Chapter 1

Piconet interconnection strategies in IEEE 802.15.3 networks

Muhi A. I. Khair, Jelena Mišić, and Vojislav B. Mišić

Wireless Mesh Networks (WMN) commonly refer to distribute, cooperative communication networks formed by many nodes with wireless communication capability, some of which may be mobile [1, 2]. The main characteristic that distinguished WMNs from their predecessors, wireless ad hoc networks, is the cooperative communication capability, which is facilitated by the fact that each node may function as data source, data destination (consumer), or router, as appropriate. On account of this capability, WMN technology has many potential applications with a huge consumer demand. Performance of a WMN is determined by many factors, not the least important of which is the Medium Access Control (MAC) protocol. Already a number of MAC protocols have been proposed for use in WMNs, but the IEEE 802.15.3 standard for high data rate Wireless Personal Area Networks (HR-WPANs) [3] is often singled out as a viable candidate. The IEEE 802.15.3 standard offers a combination of CSMA-CA and TDMA at the MAC layer, as well as a set of several physical (PHY) layer modulation techniques that allow operation at data rates up to 55 Mbps. Recently, the 802.15.3 MAC has even been coupled with Ultra Wideband (UWB) PHY layer technology to offer even higher data rates at reduced collision probability. Similar to Bluetooth [17], 802.15.3 devices are organised in piconets controlled by a dedicated piconet coordinator (PNC). Unlike Bluetooth, however, devices in a 802.15.3 piconet can directly communicate with one another, which simplifies routing and improves throughput. Together, these features make the 802.15.3 standard a promising candidate for the implementation of wireless mesh networks.

In many cases, mesh networking requires coverage of larger physical areas in which distances can easily exceed the transmission range allowed by the power levels prescribed in the 802.15.3 standard. In such cases, the 802.15.3 technology can still be used, but the network must include two or more piconets interconnected through shared devices – bridges. Standard bridge configurations include the so-called Master-Slave bridge, where the coordinator of one piconet acts as the bridge during inactive periods, and Slave-Slave bridge, where the bridge node is an ordinary (i.e., non-coordinator) device in each of the piconets it visits. The main problems in multi-piconet networks are bridge and piconet scheduling, which are the focus of discussions in this Chapter; while issues related to topology formation and maintenance are also important, they are beyond the scope of this Chapter, and will not be covered.

We begin this Chapter with a brief overview of the MAC layer features and protocols as prescribed by the 802.15.3 standard. Then, we examine the problem of piconet interconnection, as well as bridge and piconet time scheduling, in the context of 802.15.3 networks. We present two basic strategies to interconnect the piconets to form a mesh network and discuss their pros and cons. Finally, we describe a simple piconet interconnection and scheduling protocol for IEEE 802.15.3 based mesh networks.

1.1 Basics of the IEEE 802.15.3 HR-WPAN standard

The IEEE 802.15.3 standard [3] for high data rate Wireless Personal Area Networks (HR-WPANs) is designed to fulfill the requirements of high data rate suitable for multimedia applications whilst ensuring low end-to-end delay. It is also designed to provide easy reconfigurability and high resilience to interference, since it uses the unlicensed Industrial, Scientific, and Medical (ISM) band at 2.4GHz which is shared with a number of other communication technologies such as WLAN (802.11b/g) and Bluetooth (802.15.1), among others.

Devices in 802.15.3 networks are organized in small networks called piconets, each of which is formed, controlled, and maintained by a single dedicated device referred to as the piconet coordinator (PNC). The network is formed in an ad hoc fashion: upon discovering a free channel, the PNC capable device starts the piconet by simply transmitting period beacon frames; other devices that detect those frames then request admission, or association (as it is referred to in the 802.15.3 standard). The coordinator duties include transmission of periodic beacon frames for synchronization, admission of new devices to the piconet, as well as allocation of dedicated time periods to allow unhindered packet transmission by the requesting device.

Time in an 802.15.3 piconet is structured in superframes delimited by successive beacon frame transmissions from the piconet coordinator. The structure of the superframe is shown in Figure 1.1. Each superframe contains three distinct parts: the beacon frame, the contention access period (CAP), and the channel time allocation period (CTAP). During the Contention Access Period, devices compete with each other for access; a form of CSMA-CA algorithm is used. This period is used to send requests for CTAs (defined below) and other administrative information, but also for smaller amounts of asynchronous data.

Channel Time Allocation Period contains a number of individual sub-periods (referred to as Channel Time Allocation, or CTA) which are allocated by the piconet coordinator upon explicit requests by the devices that have data to transmit. Requests for CTAs are sent during the Contention Access Period; as such, they are subject to collision with similar requests from other devices. The decision to grant the allocation request or not rests exclusively with the piconet coordinator, which must take into account the amount of resources available – most often, the traffic parameters of other devices in the network and the available time in the superframe. If a device is allocated a CTA, other devices may not use it, and contention-free access is guaranteed. CTA allocation is announced in the next beacon frame; it may be temporary or may last until explicit deallocation by the piconet coordinator. Typically, CTAs are used to send commands and larger quantities of isochronous and asynchronous data.

Special CTAs known as Management Channel Time Allocation (MCTA) are used for communication and dissemination of administrative information from the piconet coordinator to the devices, and vice versa. There are three types of MCTA defined in the standard - Association, Open, and Regular MCTA. Association and Open MCTAs use the Slotted Aloha [5] medium access technique, while Regular MCTAs use the TDMA mechanism.

Unlike other WPANs such as Bluetooth and 802.15.4, direct device-to-device communication is possible in an 802.15.3 piconet. In case the communicating devices are not within the transmission range of each other, the piconet coordinator (which, by default, must be able to communicate with both) may be involved as an intermediary, leading in effect to multi-hop intra-piconet communication. It is worth noting that problems of this nature may be alleviated by adjusting the transmission power, but also by making use of the adaptive data rate facility provided by the 802.15.3 standard. Namely, if transmission at the full data rate of 55 Mbps suffers from too may errors because the signal-to-noise-plus-interference ratio (SINR) is too low, different modulation schemes with lower data rate may be used to give additional resilience. This problem and its solutions, however, are beyond the scope of the present chapter.

