Configuring the IEEE 802.15.4 MAC Layer for Single-sink Wireless Sensor Network Applications

Joe Hoffert, Kevin Klues, Obi Orjih
Washington University in St. Louis
{jwh1,kak1,ocol}@cec.wustl.edu

Abstract

With the advent of low-rate wireless personal area networks (LR-WPANs) have come the medium access control (MAC) and physical (PHY) layer protocols that manage the communication of the multiple nodes in a network. The 802.15.4 MAC and PHY protocol has gained acceptance in industry and is increasing in popularity. 802.15.4 is highly configurable and as such, the user can be overwhelmed by all the possible settings. Additionally, configurations generated by novice users can be poorly formed or error prone. This paper provides a brief introduction to the 802.15.4 standard and investigates its use in wireless sensor networks (WSNs). A subset of the 802.15.4 configurations are examined to determine how to improve performance in the areas of reliability, throughput, and power management, for some common network topologies. Simulations have been run using ns2 to test some these configurations, and the results obtained have been analysed, yielding recommendations on how best to configure the standard and how to improve upon it.

1. Introduction

While the IEEE 802.15.4 wireless networking standard is quickly gaining popularity in industry as the physical (PHY) and media access control (MAC) layer of choice for developing LR-WPAN applications, the academic community has tended to neglect the impact this standard is having in the field of wireless networking.

In contrast to other MAC protocols that have been developed for wireless sensor networks (WSNs), 802.15.4 is highly configurable. This comes at the cost of increased code size, which is an issue for resource-constrained WSN platforms. For this reason, implementing the entire standard on a WSN platform has proved to be a challenge, and as of this writing we are not aware of any such open-source implementations. The primary benefit of having such a highly configurable MAC is that it allows applications to more easily adapt to changes in network workload on the fly. Since the standard is still in its relative infancy, however, it is not widely understood how it should be configured to optimize performance in aspects that might be important WSNs.

Three such areas of importance include: 1) reliability, 2) throughput, and 3) power management.

802.15.4 supports acknowledgments, which may be turned on and off in certain data transfer modes. If acknowledgments are turned on then transmissions should always be reliable. What is more interesting, however, is what can be termed as ”efficient reliability”. Efficient reliability is the idea of supporting reliability with the fewest number of retransmissions. In an application where data is to be gathered reliably from a WSN in close to real time, constraints must be placed on the latency of the data. While acknowledging transmissions may ensure that all the data is eventually received, it would be hard to guarantee the freshness of the data unless the number of retransmissions is minimized. One of our aims is to determine how best to configure the MAC if efficient reliability is desired in a WSN.

The idea of maximizing throughput may seem contradictory to the concept of LR-WPANs with the emphasis being on low rate. However, it seems plausible that there will be times for certain applications when throughput will want to be maximized for a limited amount of time. For instance, an intrusion detection system might operate at a low duty cycle for most of the time, but change operating modes when an intruder is detected. Once this event occurs, it becomes critical for the network to track the intruder and provide as much data as possible to the base station. In such a case, the network should reconfigure from an energy-saving mode to a maximum-throughput mode.

The standard supports power management by providing means to specify the duty cycle of nodes. When employing a sleep schedule, it is important to know what sort of performance to expect from the network. We explore how one should set the MAC duty cycle in order to minimize power consumption while meeting other performance constraints, such as reliability of data delivery.

While attempting to optimally configure 802.15.4, it should be noted that the standard is not perfect. There are cases in which its definition hinders achieving ideal performance. We comment on how the standard could be improved and/or how application developers can implement mechanisms on top of the standard to improve performance.

Our goal is to give some preliminary guidance on how to configure the MAC and extend its services in order to optimize performance based on the criteria above. To arrive at this, we first introduce the 802.15.4 PHY and MAC specifications, and the Zigbee routing layer being developed on top of it. Next, we present some examples of prior work related to the evaluation of the standard. We then describe the topologies we have chosen to represent application scenarios and discuss our experiments based on these scenarios and the results they yielded. Finally, we make our recommendations, suggest improvements, mention future work direction, and conclude with a summary.

2. Introduction to 802.15.4

The IEEE 802.15.4 standard[4] was created for low-rate, wireless personal area networks. Other existing standards for wireless communication are optimized for throughput, and are often not concerned with power consumption. Devices in these networks are either mains powered or their batteries are easily recharged. 802.15.4 is targeted for low cost, resource constrained devices that are deployed for lengthy periods of time without such maintenance as battery replacement. The application domain for such a standard includes wireless sensor networks, industrial and commercial control and monitoring, and home automation. These devices would typically act as stick-on sensors, virtual wires, wireless hubs or cable replacements.

The standard is divided into two layers, the physical layer (PHY) and the media access control (MAC) layer. These layers sit below the routing, e.g. Zigbee, or application layers (as shown in figure 1). The PHY and MAC layers provide building blocks for creating different network topologies, including star, mesh, and cluster tree networks. It is designed to operate on two classes of devices: reduced function devices (RFDs) and fully functional devices (FFDs). FFDs have the capability to communicate with any device in a network within range of them, while RFDs are only able to directly communicate with FFDs. Every network consists of multiple FFDs and RFDs, with one of the FFDs designated as the personal area network (PAN) coordinator.

In the following subsections, we give a brief overview of the 802.15.4 PHY and MAC layers, followed by an introduction into the configurability of these two layers.

