST packets are important as they advertise multicast sessions of other sources which the device may be interested in and also are necessary to change into and maintain passive mode. However, an end device cannot receive frames directly but polls all data from its parent.

Position 2 shows the most simple setup of a ZigBee-DCMP multicast device. A ZR has the role of an active source. As routers are mostly mains powered it can fulfill the requirement of flooding control packets periodically in order to maintain the multicast paths without any special precautions.

The node at position 3 is a ZR that is also a DCMP passive source. In relation to multicast communication, the node is passive, i.e., it does not send out periodic beacons itself. But in contrast to the constellation shown in pos. 1, the ZR is not a child of any other node. Hence, the intermediary node between the core active source and the passive source is only a forwarding node.

The ZigBee router located at position 4 is a multicast receiver. As in position 2, this shows a simple DCMP network setup.

Finally, as shown in position 6, also end devices may be multicast receivers. Again, the ZED polls its parent router at a predefined rate. In order to guarantee good performance of DCMP, a ZED will have to poll its parent more often in a multicast-capable networks than in purely unicast-based networks. This is necessary to stay informed about sources advertising groups with the help of JOIN-REQUESTS that the ZED may be part of.

For working multicast communication, the ZR in position 5 has to know which multicast groups its child is part of. Upon reception of a multicast message, the ZR first determines if it or any of its children are part of the multicast group. If at least one child is an intended recipient, it buffers the frame and delivers it to the child as soon as possible (i.e., at the next child-initiated polling).

For good performance of multicast communication it is required that end devices wake up periodically from their sleep mode to check for pending data. A ZED which has multicast data to send (i.e., a multicast source) will try to become a passive source as this reduces the time it is required to be enabled. In ZigBee networks with many designated multicast sources, end devices should be privileged over ZigBee routers when requesting passive status as a passive status is more valuable to them.

The main advantage of this scheme over the ZigBee 2006 multicast protocol is that redundant paths are available that make communication more reliable. In contrast to ZigBee, DCMP does not broadcast multicast packets but delivers them using unicast links as long as possible. Broadcasts are used in DCMP only for path initialization and maintenance. However, in static networks the broadcast interval will be long compared to mobile networks thus reducing control overhead.

Chapter 8

Conclusion

Comparison of the different wireless protocols has shown that not all of them target the same market. While IEEE802.15.4/ZigBee can be successfully deployed in both homes and buildings (and can also be found in industrial surroundings), Z-Wave is tailored for the home use exclusively. Other protocols, especially EnOcean, with its revolutionary energy harvesting technology, will probably be used in context of other wireless protocols than as stand-alone systems. However, it is the responsibility of a system designer to choose the most suitable protocol for a given task. The protocol overview outlined in Chapter 3 provides the necessary information.

Furthermore, it has shown that a native protocol extension, e.g., extension of KNX with KNX RF, need not exhibit a better overall performance than hybrid systems such as the tunneling approaches presented in Chapter 4. Although not perfect yet, these tunneling devices provide wireless control for existing wired automation systems without requiring extensive configuration effort.

A possible hook for improvement of these hybrid installations comes in form of multicast communication. Using multicast communication instead of broadcasts or multiple unicasts significantly reduces the control overhead of the network. While multicasts are commonly used in wired home and building automation networks (e.g., KNX), their wireless counterparts often do not even specify any group communication support. This may also be rooted in the fact that special precautions must be employed in networks with highly fluctuating link quality between the nodes.

Based on this motivation, different multicast algorithms were analyzed for their suitability in wireless HBA networks. As many different approaches have already been proposed, the protocols were classified and different criteria of the ideal algorithm were identified. Finally, based on the results of Chapter 5, a mapping of the Dynamic Core Multicast Protocol to ZigBee was done.

In the future, starting from a ZigBee implementation with multicast support, also the tunneling devices could be developed further. A main task for future versions is the integration of communication security methods. Especially in hybrid networks, the wireless part is prone to attacks as no physical connection has to be established first.