Reliable data transfer in 802.15.3 networks is achieved by utilizing acknowledgements and retransmission of non-acknowledged packets. The standard defines three acknowledgment modes:

- no acknowledgement (No-ACK) is typically used for delay sensitive but loss tolerant traffic such as multimedia (typically transferred through UDP or some similar protocol);
- immediate acknowledgement (Imm-ACK) means that each packet is immediately acknowledged with a small packet sent back to the sender of the original packet; and
- delayed acknowledgement (Dly-ACK), where an acknowledgment packet is sent after successfully receiving a batch of successive data packets; obviously, this allows for

higher throughput due to reduced acknowledgment overhead – but the application requirements must tolerate the delay incurred in this case, and some means of selective retransmission must be employed to maintain efficiency.

1.2 Interconnecting IEEE 802.15.3 piconets

The 802.15.3 standard contains provisions for the coexistence of multiple piconets in the same (or partially overlapping) physical space. Since the data rate is high, up to 55 Mbps, the channel width is large and there are, in fact, only five channels available in the ISM band for use of 802.15.3 networks. If 802.11-compatible WLAN (or, perhaps, several of them) is/are present in the vicinity, the number of available channels is reduced to only three in order to prevent excessive interference between the networks adhering to two standards. As a result, the formation of multiple piconets must utilize time division multiplexing, rather than the frequency division one, as is the case with Bluetooth. Namely, a piconet can allocate a special CTA during which another piconet can operate. There are two types of such piconets: a child piconet and a neighbor piconet.

A child piconet is the one in which the piconet coordinator is a member of the parent piconet. It is formed when a PNC-capable device which is a member of the parent piconet sends a request to the parent piconet coordinator, asking for a special CTA known as a private CTA. Regular CTA requests include the device addresses of both the sender and the receiver; a request for a private CTA is distinguished by virtue of containing the same device address as both the sending and the receiving node. When the parent piconet coordinator allocates the required CTA, the child piconet coordinator may begin sending beacon frames of its own within that CTA, and thus may form another piconet which operates on the same channel as the parent piconet, but is independent from it. The private CTA is, effectively, the active portion of the superframe of the child piconet. The child superframe consists, then, of this private CTA which can be used for communication between child piconet coordinator (PNC) and its devices (DEVs); the remainder of the parent superframe is reserved time – it can't be used for communication in the child piconet.

Master-Slave bridge. From the standpoint of piconet interconnection, the child piconet mechanism allows for simple implementation of the Master-Slave interconnection topology, since the two piconets are linked through the child piconet coordinator which partakes in both of them, and thus can act as the bridge. Figure 1.2 schematically shows such topology in which Piconet 2 is the child of Piconet 1; the child piconet PNC is also acting as a Master-Slave bridge that links the piconets. The timing relationship of superframes in parent and child piconets is shown in Figure 1.3, where the top part corresponds to the parent piconet and the bottom part to the child piconet. Note that the distinction is logical rather than physical, since the piconets share the same RF channel.

A given piconet can have multiple child piconets, and a child piconet may have another child of its own. Obviously, the available channel time is shared between those piconets, at the expense of decreased throughput and increased delay; but the effective transmission range may be increased.

The Slave-Slave bridge. This interconnection topology may also be implemented, but in a slightly more complex manner. Namely, direct communication between the members of different piconets is not possible; the only shared device is the PNC of the child piconet. If an ordinary device wants to act as a bridge, it must explicitly associate with both parent and child piconets, and obtain a distinct device address in each of them. In this manner, multiple bridges may exist between the two piconets. The topology of two piconets interconnected through a Slave-Slave bridge is shown in Figure 1.4. Note that, in this case, the piconet may be linked through a parent-child relationship; but they could also use different RF channels, with a certain penalty because of the need for the bridge to synchronize with two independently running superframe structures.

Challenges. As can be seen from the discussion above, the main challenge in forming a multi-piconet network that uses the same RF channel – i.e., a complex network in which all piconets are related through parent-child relationships – is to develop a network-wide distributed scheduling algorithm that will allocate channel time to all devices in an efficient and fair manner. Since time division multiplexing among each parent-child piconet pair is used, we need not worry about the conflicts – i.e., collisions – between transmissions originating from different piconets in the networks: the transmissions during allocated CTAs are guaranteed to be conflict-free. The need to wait until the appropriate active portion of the superframe incurs some additional delays besides the usual transmission delay and access delay in the outbound queue of the source device; furthermore, the bridge device operates its own queues (one for each direction of the traffic) and these can also add delay to the total packet transmission time. As those queues are necessarily implemented with buffers of finite size, there exists non-zero probability that the buffer will overflow, in which case packets may be blocked from entering the queue; if reliable transfer is needed, the possibility of packet blocking necessitates the use of Imm-ACK or Dly-ACK acknowledgment policy. In addition, we must devise an efficient and fair algorithm to partition the available channel time between the piconets, taking into account the traffic intensity both within the piconets and between them.

Using different RF channels. Mesh networks can also be created using a different scenario, in which several multi-piconet networks operate in the same physical space but on different RF channels. While physical conflicts between transmissions originating from different multi-piconet networks are still absent by virtue of frequency division multiplexing, scheduling conflicts between the piconets will be the main source of complexity, as the device that wants to act as a bridge must alternatively synchronize with piconets that operate according to entirely unrelated schedules. This precludes the use of Master-Slave bridges to interconnect such piconets. Namely, the Master-Slave bridges must not abstain from their duties as the PNCs in their respective piconets for prolonged periods of time. As a result, piconets operating on different RF channels favor interconnection through Slave-Slave bridges, i.e., devices that act as ordinary nodes in each of the piconets they belong to. As such devices have no coordinator duties, their absence from a given piconet will not cause any problems there. In fact, their absence might even go unnoticed if there happens to be no traffic directed to such devices during that time interval.