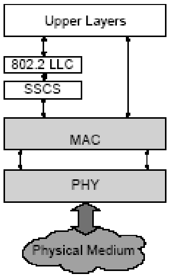


Figure 1. The 802.15.4 Protocol Stack ([4])

2.1. 802.15.4 PHY

The PHY layer specification dictates how 802.15.4 devices may communicate with each other over the wireless channel. It allows for the use of three frequency bands with varying data rates. The bit rates are 20 kb/s in the European 868 MHz band (868-868.6 MHz), 40 kb/s in the North American 915 MHz band (902-928 MHz), and 250 kb/s in the worldwide 2.45 GHz band (2.4-2.4835 GHz). This layer is responsible for activation and deactivation of the transceiver, channel frequency selection, and data transmission/reception. In addition, it performs channel energy detection (ED), link quality indication (LQI) for received packets, and clear channel assessment (CCA) for the MAC's carrier sense multiple access with collision avoidance (CSMA-CA) protocol. In addition to the packet length information and the PHY payload (the MAC protocol data unit), a PHY packet includes a 5 byte synchronization header (SHR) which allows devices to synchronize with the bit stream which forms the message.

2.2. 802.15.4 MAC

The MAC protocol specifies when devices may access the channel for communication. The basic services provided by the MAC are beacon generation and synchronization, supporting PAN association and disassociation, supporting optional device security, managing channel access via CSMA-CA, maintaining guaranteed time slot (GTS) communication, providing message validation, and providing message acknowledgments.

A PAN may be set up in one of two basic configurations: beacon-enabled and nonbeacon-enabled. In a nonbeacon-enabled network, devices may communicate with each other at any time after an initial association phase. Channel access and contention are managed using an unslotted CSMA-CA mechanism and any node-level synchronization must be performed at some higher layer. In a beacon-enabled network, the PAN coordinator periodically transmits a beacon which other devices use both for synchronization and for determining when to enable transmission and reception of messages. This beacon message is used to define a superframe structure that all nodes in the PAN should synchronize to. This superframe structure is shown in Figure 2.

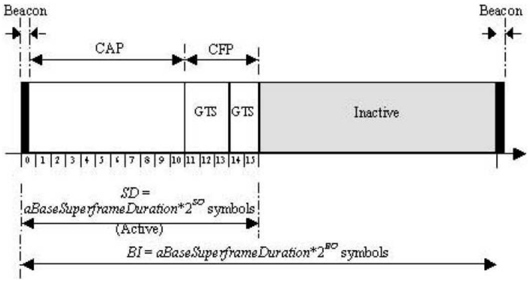
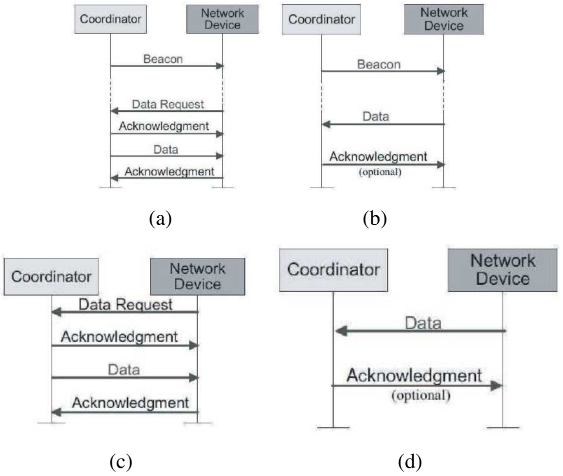


Figure 2. The 802.15.4 MAC superframe structure ([4])

The superframe is divided into several sections, the lengths of which are configurable. There is an active period, during which communication takes place, and an inactive period, during which devices may turn off their transceivers in order to conserve energy. The active period is divided into 16 equally-spaced slots. Immediately following the beacon is the contention access period (CAP). During this period, devices may communicate using a slotted CSMA-CA mechanism. This is similar to unslotted CSMA-CA, except that the back-off periods are aligned with slot boundaries, meaning that the devices are contending for the right to transmit over entire slots. The CAP must contain at least nine active period slots but may take up all 16. Following the CAP is an optional contention free period (CFP), which may last up to seven active period slots. In the CFP, devices are allocated GTS slots by the PAN coordinator. During a GTS a device has exclusive access to the channel and does not perform CSMA-CA. During one of these GTSs, a device may either transmit data to or receive data from its PAN coordinator, but not both. The length of a GTS must be an integral multiple of an active period slot. All GTSs must be contiguous in the CFP and are located at the end of the superframe active period. A device may disable its transceiver during a GTS designated for another device in order to conserve energy.

As mentioned previously, all devices must go through an initial association phase in order to become part of a PAN. This association is prompted by a higher layer service, but it uses primitives defined in the MAC to perform the associations. The MAC allows configurations to be set for starting a device as a PAN as a coordinator, allowing a coordinator to have devices associate with it, and performing the actual association of a device with some coordinator. Once part of a PAN, the way in which data is sent between a device and its coordinator performed in one of the ways shown in figure Figure 3. Note that acknowledgments are optional in all transfers from a device to its coordinator, but they are required in transfers from the coordinator. When transferring from a coordinator to a device, the device must first request the data from the coordinator. In a nonbeacon-enabled network, devices must poll the coordinator for data at an application-specified rate, as there are no beacons to indicate to the device that there is data pending for it.