Additionally, performance comparison of the presented tunneling solutions merits attention as well as the emergence of new wireless standards has to be monitored. This is especially important as further development of wireless protocols may also bring along advancements in reliability, multicast support and security issues.

Also the robustness of wireless communication technologies with regard to interference is an important issue. However, existing reports and comparisons are typically biased and seldom take the differences between the US and European sub-GHz ISM bands into account [60, 61]. Conducting such comparisons on a sound, objective basis would provide important information to prospective users.

Bibliography

[1] F. Schmidt and M. Heiden, “Wireless sensors enabled by smart energy - concepts and solutions,” Retrieved September 15, 2006, from http://www.enocean.com, 2006.

[2] Nanotron Technologies GmbH, “nanoNET Portable Protocol Stack,” Retrieved Dec. 12, 2006, from http://www.nanotron.com, 2005.

[3] W. C. Craig, “Zigbee: Wireless control that simply works,” Retrieved Jan. 24, 2006, from http://www.zigbee.org/en/resources/, ZigBee Alliance, Tech. Rep.

[4] E. Callaway, P. Gorday, L. Hester, J. Gutierrez, M. Naeve, B. Heile, and V. Bahl, “Home networking with IEEE 802.15.4: a developing standard for low-rate wireless personal area networks,” *IEEE Communications Magazine*, vol. 40, no. Iss.8, pp. 70–77, Aug 2002.

[5] *IEEE Std 802.15.4-2003*, IEEE Computer Society, 2003.

[6] P. Kinney, “Zigbee technology: Wireless control that simply works,” Retrieved Jan. 24, 2006, from http://www.zigbee.org/en/resources/, Tech. Rep., 2003.

[7] *ZigBee Specification 2004*, ZigBee Alliance, San Ramon, 2004.

[8] K. Wei, “Multicast over wireless mobile ad hoc networks,” Presentation. Retrieved Jan. 14, 2007, from http://www.nicetree.com/presentation/Presentation.pdf, 2006.

[9] C. Wu and Y. Tay, “AMRIS: A multicast protocol for ad hoc wireless networks,” in *Proceedings of IEEE MILCOM'99*, 1999.

[10] S. Lee, W. Su, and M. Gerla, “On-Demand Multicast Routing Protocol in Multihop Wireless Mobile Networks,” *Mobile Networks and Applications*, vol. 7, no. 6, pp. 441–453, 2002.

[11] S. K. Das, B. S. Manoj, and C. S. R. Murthy, “A dynamic core based multicast routing protocol for ad hoc wireless networks,” in *MobiHoc '02: Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing.* New York, NY, USA: ACM Press, 2002, pp. 24–35.

[12] J. Xie, R. Talpade, A. McAuley, and M. Liu, “AMRoute: Ad Hoc Multicast Routing Protocol,” *Mobile Networks and Applications*, vol. 7, no. 6, pp. 429–439, 2002.

[13] Y. Ko and N. Vaidya, “Geocasting in mobile ad hoc networks: Location-based Multicast algorithms,” *Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA'99. Second IEEE Workshop on*, pp. 101–110, 1999.

[14] *ZigBee Specification 2006*, ZigBee Alliance, San Ramon, 2006.

[15] T. Haenselmann, “An FDL'ed Textbook on Sensor Networks,” Retrieved Jan. 16, 2007, from http://www.informatik.uni-mannheim.de/~haensel/sn_book/, 2007.

[16] Culler, D. and Estrin, D. and Srivastava, M., “Overview of sensor networks,” *Computer(Long Beach, CA)*, vol. 37, no. 8, pp. 41–49, 2004.

[17] W. Granzer, W. Kastner, G. Neugschwandtner, and F. Praus, “Security in networked building automation systems,” in *Proc. 6th IEEE WFCS*, 2006, pp. 283–292.

[18] W. Kastner, G. Neugschwandtner, S. Soucek, and H. Newman, “Communication Systems for Building Automation and Control,” *Proceedings of the IEEE*, vol. 93, no. 6, pp. 1178–1203, 2005.