Neighbor piconets. The 802.15.3 standard also provides the concept of the neighbor piconet, which is intended to enable an 802.15.3 piconet to coexist with another network that may or may not use the 802.15.3 communication protocols; for example, an 802.11 WLAN in which one of the devices is 802.15.3-capable. A PNC-capable device that wants to form a neighbor piconet will first associate with the parent piconet, but not as an ordinary piconet member; the parent piconet coordinator may reject the association request if it does not support neighbor piconets. If the request is granted, the device then requests a private CTA from the coordinator of the parent piconet. once a private CTA is allocated, the neighbor piconet can begin to operate. The neighbor piconet coordinator may exchange commands with the parent piconet coordinator, but no data exchange is allowed. In other words, the neighbor piconet is simply a means to share the channel time between the two networks. Since, unlike the child piconet, data communications between the two piconets are not possible, this mechanism is unsuitable for the creation of multi-piconet networks, and, consequently, for mesh networking.

1.3 Implementing Mesh Networks with 802.15.3

In this Section we will first explain the interconnection (bridging) mechanism, followed by our proposed scheduling algorithm for channel time allocation in the mesh network. The superframe structure of our mesh MAC protocol follows the IEEE 802.15.3 MAC superframe and the channel time allocation is based on TDMA, during the guaranteed access period, and CSMA/CA, during the contention period.

Two common approaches, namely the Master-Slave bridge and the Slave-Slave bridge are used for piconet interconnection in different networks. In the case of a Master-Slave bridge, Figure 1.2, the bridge device is the PNC for Piconet 2 and a normal member of Piconet 1. In the case of a Slave-Slave bridge, Figure 1.4, the bridge device is an ordinary member (DEV) in both piconets. We can combine both types of bridges in the mesh environment in order to cover larger areas. The choice of the type of interconnection depends on location of the bridge device within the mesh network. The interconnection will be established through a Master-Slave bridge if a PNC-capable device is located in such a way that it can easily control one piconet and participate in the other one. On the other hand, the Slave-Slave bridge can be used if no suitable PNC-capable device can be found, or if the two piconets operate on different RF channels, possibly because the traffic volume is too high to be serviced with half the available bandwidth.

Operation of the Master-Slave bridge. The bridge establishes a connection between a parent and a child piconet where the bridge device acts as the PNC of the child piconet. The bridge device maintains two queues to temporarily store, and subsequently deliver, the traffic in both directions. As can be seen from Figure 1.3, the superframe duration is the same for both parent and child piconets; in fact, the child superframe is simply a private CTA from the parent superframe. The only setup operation needed in this case is for the child piconet PNC to request a private CTA as explained above. Once such a CTA is allocated by the parent piconet PNC, the child piconet PNC simply begins to send beacons at the beginning of the CTA, which is also the beginning of its own superframe. Devices that need to send data to the other piconet can simply request their own CTAs from their respective PNCs.

Operation of the Slave-Slave bridge. A device that is already associated with a piconet can detect the presence of a new piconet by receiving a beacon sent by its PNC, or a control packet with a piconet identification number (PNCID) that is different from the existing one. Whenever a prospective bridge device detects the presence of two piconets within its transmission range, it initiates the connection establishment algorithm (Algorithm 1). First, the device waits for the MCTA period or CAP period to send a request command for bridging. Then it will use the four-way handshake (RTS-CTS-DATA-ACK) to send the request command, piggybacking its current scheduling information to the neighbour PNC. The neighbour PNC adjusts its scheduling information based on the received scheduling information from its neighbour piconet. If the PNC is a Master-Slave bridge in its own right, it will request a private CTA from its parent PNC, trying to accommodate the demands of the bridge device. The bridge requirements are, simply, that the neighbouring child piconets obtain channel time for transmission (i.e., private CTAs) without interfering with each other. A positive response from the parent PNC establishes the connection between the child piconets. After the connection establishment, the bridge device needs to maintain a table that keeps track of the scheduled times of activity in each piconet. The PNCID uniquely identifies each record in the table and helps the bridge device switch in a timely fashion between different piconets.

Algorithm 1: The Slave-Slave bridge connection.

scan presence of overlapped coverage ;
if scan == positive **then**
  send join-request to neighbour PNC using four way handshaking protocol ;
feedback from neighbour PNC ;
update scheduling table with PNCID and received scheduling information ;

Channel Scheduling. The channel time scheduling of the network under consideration will be based on average queue size of the devices. The queue size of an ordinary device (i.e., not a PNC or a bridge device) primarily depends on packet arrival rate of that device. The queue size for a bridge device is based on the incoming packet queue sizes from neighbour piconets and outgoing packet queue sizes to the neighbour piconets. The bridge device will use the average of these two queues size to determine its channel time requirement. The devices send requests for channel time based on the average queue size to their respective PNCs. The PNC uses Algorithm 2 based on the request from the bridge in question. In case of a request from a bridge device, the PNC schedules channel time and a private CTA (for the child piconet) in such a way that there will be no overlap of channel time between the two adjacent piconets. A representative topology that employs both types of bridge interconnection is shown in Figure 1.5. In this network, the parent piconets P1 and P2 are located beyond each other's transmission ranges and, thus, can operate on the same RF channel. However, the presence of two child piconets that can hear each other – they are, in fact, interconnected – presents a challenge for scheduling. In order to resolve this, the two parent piconets P1 and P2 will assign channel time for their children in different time slots, based on the scheduling information they exchanged during connection establishment. Let us consider time slots in the superframe in Figure 1.5. The time slots represented by P1/P2 (or P2/P1) imply that P1 and P2 can communicate at the same time. On the other hand, when a child piconet is operating, no other piconet in its range can talk. In this case we can assume that a single superframe (actually two superframes from two different parent piconets) are divided into four time slots. Within each time slot, the devices will have guaranteed channel time and contention period. There are also MCTAs in each time slot during which a new node can join or a bridge can establish a connection. There is a chance of conflict during the MCTA period as the new devices do not have any knowledge of the current scheduling information resulting in the hidden terminal problem. We will use the four way handshaking protocol to resolve this problem.