(a) (b)

(c) (d)

Figure 3. Data transfer between a coordinator and its devices a)Coordinator to Device with Beacons enabled b) Device to Coordinator with Beacons enabled c) Coordinator to Device with Beacons disabled d)Device to Coordinator with Beacons disabled ([4])

2.3. Configuring 802.15.4

The interface between the MAC and any layers implemented on top of it is known as the Service Specific Convergence Sublayer (SSCS). This provides access to the primitives defined for the MAC. Primitives are essentially functions that are used to interact with the MAC. They are used to perform actions such as making a request, receiving a notification, and examining or modifying a MAC attribute.

Two attributes of particular importance are the $macBeaconOrder(BO)$ and the $macSuperframeOrder(SO)$. These two attributes define the interval at which beacons are sent by a coordinator and the length of a superframe's active period, respectively. As shown in the equations from Figure 2, these two periods can be defined as:

$$aBaseSuperframeDuration * 2^{[BO,SO]}$$

When beacons are enabled, BO and SO range from 0 to 14, with the constraint that $SO \leq BO$. Table 1 shows the the lengths of the superframe and each active slot when

SO/BO	868 MHz	915 MHz	2450 MHz
0	0.048	0.024	0.01536
1	0.096	0.048	0.03072
2	0.192	0.096	0.06144
3	0.384	0.192	0.12288
4	0.768	0.384	0.24576
5	1.536	0.768	0.49152
6	3.072	1.536	0.98304
7	6.144	3.072	1.96608
8	12.288	6.144	3.93216
9	24.576	12.288	7.86432
10	49.152	24.576	15.72864
11	98.304	49.152	31.45728
12	196.608	98.304	62.91456
13	393.216	196.608	125.82912
14	786.432	393.216	251.65824

Table 2. Beacon Interval Times/Super Frame Durations (in seconds) for available BeaconOrder and SuperFrameOrder settings

SO is zero for all three frequency bands. Using this information, we are able to calculate all the possible beacon intervals/superframe durations for each frequency band, as shown in Table 2. These times are useful when trying to configure the MAC for an application. In a query service application, for example, source nodes would only want to generate periodic data at a rate equal to that of the beacons in a beacon-enabled network. They would then be free to sleep at all other times.

3. Introduction to LR-WPAN Routing Protocols

Above the PHY and MAC layers in the 802.15.4 standard, services can be built that exploit the configurability discussed in the previous section.

One routing protocol especially popular in tandem with 802.15.4 is Zigbee[2]. In fact, 802.15.4 is sometimes called Zigbee even though Zigbee specifically refers to the routing protocol and 802.15.4 refers to the MAC and PHY protocols. Like 802.15.4, Zigbee is an industry standard. Zigbee is targeted to low data rate, low power consumption, low cost, long lived wireless networks. Zigbee employs a basic master-slave configuration suited to static multiple-source, single-sink topologies. Each master can support up to 254 slaves and these can be nested to support a hierarchical routing strategy. A unique feature of Zigbee is that it provides communication redundancy, eliminating single points of failure in mesh networks. The routing algorithms defined for use by Zigbee are the Ad-Hoc On Demand Distance Vector (AODV) protocol and the Cluster Tree protocol. Documentation for these two protocols is provided in the Zigbee documentation.

MinT [8], a.k.a. MT, (for Minimum Transmission) is a routing protocol used for Wireless Sensor Networks (WSNs). It is conceivable that MinT could be built on top of the 802.15.4 MAC layer and could benefit from the configuration settings provided by the 802.15.4 standard. MinT relies upon link quality information gained from the MAC layer to form appropriate neighbor tables and determine routing. The routing protocols are distributed distance vector-based which utilize estimated routing costs plus reception link estimations from neighbors.

With the introduction of these two routing protocols for LR-WPANS, we now have a better context within which to describe our work configuring and testing 802.15.4 for the three performance criteria mentioned in the introduction (i.e., efficient reliability, throughput, and power management). How such routing protocols can actually use these configuration settings to implement their routing protocols will be introduced in the Results and Recommendations sections.

4. Related Work

Some work has already been done in evaluating the performance of the 802.15.4 standard. The authors of [9] developed the ns2 support for 802.15.4 that is used in our study. They also performed simulation experiments to test beacon/nonbeacon transmission, association and tree formation, orphaning, and CSMA. They compare the results to the performance of IEEE 802.11 in the same scenarios.

In [6], they concentrate on beacon-enabled transmission in star networks. They examine the trade-offs between power consumption and throughput or latency. They determine that while duty cycling the nodes via the superframe can yield significant energy savings, the cost in energy of synchronizing to the beacons is not negligible.

An algorithm to adapt the beacon interval at runtime based on the network workload and the required duty cycle is proposed in [7]. Again, this work is concentrated on star topologies, where enabling beacons is most useful.

Another performance evaluation is done in [3], but this time it is to determine the suitability of 802.15.4 in medical applications. Their focus is on interoperability and scalability, since there can potentially be many different wireless applications used in patient care.

The study in [1] concentrates on power consumption. They determine the minimum expected power consumption in a typical WSN scenario and examine how energy is used in different phases of data transmission.

While this work is all useful for future application developers, the authors stop short of giving explicit recommen-

Frequency (MHz)	BitRate (kbits/s)	aBaseSuperFrameDuration (ms)	aBaseSlotDuration (ms)
868	20	48	3
915	40	24	1.5
2450	250	15.36	0.960

Table 1. The bitRate, super frame durations, and slot durations at the different 802.15.4 operating frequencies

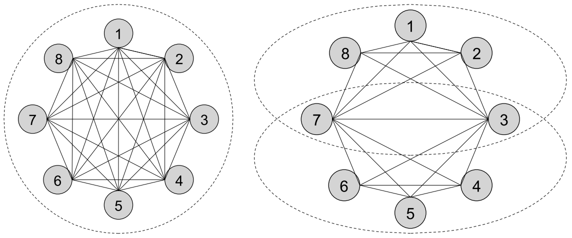
dations when working with the standard.