[19] *IEEE Standard 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications (ISO/IEC 8802-11: 1999)*, IEEE Computer Society, 1999.

[20] *Specification of the Bluetooth System, Core v2.0*, Retrieved Jan. 02, 2007, from http://www.bluetooth.org, Bluetooth SIG, 2004.

[21] T. Jorgensen and N. T. Johansen, “Z-wave as home control rf platform,” Retrieved September 15, 2006, from http://www.zen-sys.com/media.php?id=321, Zensys A/S, 2005.

[22] *Z-Wave System Design Specification: Z-Wave Protocol Overview*, Zensys A/S, Fremont, 2005.

[23] M. Knight, “How safe is z-wave?” *Institution of Engineering and Technology*, 2006.

[24] Nanotron Technologies GmbH, “nanoNET chirp based wireless networks,” Nanotron Doc. ID NA-04-0000-0298-1.03, Retrieved Dec. 12, 2006, from http://www.nanotron.com, 2005.

[25] ——, “nanoNET TRX transceiver (na1tr8), datasheet, version 2.08, Nanotron doc. id na-03-0111-0239-2.08,” Retrieved Dec. 12, 2006, from http://www.nanotron.com, 2006.

[26] N. Abramson, “The Throughput of Packet Broadcasting Channels,” *Communications, IEEE Transactions on [legacy, pre-1988]*, vol. 25, no. 1, pp. 117–128, 1977.

[27] K. Oikonomou and I. Stavrakakis, “A Probabilistic Topology Unaware TDMA Medium Access Control Policy for Ad-Hoc Environments,” *Personal Wireless Communications (PWC 2003), September*, pp. 23–25, 2003.

[28] *KNX Specification, Version 1.1*, Konnex Association, Diegem, 2004.

[29] Chipcon AS, “Chipcon CC2420 2.4 GHz IEEE 802.15.4 / ZigBee-ready RF Transceiver, Datasheet, Version 1.4,” Retrieved Feb. 10, 2007, from http://www.chipcon.com, 2006.

[30] Microchip Technology Inc., “Microchip MRF24J40 2.4 GHz IEEE 802.15.4 RF Transceiver, Datasheet, Version DS39776A,” Retrieved Feb. 10, 2007, from http://www.microchip.com, 2006.

[31] *Advanced Encryption Standard (AES)*, National Institute of Standards and Technology, 2001.

[32] C. Reinisch, W. Granzer, G. Neugschwandtner, F. Praus, and W. Kastner, “Wireless Communication in KNX/EIB,” in *Proceedings of KNX Scientific Conference*, 2006.

[33] C. Reinisch, W. Kastner, G. Neugschwandtner, and W. Granzer, “Wireless technologies in home and building automation,” in *Submitted to INDIN International Conference on Industrial Informatics*, 2007.

[34] W. Granzer, W. Kastner, G. Neugschwandtner, and F. Praus, “A modular architecture for building automation systems,” in *Proc. 6th IEEE WFCS*, 2006, pp. 99–102.

[35] G. Coulouris, J. Dollimore, and T. Kindberg, *Distributed systems: concepts and design.* Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 2000.

[36] A. Tanenbaum and M. Van Steen, *Distributed Systems: Principles and Paradigms.* Prentice Hall PTR Upper Saddle River, NJ, USA, 2001.

[37] (2007) Count-to-infinity problem. [Online]. Available: http://wiki.uni.lu/secan-lab/Count-To-Infinity+Problem.html

[38] G. Malkin, "RFC 2453: Routing Information Protocol (Version 2)," 1998.

[39] L. Kleinrock and F. Tobagi, "Packet switching in radio channels: Part 2-the hidden node problem in carrier sense multiple access modes and the busy tone solution," *IEEE Transactions on Communications*, vol. 23, no. 12, pp. 1417–1433, 1975.

[40] J. Moy, "RFC 2328: OSPF Version 2," 1994.

[41] ——, "RFC1585: MOSPF: Analysis and Experience," *Internet RFCs*, 1994.