Algorithm 2: Scheduling of channel time.

```
if request command from Master-Slave bridge then
    assign private CTA anywhere in the superframe;
if request command from Slave-Slave bridge then
    check piggyback data for neighbour scheduling information ;
    if no scheduling information then
        request for scheduling information ;
    scheduling information received ;
    calculate required channel time based on average queue size ;
    determine private CTA position ;
    assign private CTA and channel time ;
```

1.4 Related Work

The MAC protocols for wireless mesh networks are different from the traditional wireless MACs in terms of self organization, distributed nature, multi-hop, and mobility. The WMNs can be designed with a single channel or multiple channels. For simplicity, we will focus on a single channel wireless mesh network in a parent-child interconnection.

Wireless mesh networks have often been developed using the IEEE 802.11 DCF MAC protocol, and most of the research work in mesh networks has explored or modified the features of 802.11 MAC protocol to improve network performance. Bicket et al. [7] have evaluated the performance of 802.11b mesh networks; their experiments have shown that an ad hoc mesh network implemented using 802.11b technology can achieve sustained throughput of around 630 Kbps, significantly below the supported data rate of 11 Mbps. By the same token, Yamada et al. [8] have identified two problems of 802.11b based mesh networks: limited throughput and degradation of fairness. To solve these problems they have introduced two new control packets, namely Invite-to-send (ITS) and Copied CTS (CCTS). The use of

ITS and CCTS leads to improvements in throughput, but at the cost of increased control overhead and delay. Also, the overhead due to ITS and CCTS packets and end-to-end packet delay will increase with the network load. However, in an 802.15.3 network, data communications are accomplished using dedicated time periods, hence there is no need to introduce additional control packets such as ITS and CCTS.

MACA was developed to solve the hidden and exposed terminal problems of traditional CSMA [9] protocols. In MACA, the sender and receiver exchange RTS and CTS control packets before sending a data packet to avoid collisions. Fullmer and Garcia-Luna-Aceves [10] describe the scenario where MACA fails to avoid collisions due to hidden terminals. MACA may also make a device wait for a long period to access the medium because its use of the BEB [1] algorithm [6]. To overcome the problems of MACA, a new solution was proposed by Bharghavan et al. called Media Access Protocol for Wireless LANs (MACAW) [11]. Basically MACAW is a modification of the BEB algorithm in MACA. It introduces acknowledgement and data-sending (DS) control packets producing the RTS-CTS-DS-DATA-ACK sequence for data transfer. The IEEE 802.11 standard [12] has been developed by adopting the CSMA and MACAW with further modifications to support wireless LANs. Both the IEEE 802.11 MAC and MACAW do not support real time data transfer because of the absence of guaranteed time slots. Therefore, Lin and Gerla [13] proposed an enhanced version of MACA called MACA with Piggybacked Reservation (MACA/PR) to support real-time traffic.

The MACA/PR protocol is a contention based protocol with a reservation mechanism. It has been designed to support multimedia data in multihop mobile wireless network providing guaranteed bandwidth through reservation. Every node keeps the reservation information of sending and receiving windows of all the neighbour nodes in a table, which is refreshed after every successful RTS-CTS-DATA-ACK (known as four way handshaking protocol) cycle. The RTS and CTS packets are exchanged for the first packet in the transfer of a series of real-time data packets. The reservation information for the next data packet is piggybacked with the prior data packet and the receiver confirms this reservation in the acknowledgement control packet. The limitation of MACA/PR is that it requires help from the network layer routing protocol. However, MACA/PR has better performance in terms of latency, packet

[1] In Binary Exponential Backoff (BEB) a device doubles the size of its backoff window if a collision is detected.

loss, mobility, and bandwidth share than both asynchronous packet radio network (PRNET [2]) and synchronous TDMA based MACs. The use of fixed reserved time slots in MACA/PR can result in wastage of bandwidth. Manoj and Murthy [14] have proposed a modification to the reservation mechanism of MACA/PR to prevent bandwidth wastage. In the modified scheme, the reserved slots can be placed at any position in the superframe and unused resources (channel time) are released after a successful transmission.

We note that the 802.15.3 MAC uses TDMA based channel allocation to provide guaranteed time slots for data transfer. However, the piggybacked reservation information of MACA/PR can be employed together with the TDMA based MAC to support real-time data transfer along with best-effort traffic in 802.15.3 based wireless mesh networks.

Xiao [15] has performed a detailed performance evaluation of the IEEE 802.15.3 [3] and IEEE 802.15.3a [16] standards through simulation and mathematical analysis. He has also done a throughput analysis of the 802.11 [12] protocol, which uses backoff with counter freezing during inactive portions of the superframe. The freezing and backoff techniques are essentially the same in the 802.11 and 802.15.3 MACs, except that different ways of calculating the backoff time are utilized. The backoff and freezing have an impact on the performance of the network; especially the backoff has a direct impact on the delay parameter. Large backoff windows can result in longer delays. On the other hand, small backoff windows may increase the probability of collisions. Xiao used the backoff procedures defined in the 802.11 and 802.15.3 MAC specifications; this work gives us performance of the protocol in terms of throughput over various payload sizes, but the performance of reliable transmission in error-prone wireless network during contention period needs more study.

1.5 Experimental results

We have built a discrete event simulator of a two-piconet 802.15.3 network in a parent-child topology, using the Petri-net based object oriented simulation engine Artifex [18]. Different MAC parameters in our simulation have default values defined in the IEEE 802.15.3 standard,

[2] In a PRNET, the devices use the same channel and share it dynamically.

except where explicitly noted. In this master-slave architecture, the bridge device acts as the PNC for the child piconet, while at the same time acting as a normal member (DEV) of the parent piconet. The parent piconet has the PNC and nine devices, while the child piconet has the bridge (acting as PNC) and five devices. The bridge and the PNC of the parent piconet do not generate any packets: they simply receive and forward packets to their proper destinations. Ordinary devices generate packets according to a Poisson distribution for the overall data rate of 11 Mbps; another parameter, the locality probability P_l, determines the proportion of the packets that are sent to the destinations in the other piconet.