5. Background of Experimental Setup

The topology of a wireless network can be modeled as a directed, weighted graph of the form $G = (V, E)$, where each vertex $v \in V$ represents one of the nodes in the network, and each directed edge $e = (u, v) \in E$ represents the wireless link from node u to node v in the network. The weight w assigned to edge e represents the strength of the wireless link from node u to v. All weights range between 0 and 1, and the lower the weight, the less likely it is that node u will be able to successfully send a packet to node v.

In the case where all wireless links between any two nodes u and v are symmetrical (i.e. when $w_{(u,v)} = w_{(v,u)}$ for every pair of vertices $(u \in G, v \in G)$), the graph representing such a network no longer needs to be viewed as directed, and a single undirected edge with weight $w_{(u,v)}$ can be placed between vertices u and v in G.

In the special case of an ideal, fully-connected network topology, the strength of the wireless link between any two nodes in the network is infinite. Every node is able to communicate with any other node in the network over a perfect link, as well as overhear all of the messages sent between any other nodes in the network. An example of such a network can be seen in figure 4(a).



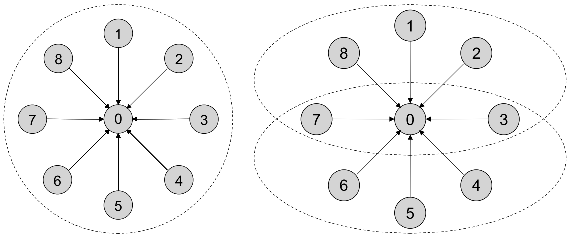
(a) (b)

Figure 4. a) Ideal fully-connected network topology. b) Non-ideal fully-connected network topology. All edges that shown have weight equal to infinity. All edges not shown have weight equal to 0.

In a non-ideal, fully-connected network topology, not all nodes are able to communicate with each other. Each node is, however, indirectly connected to every other node if some intermediate node is used as a forwarder. An example of such a network can be seen in figure 4(b).

The actual flow of messages through a network can be represented by a directed graph $G' = (V, E')$ that contains the same vertex set V as G, but each edge $e \in E'$ represents the flow of a messages from a source to its destination.

One way in which an ideal, fully-connected network topology might operate is to have one of the nodes in the network be designated as a sink node, and all other nodes in the network be designated as source nodes. All sources send messages only to the single sink node. Since any individual node can overhear all messages sent by any other nodes in the network, each node will be able to decide whether it is able to send to the sink at any given time by checking if anyone else is currently sending or not. We will refer to such a setup from now on as an ideal star network topology. Such a topology can be seen in figure 5(a).



(a) (b)

Figure 5. a) Ideal star network topology. b) Non-ideal star network topology. Connectivity of nodes represented by dashed circle. Arrows represent sending of messages

The notion of a non-ideal star network topology can be developed in the same way. Given a non-ideal, fully-connected network, designate one of the nodes in the network as a sink node and all others as source nodes. Although it is not necessarily true that each node in the network will be able to communicate directly with every other node in the network, the single sink node must be able to do so. An example of this sort of setup can be seen in figure 5(b).

It is with these two types of topologies that we focus our research (i.e. ideal star and non-ideal star topologies). We believe that these two types of topologies form the basic building blocks for the more generalized network topologies that can be built out of configuring the 802.15.4 MAC layer appropriately. By evaluating the performance of the standard when operating in a network with one of these two topologies, we are able to infer some generalizations about the capabilities of the standard as a whole.

6. Experimental Setup

The ns2 simulator was used to gather information about the performance of ideal and non-ideal star network topologies built on top of the 802.15.4 LR-WPAN standard. Ns2 version 2.29 was used with a custom routing agent implementation compiled into it. All code (along with installation instructions and usage notes) used to perform these evaluations can be found in [5].

A total of 19,200 simulations were run producing 19,200 individual trace files from which information about packet transmission, reception, and collision could be analyzed. In each scenario there was a single sink node and multiple sending nodes. Each simulation varies one of the following variables concerning the configuration of the 802.15.4 MAC.

- Ideal Star Topology [true, false]
- Number of sending nodes in the network [1..20]
- Beacon Enabled [true,false]
- Synchronized Sending [true,false]
- Beacon Order [0..14]
- Superframe Order [0..14] ($SO \leq BO$)

The total number of simulations comes from varying each of these variables over their entire range of values:

$$20 * 2 * 2 * 2 * 15 * (15 * 16/2) = 19200$$

All sending nodes in the ideal star topologies are distributed in a circular fashion around a single sink node at a distance of 5 meters. In the non-ideal star topologies, half of the sending nodes are arranged in a half circle 3.5 meters above the sink node, and the other half are arranged in a half circle 3.5 meters below the sink node. With an odd number of nodes, the extra node is added to the half circle placed above the sink node.

In each scenario, the PAN coordinator is used as the sink node for the network, and multiple RFD devices are used as source nodes. All simulations start at time 0 with the PAN coordinator starting up, followed by each of the RFD devices every 0.5 seconds in turn. Whenever a device is started it begins to associate with the PAN coordinator. We allow 20 seconds for all RFD devices to associate with the PAN before the transmission of data packets begins.