[42] ——, "Multicast routing extensions for OSPF," *Communications of the ACM*, vol. 37, no. 8, pp. 61–66, 1994.

[43] R. Sedgewick, *Algorithms in C++ Part 5: Graph Algorithms.* Addison-Wesley, 2002.

[44] W. Fenner, "RFC2236: Internet Group Management Protocol, Version 2," *Internet RFCs*, 1997.

[45] S. Lee, W. Su, and M. Gerla, "On-Demand Multicast Routing Protocol (ODMRP) for Ad Hoc Networks," *IETF manet (draft-ietf-manet-odmrp-02. txt)*, 2000.

[46] Y. Dalal and R. Metcalfe, “Reverse path forwarding of broadcast packets,” *Communications of the ACM*, vol. 21, no. 12, pp. 1040–1048, 1978.

[47] S. Lee and M. Gerla, “Unicast performance analysis of the ODMRP in a mobile ad hocnetwork testbed,” *Computer Communications and Networks, 2000. Proceedings. Ninth International Conference on*, pp. 148–153, 2000.

[48] T. Ballardie, P. Francis, and J. Crowcroft, “Core based trees (CBT),” in *SIGCOMM ’93: Conference proceedings on Communications architectures, protocols and applications.* New York, NY, USA: ACM Press, 1993, pp. 85–95.

[49] P. Mohapatra, C. Gui, and J. Li, “Group Communications in Mobile Ad Hoc Networks,” *IEEE Computer Magazine*, 2004.

[50] S. Lee, W. Su, J. Hsu, M. Gerla, and R. Bagrodia, “A performance comparison study of ad hoc wireless multicast protocols,” in *Proceedings of INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies.*, vol. 2, pp. 565–574.

[51] P. Ji, Z. Ge, J. Kurose, and D. Towsley, “A comparison of hard-state and soft-state signaling protocols,” in *SIGCOMM ’03: Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications.* New York, NY, USA: ACM Press, 2003, pp. 251–262.

[52] B. Chen, K. Muniswamy-Reddy, and M. Welsh, “Ad-hoc multicast routing on resource-limited sensor nodes,” *Proceedings of the second international workshop on Multi-hop ad hoc networks: from theory to reality*, pp. 87–94, 2006.

[53] L. Zhang, D. Shen, X. Shan, V. Li, and Y. Ren, “A Receiver-Initiated Soft-State Probabilistic Multicasting Protocol in Wireless Ad Hoc Net-

works," *IEEE International Conference on Communications*, vol. 5, p. 3365, 2005.

[54] C. Perkins and P. Bhagwat, "Highly dynamic Destination-Sequenced Distance-Vector routing (DSDV) for mobile computers," *Proceedings of the conference on Communications architectures, protocols and applications*, pp. 234–244, 1994.

[55] C. Partridge and S. Pink, "An Implementation of the Revised Internet Stream Protocol (ST-2)," 1992.

[56] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc ondemand distance vector (AODV) routing," *IETF RFC3561–Experimental Standard, July*, 2003.

[57] S. Das, C. Perkins, and E. Royer, "Ad hoc on demand distance vector (AODV) routing," *Mobile Ad-hoc Network (MANET) Working Group, IETF, Jan*, vol. 81, 2002.

[58] E. Royer and C. Perkins, "Multicast Operation of the Ad hoc On-Demand Distance Vector Routing Protocol (MAODV)," *Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 207–218, 1999.

[59] ——, "Multicast Ad hoc On-Demand Distance Vector (MAODV) Routing," *draft-ietf-manet-maodv-00, July*, vol. 700, 2000.

[60] B. Grohmann, "Milliardenmaerkte durch drahtlose Kommunikation," *funkschau 22/2005*, 2005.

[61] A. Gupta and M. R. Tennefoss, "Radio frequency control networking: Why poor reliability today hampers what could be a viable technology in the future," *Echelon Doc. 005-0171-01B, Echelon Corp.*, 2005.