The bridge maintains two queues, one for the uplink traffic (i.e., packets going from child piconet to the parent piconet) and the other for the downlink traffic (i.e., packets sent from the parent piconet to the child piconet). Each queue has a buffer of size 100 packets.

The beacon period is $100\mu s$, parent piconet CAP time is $3600\mu s$, and the child piconet CAP is $2000\mu s$ long. The size of each CTA is $924\mu s$, which suffices for sending a single packet of the chosen packet size. Each of the ordinary devices gets four CTAs in each superframe, while the number of CTAs allocated to the bridge was made variable. Figure 1.6 shows the structure of the superframe.

Selected performance results for the downlink and uplink traffic are shown in Figures 1.7 and 1.8, respectively. In all the diagrams, independent variables were the locality probability and the number of CTAs allocated to the bridge for the appropriate traffic direction. For downlink traffic, Figure 1.7, the number of CTAs for uplink traffic was kept at 3. For uplink traffic, Figure 1.8, the number of CTAs for downlink traffic was kept at 4 (since the parent piconet has more devices than the child one).

As can be seen, the performance is critically dependent on the locality probability and the number of CTAs allocated to the traffic. In fact, the downlink (parent-to-child) traffic shows satisfactory performance only when the number of CTAs is above 5 and when less than 20% of the traffic is sent to the child piconet. If these conditions are not met, bridge buffer operates at high utilization ratio, above 80%, which leads to high blocking probability. The need to retransmit lost packets produces additional traffic and increases end-to-end packet delays. Similar observations hold for the uplink (child-to-parent) traffic, except that the region of 'normal' traffic is slightly wider, due to the lower number of devices in the child piconet.

These results show that wireless mesh networks can efficiently be implemented using 802.15.3 high data rate WPAN technology; however, careful network design and the development of efficient scheduling algorithms are needed to achieve the full potential of this technology.

1.6 Conclusion

In this Chapter, we have proposed interconnection schemes between 802.15.3 piconets and a network-wide scheduling policy to allocate channel time to the devices. We have discussed how the interconnection of piconets in mesh environment can affect scheduling when the bridge has overlapped coverage area with the same channel. Our proposed scheduling algorithm is simple and we expect that it will provide conflict free communication and give good throughput and delay performance. Furthermore, proposed solution will help in the development of complex heterogeneous mesh network that can support mobility and dynamic topology change.

References

[1] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey", Computer Networks, vol. 47, pp. 445-487, March 2005.

[2] Wimedia Alliance: http://www.wimedia.org, 2006.

[3] "Standard for part 15.3: Wireless medium access control (MAC) and physical layer (PHY) specifications for high rate wireless personal area networks (WPAN)", IEEE, New York, NY, IEEE standard 802.15.3, 2003.

[4] M.-G. Di Benedetto, T. Kaiser, A. F. Molisch, I. Oppermann, C. Politano, and D. Porcino, "UWB Communication Systems A Comprehensive Overview", New York, NY: Hindawi Publishing Corporation, 2006.

[5] J. F. Kurose and K. W. Ross, Computer Networking: A Top-Down Approach Featuring The Internet. Boston, MA: Addison-Wesley Longman, 2000.

[6] C. Siva Ram Murthy and B. Manoj, Ad Hoc Wireless Networks, Architecture and Protocols. Upper Saddle River, NJ: Prentice Hall, 2004.

[7] J. Bicket, D. Aguayo, S. Biswas, and R. Morris, "Architecture and evaluation of an unplanned 802.11b mesh network", 11th ACM International Conference on Mobile Computing and Networking, vol. 1, Cologne, Germany, September 2005, pp. 31–42.

[8] A. Yamada, A. Fujiwara, and Y. Matsumoto, "Enhancement of mesh network oriented IEEE 802.11MAC protocol", 10th Asia-Pacific Conference on Communications and 5th International Symposium on Multi-Dimensional Mobile Communications, vol. 1, Kanagawa, Japan, September 2004, pp. 142–146.

[9] W. Stallings, Data and Computer Communications. Upper Saddle River, NJ: Prentice Hall, 2003.

[10] C. L. Fullmer and J. J. Garcia-Luna-Aceves, "Floor acquisition multiple access (FAMA) for packet-radio networks", ACM SIGCOMM '95, Cambridge, MA, September 1995, pp. 262–273.

[11] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: a media access protocol for wireless LAN's", ACM SIGCOMM '94, London, UK, August 1994, pp. 212–225.

[12] "Standard for wireless LAN medium access control (MAC) and physical layer (PHY) specifications", IEEE, New York, NY, IEEE standard 802.11, May 1997.

[13] C. R. Lin and M. Gerla, "MACA/PR: an asynchronous multimedia multihop wireless network", IEEE INFOCOM '97, Kobe, Japan, March 1997, pp. 118–125.

[14] B. S. Manoj and C. S. Ram Murthy, "Real-time support for ad hoc wireless networks", IEEE ICON '02, Singapore, August 2002, pp. 335–340.

[15] Y. Xiao, "MAC layer issues and throughput analysis for the IEEE 802.15.3a UWB", Dynamics of Continuous, Discrete and Impulsive Systems, Series B: Applications and Algorithms, vol. 12, pp. 443–462, June 2005.

[16] IEEE, IEEE 802.15 WPAN high rate alternative PHY Task Group 3a (TG3a). Available online at http://www.ieee802.org/15/pub/TG3a.html, August 2006.

[17] J. Mišić and V. B. Mišić, Performance Modeling and Analysis of Bluetooth Networks; Polling, Scheduling, and Traffic Control, Boca Raton, FL: Auerbach, 2005.

[18] RSoft Design, Inc., Artifex v.4.4.2, San Jose, CA, 2003.