When running simulations in Beacon Enabled mode, beacon transmission begin just before the sending of any data packets. A beacon message is sent at a time equal to:

$$T_B = 20 - beaconSize * 8 * bitRate$$

In our experiments, $beaconSize$ is always equal to 12 bytes, and the $bitRate$ is equal to $250kbps$. Data packets are sent for a total of 80s, so all simulations run for a total time of 100s.

All data packets contain 43 bytes, and are sent by all sending nodes at a rate equal to the beacon interval of the 802.15.4 MAC. If synchronized sending is not turned on, then all nodes start sending at the exact same time, and the built in CSMA-CA protocol is the only mechanism in place to keep nodes from interfering with one another.

The notion of synchronized sending pertains to staggering the sending times of all sending nodes in the network to ensure that they do not interfere with one another when they are sending. In this set up, CSMA-CA would not be necessary because all nodes would be guaranteed to send at a time when no other nodes were sending. Since GTSs are not supported in the version of the 802.15.4 implementation for ns2 that we are using, performing experiments with this feature turned on allows us to simulate their use. It also allows us to evaluate how a TDMA protocol built on top of 802.15.4 might perform if one wished to implement such a scheme.

7. Results

This section presents the various graphs generated from data gathered by running all of our experiments. It was found that varying SO from 0 up to the value of BO for each experiment turned out to be a little erroneous since any time $BO \neq SO$, all nodes simply sleep and do not attempt to send any data during the offtime between beacon intervals. For this reason, all graphs shown below pertain to data gathered whenever $BO = SO$, for the cases when all nodes were always active. Since one of our objectives was to evaluate the performance of the 802.15.4 standard in terms of power management, however, the upcoming recommendations section uses our experience with the standard as a basis for how to configure the MAC for power management, rather than any direct results from the data.

The first 6 graphs given in figures 6, 7, 8, 9, 10, and 11 show graphs of the total percentage of packets received (%rx) vs. the total number of sending nodes in the network (Number of Nodes). The second 6 graphs given in figures 12, 13, 14, 15, 16, and 17 show graphs of the total percentage of packets received (%rx) vs. the case where $BO = SO$ for values of $BO = SO$ from 0 up to 14. Each data point on the first six graphs represents the exact same data as the data points on the second six graphs, except that they have been arranged in a different manner. They each display how many packets have been received when the network contains between 0 and 20 sending nodes and $BO = SO$. The caption below each chart indicates whether the graph shows data for a network with the following features:

- Ideal vs. Non-Ideal star topology
- Beacon Enabled vs. Non-Beacon Enabled mode
- Synchronized Sending vs. Non-Synchronized Sending

One thing to notice about each of the graphs is that when exactly two nodes are sending, packet receptions seems to be alot lower than expected. We have attributed this to a bug in the CSMA-CA scheme implemented in the ns2 simulator, but have not yet investigated this claim fully. We leave this investigation as something to do for future work.

The following section uses the results shown in these graphs to analyze the performance of the 802.15.4 standard and make recommendations for its improvement in regards to 1) reliability, 2) throughput maximization, and 3) power management.

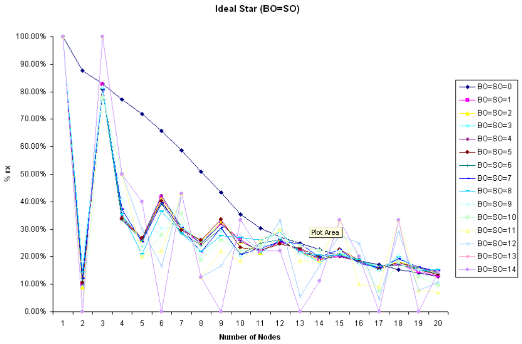


Figure 6. Ideal star network topology in Non-Beacon Enabled mode without Synchronized Sending

8. Recommendations for Improvements to the 802.15.4 Standard

There are several recommendations that emerge from our work with 802.15.4. These recommendations fall into two categories: 1) recommendations for the 802.15.4 standard and 2) recommendations for an application that uses the standard. We will address these here in that order.

The first recommendation we have for 802.15.4 is to enable GTSs to be propagated to nodes that are not associated directly with the PAN coordinator. In other words, allow

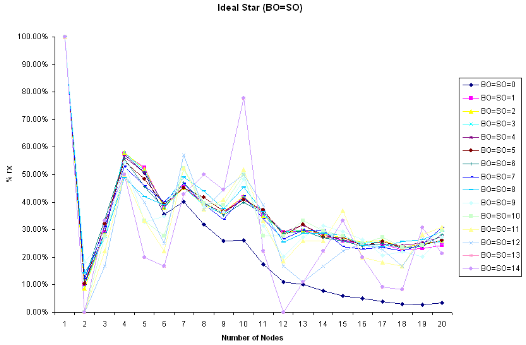


Figure 7. Non-Ideal star network topology in Non-Beacon Enabled mode without Synchronized Sending

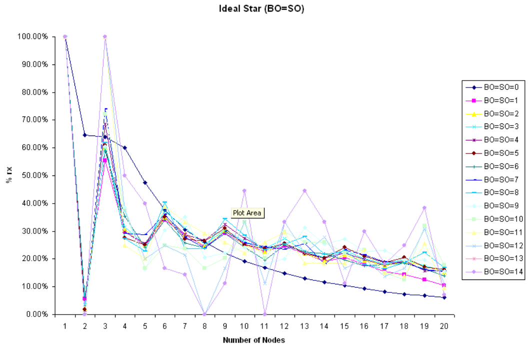


Figure 8. Ideal star network topology in Beacon Enabled mode without Synchronized Sending

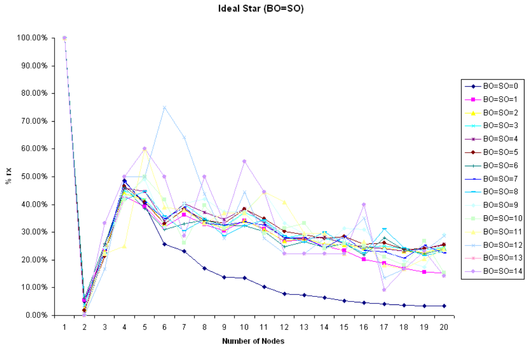


Figure 9. Non-Ideal star network topology in Beacon Enabled mode without Synchronized Sending

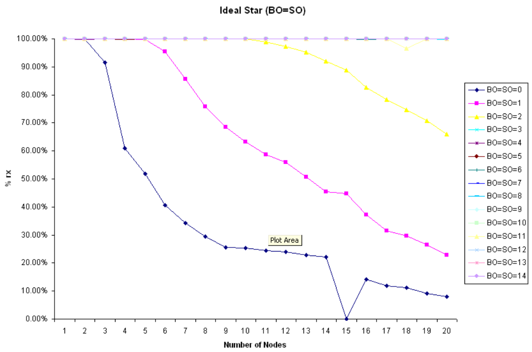


Figure 10. Ideal star network topology in Beacon Enabled mode with Synchronized Sending

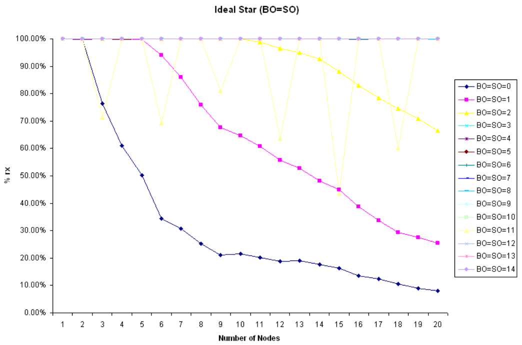


Figure 11. Non-Ideal star network topology in Beacon Enabled mode with Synchronized Sending

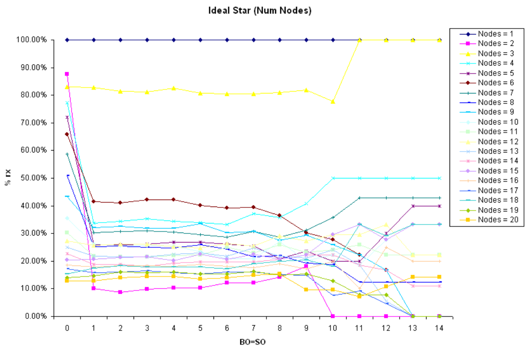


Figure 12. Ideal star network topology in Non-Beacon Enabled mode without Synchronized Sending

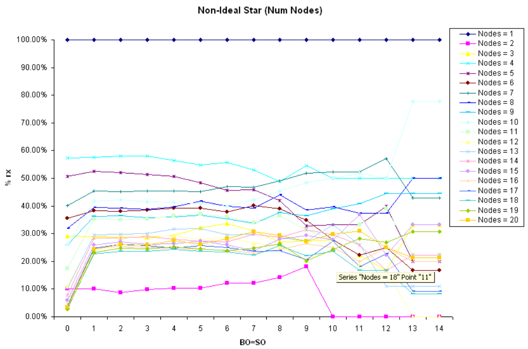


Figure 13. Non-Ideal star network topology in Non-Beacon Enabled mode without Synchronized Sending

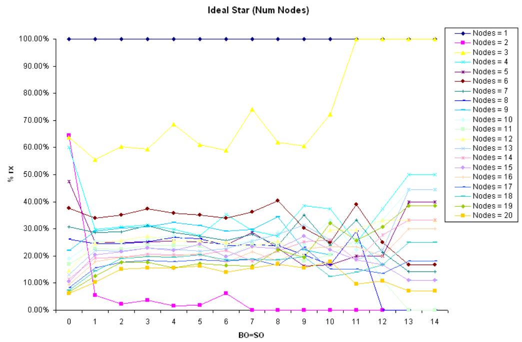


Figure 14. Ideal star network topology in Beacon Enabled mode without Synchronized Sending

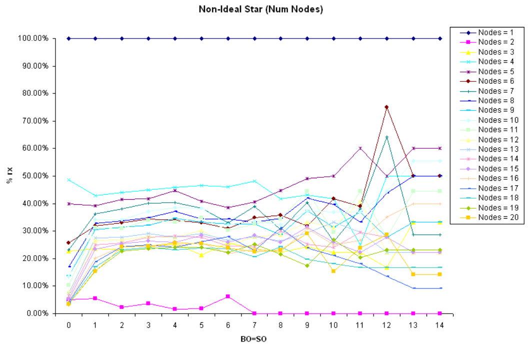


Figure 15. Non-Ideal star network topology in Beacon Enabled mode without Synchronized Sending