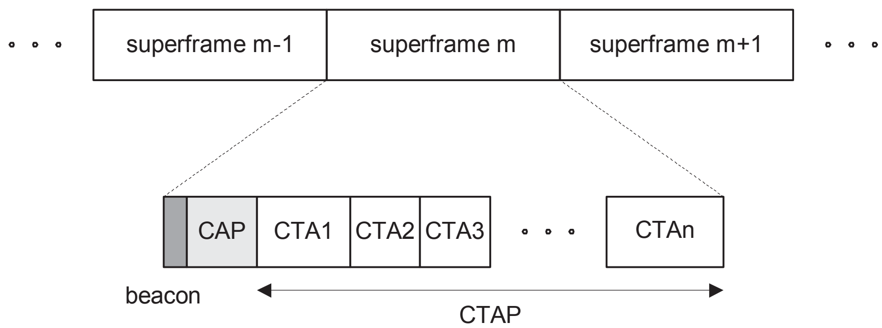


Figure 1.1: Superframe format in an IEEE 802.15.3 piconet (adapted from [3]).

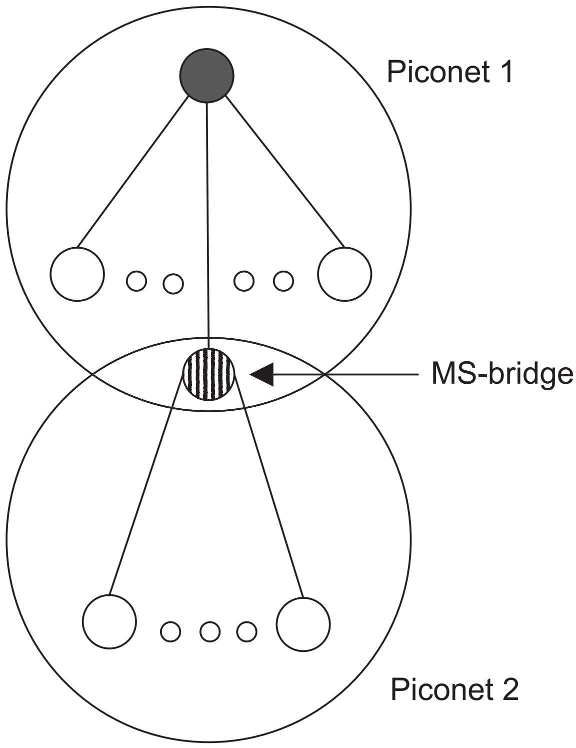


Figure 1.2: Piconet interconnection through a Master-Slave bridge.

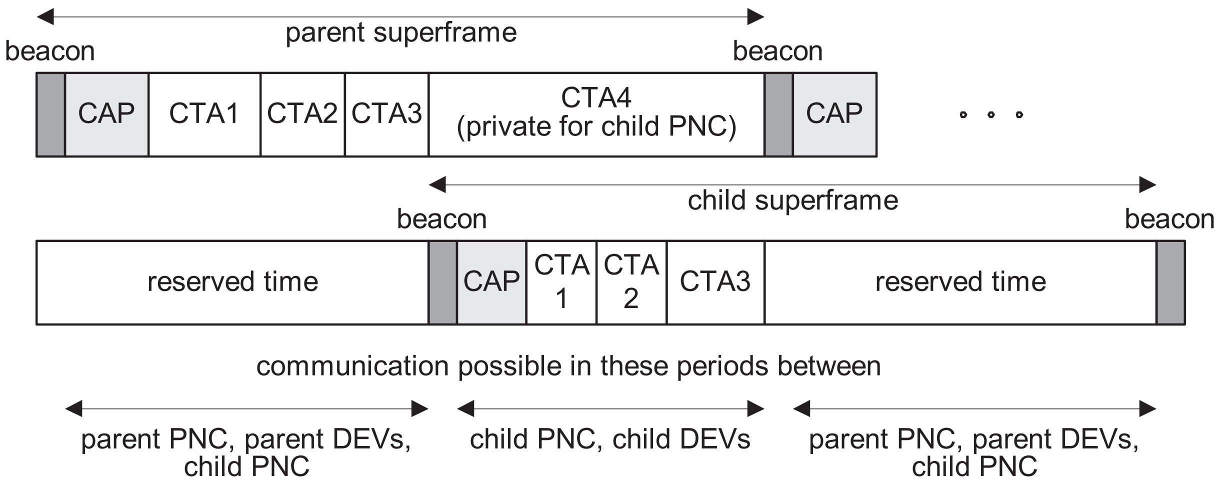


Figure 1.3: Superframe structure for parent and child piconets.

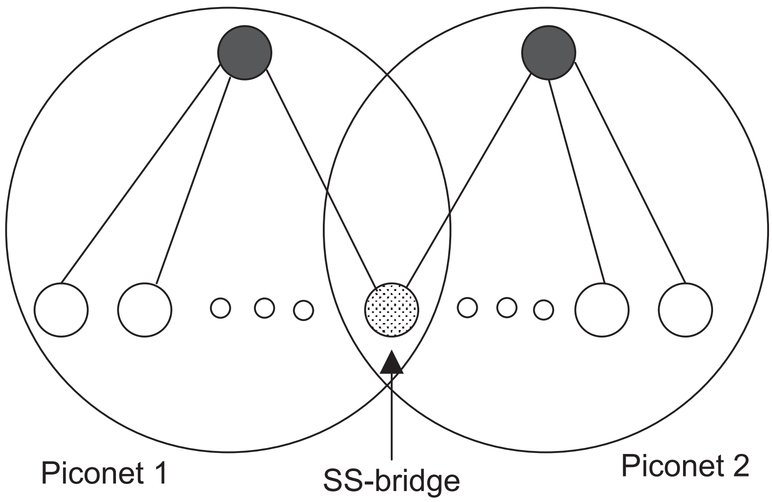


Figure 1.4: Piconet interconnection through a Slave-Slave bridge.