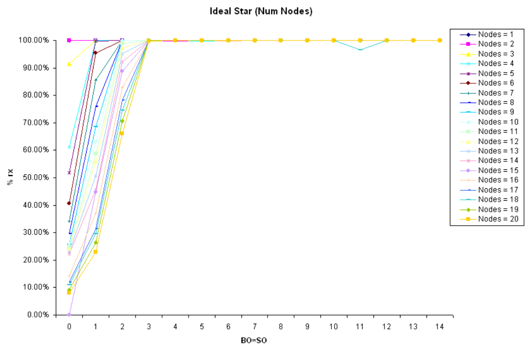


Figure 16. Ideal star network topology in Beacon Enabled mode with Synchronized Sending

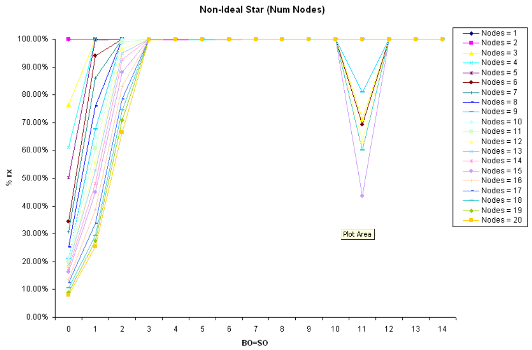


Figure 17. Non-Ideal star network topology in Beacon Enabled mode with Synchronized Sending

nodes that associate with an FFD to also have GTSs. Currently, GTSs are not supported for nodes that are not directly associated with the PAN coordinator. Having this could be useful for applications with hierarchical single-sink topologies (e.g., star network that sends to a single sink who in turn with other sinks send packets to its sink, etc.).

The second 802.15.4 recommendation is to allow more flexibility in configuring the slots of the SuperFrame. As 802.15.4 exists now the MAC can not be configured to only have GTSs. The CAP is always required if beacons are enabled. This can be improved by applications with time slot coordination for the nodes during the CAP. Specifically as shown in the work for this paper, TDMA can be simulated essentially to add GTSs to the CAP. However, it would be much cleaner, easier, and less error-prone if this was available natively from the MAC. There may be reasons to have some CAP time to allow the PAN coordinator send control messages but this should be left to the application developer.

The third 802.15.4 recommendation we make is to provide for beacons from the PAN coordinator to propagate to nodes that aren't directly associated with the PAN coordinator. Currently, there is no provision for an FFD to forward PAN coordinator beacons to the FFD's associated nodes. An FFD can only send out its own beacons. Work would be needed by the application to coordinate (possibly multiple) FFD's beacons with PAN Coordinator's beacons. We realize that synchronizing beacons across multi-hop connections is not trivial but we also believe this is an open and interesting area for research.

We now turn out attention to recommendations for applications using 802.15.4. Our first recommendation in this area pertains to maximizing throughput. To maximize throughput/goodput an application should configure 802.15.4 to use slotted CSMA and perform the calculations needed to stagger the output intervals of the different nodes appropriately. This is part of the work mentioned earlier in the Results section. It should be noted that doing this incorrectly can decrease the goodput even below what unslotted CSMA would be (due to contention only occurring during slot boundaries).

Our second recommendation is related to reliability as mentioned above but focuses specifically on power management. When duty cycling the nodes, the application should use either GTS or TDMA to ensure that the data is delivered reliably. In essence, make sure the slots times allocated fit the bandwidth requirements of the packets that need to be sent as closely as possible (without being smaller than needed). Otherwise, if no time division of the active period is used, ensuring that the active slot lengths are long enough to transmit a packet (and an optional acknowledgment) should yield a throughput as high as if there is no duty cycling.

Our final application recommendation deals with the analysis of topologies and configurations to maximize efficient reliability. This analysis also motivates the inclusion of GTS propagation for nodes that do not directly communicate with the PAN coordinator. First we define efficient reliability. By the term efficient reliability we mean reliability with the number of retransmissions minimized. The 802.15.4 standard does support acknowledgments and we could enable these to get reliability. However, an interesting goal is to minimize the bandwidth and latency properties of reliability. This implies not using acknowledgments since these take up bandwidth and increase latency of packets that need to be retransmitted due to the lack of an acknowledgment being received. This can be done if the topology and 802.15.4 MAC are configured properly. There are also certain assumptions that are made in order to make this proposition feasible.

First, we make the assumption that the sink nodes will aggregate data coming from their sources and that the packet with the aggregated data that a sink sends out will be no bigger than the packets that its sources were sending to it. Otherwise, if the sink simply forwards all the packets it receives we quickly run out of GTS since the number of data packets that a single sink will need to forward will eventually overflow the bandwidth available in the GTS.

Second, we also make the assumption that support is available to propagate GTS for nodes that do not directly interact with the PAN coordinator. 802.15.4 currently only supports GTSs for nodes that are directly communicating with the PAN coordinator and there can only ever be a single PAN coordinator in the entire network. If the topology of the network is such that not all nodes can directly interact with the PAN coordinator (as is the case for arbitrary hierarchical networks) then not all nodes can participate in the GTSs. We believe the topology and configuration outlined below is a motivating example to make GTS support available not just for the PAN coordinator but also for any FFD.

Finally, we make the assumption that there is little or no interference with transmissions outside of the star topologies. Depending on the application and environment this may or may not be a valid assumption.

Since we want to eliminate the need for and overhead of acknowledgments we turn them off. Instead, we use the GTSs provided by the 802.15.4 MAC. If the topology is such that there are six sources per sink we can arrange these in a hierarchical manner to create an arbitrarily large tree of source nodes propagating input to the single ultimate sink. We limit the number of source nodes per sink node to 6 since 802.15.4 supports at most 7 GTSs and we need to leave one for the sink to forward its data to its sink.