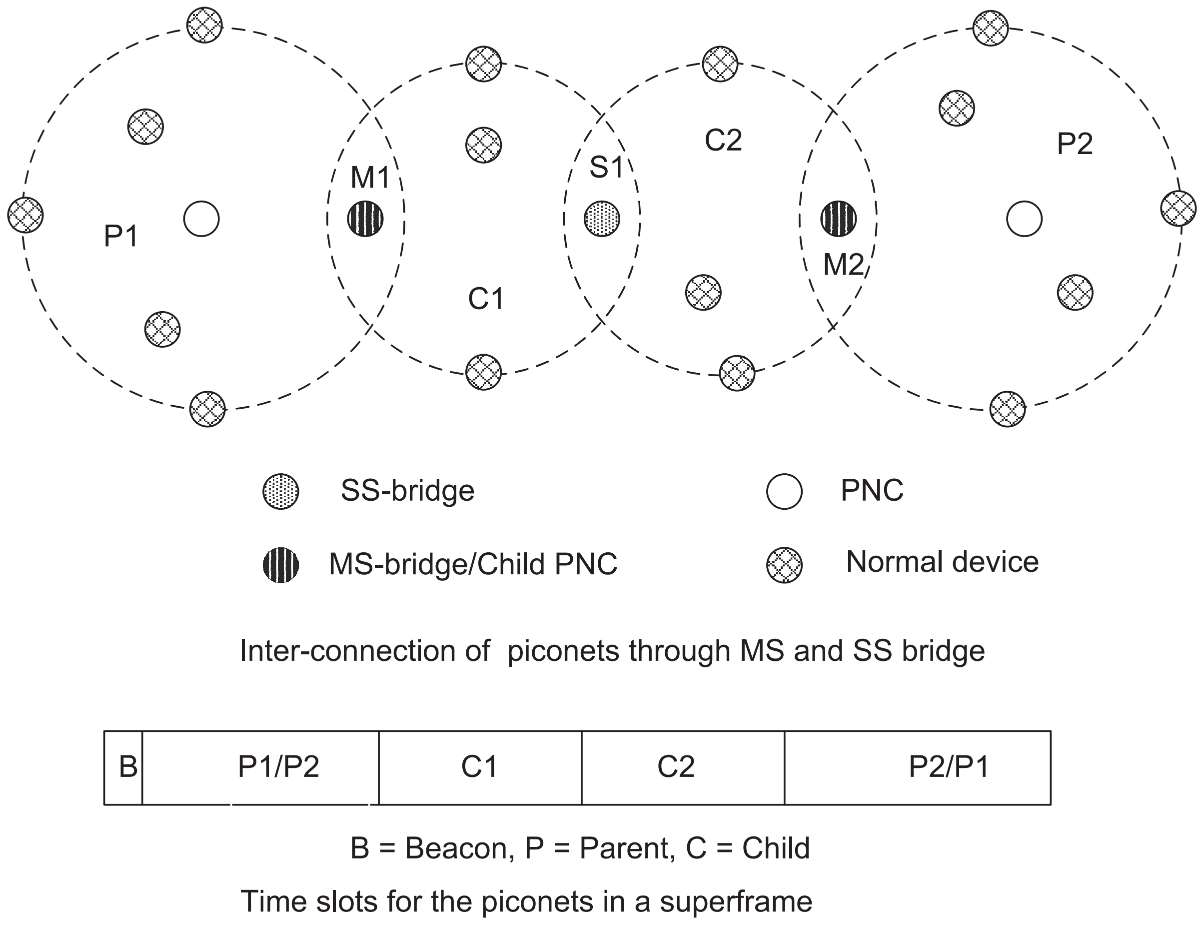


Figure 1.5: Multiple piconet interconnection with overlapped coverage and time slot management in the superframe.

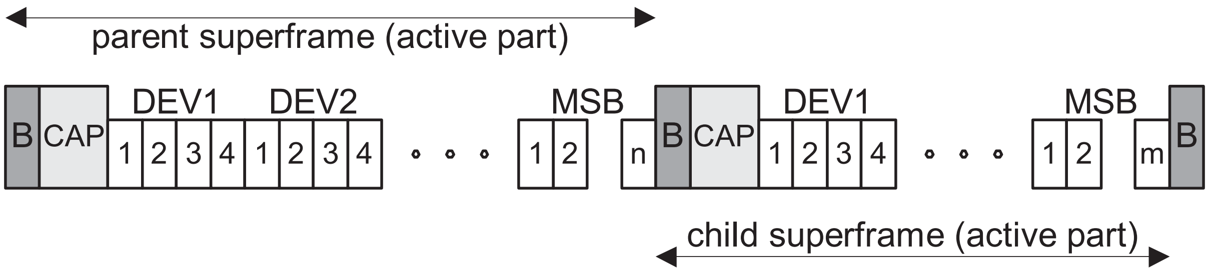
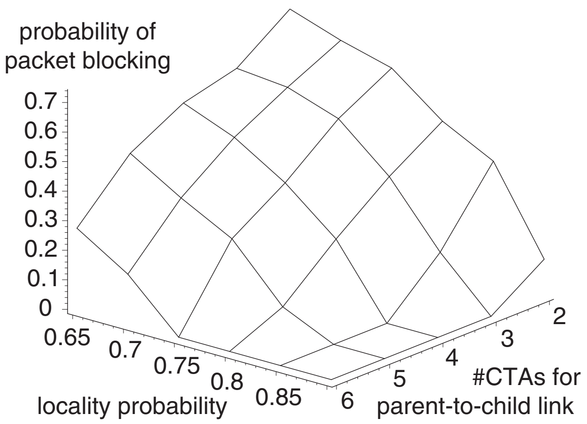
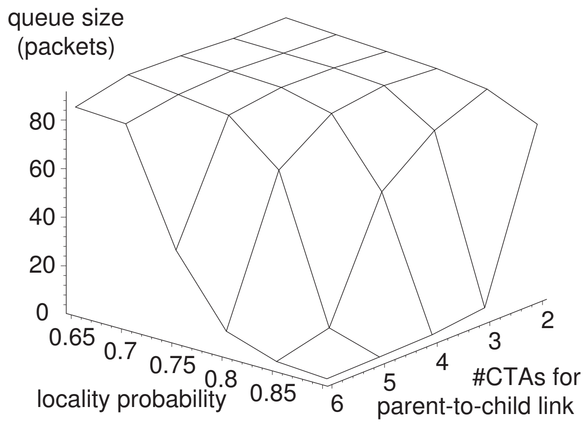


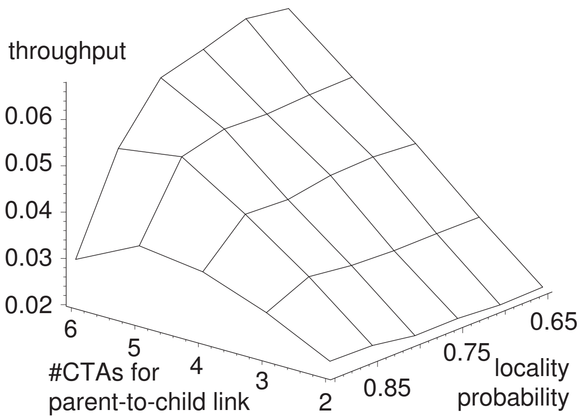
Figure 1.6: Superframe structure in the experimental setup.



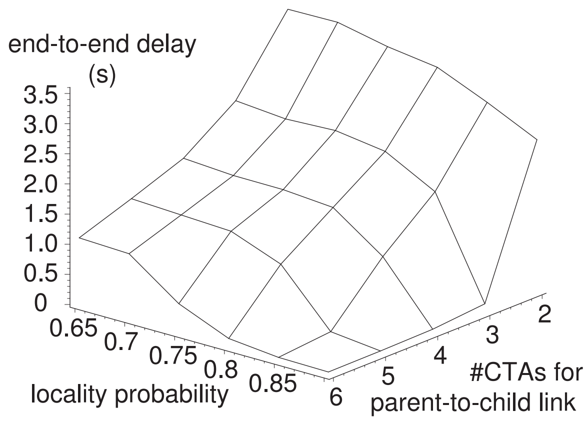
(a) Packet blocking probability at the bridge.



(b) Mean queue size at the bridge.

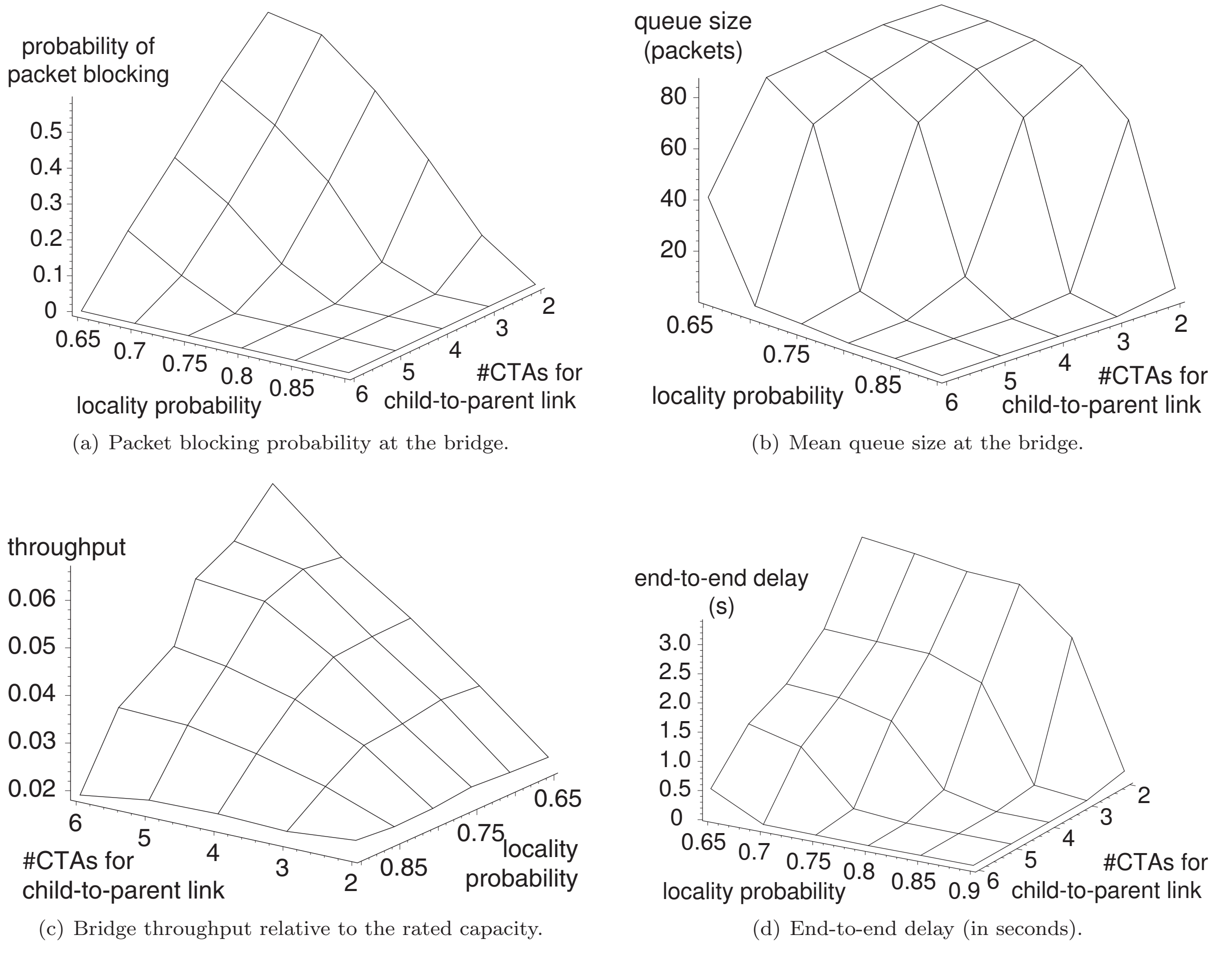


(c) Bridge throughput relative to the rated capacity.



(d) End-to-end delay (in seconds).

Figure 1.7: Performance of traffic sent from the parent piconet to the child piconet.



(a) Packet blocking probability at the bridge.

(b) Mean queue size at the bridge.

(c) Bridge throughput relative to the rated capacity.

(d) End-to-end delay (in seconds).

Figure 1.8: Performance of traffic sent from the child piconet to the parent piconet.