As shown in figure 18 we can alternate the use of the GTSs so that for the different child star topologies such that the parent sink/FFD can use a GTS for each one of its six child stars. In this manner it is possible to use GTS to ensure reliability and since GTSs are used retransmissions should theoretically non-existent and in practice should be minimal. This configuration described also leaves bandwidth in the CAP for other messages. Alternatively, as discussed elsewhere in the paper the CAP can be configured to simulate TDMA which is similar to GTS. Adding this functionality would then increase the maximum number of nodes in a single star topology to be 15 (i.e., 16 being the number of slots for the beacon order minus one for the sink to use to transmit) rather than the current maximum of 6.

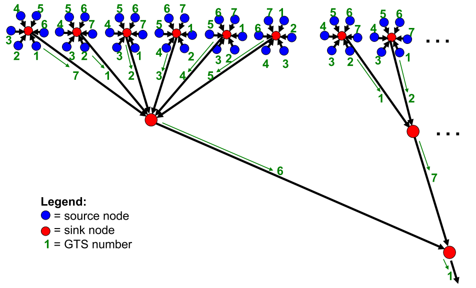


Figure 18. Allocation of GTS time slots for minimizing retransmissions in a single-sink multihop network

It should be noted that even if 802.15.4 supported the use of GTSs with FFDs there would still need to be some configuration work to make sure the GTSs used at different places in the network hierarchy do not overlap inappropriately. However, having 802.15.4 support GTSs from FFDs would make this job feasible.

Finally, in general an application should be careful how it configures 802.15.4. There are lots of options from which to choose. There are cases where conflicting or non-sensical configurations can be created. An area for further research is creating an API that would eliminate these kinds of errors.

9. Future Work

There are several areas for future work. Some of these have been mentioned in the Recommendations section. We list other areas here.

Our first area for future work is to add full support of 802.15.4 to ns. For example, there are several experiments that we would have run if ns would have supported GTSs. We do not have an estimate as to how long this work would take or how hard it would be. There may be other areas of the 802.15.4 standard that need to be incorporated into ns2 as well. We have not done an exhaustive comparison between the standard and what ns has implemented for it.

Another obvious area for future work is to test out the scenarios and experiments in an actual LR-WPAN. Attempts were made early on to find an 802.15.4 implementation for WSN motes but we were not able to acquire one. If an implementation could be found a good first step would be to recreate the scenarios we ran in ns in a LR-WPAN.

Finally, an interesting area of research would be to build on the work presented in this paper and perform experiments with different and more complicated topologies. Our work focused on star topologies with analysis concerning the creation of arbitrary cluster networks using star topologies as building blocks. Future work could include building up arbitrary cluster networks from the star building blocks and then testing the assertions made in this paper. Our setup (including the ns tcl script and the shell script to cycle through different scenarios) is conducive to picking up where this work left off.

10. Conclusion

The 802.15.4 wireless networking standard has been gaining popularity in industry over the past few years. Although many companies are beginning to use this standard when developing LR-WPAN applications, very few evaluations of its performance have actually been performed to evaluate if it is a good standard for use. In this work, we have tried to provide one such evaluation of the standard in regards to its performance in both ideal and non-ideal star network topologies. Although our work has not been performed in a real world setting, it does provide a good starting point for someone else to pick it up and do so in the future. We have provided recommendations for improvements to the standard based on our experience with it, and have tried to provide suggestions for application developers on how to configure the 802.15.4 MAC when writing applications that use one of the two topologies we have evaluated. We have outlined some future work that could be performed based on our experiments, and hope that someone will use the knowledge we are presenting here to gain a better understanding of how the 802.15.4 LR-WPAN can be configured and used.

References

[1] B. Bougard, F. Catthoor, D. C. Daly, A. Chandrakasan, and W. Dehaene. Energy efficiency of the IEEE 802.15.4 standard in dense wireless microsensor networks: Modeling and improvement perspectives. *Design, Automation, and Test in Europe (DATE)*, 2005.

[2] S. C. Ergen. Zigbee/ieee 802.15.4 summary. 2004.

[3] N. Golmie, D. Cypher, and O. Rebala. Performance evaluation of low rate WPANs for sensors and medical applications. In *Military Communications Conference (MILCOM)*, 2004.

[4] IEEE Computer Society. *IEEE Std. 802.15.4-2003, IEEE Standard for Information Technology - Telecommunications and Information Exchange between Systems - Local and Metropolitan Area Networks - Specific Requirements - Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (WPANs).* Institute of Electrical and Electronics Engineers, Inc., New York, 2003.

[5] K. Klues, J. Hoffert, and O. Orjih. `http://www.cs.wustl.edu/~klueska/cs520/Project/802_15_4_Eval.tar.gz`. Cse 520 Course Project, Washington University, St. Louis, MO, Fall 2005.

[6] G. Lu, B. Krishnamachari, and C. Raghavendra. Performance evaluation of the IEEE 802.15.4 MAC for low-rate low-power wireless networks. In *Workshop on Energy-Efficient Wireless Communications and Networks (EWCN), held in conjunction with the IEEE International Performance Computing and Communications Conference (IPCCC)*, 2004.

[7] M. Neugebauer, J. Plnnigs, and K. Kabitzsch. A new beacon order adaptation algorithm for IEEE 802.15.4 networks. In *European Workshop on Wireless Sensor Networks (EWSN)*, 2005.

[8] A. Woo, T. Tong, and D. Culler. Taming the underlying challenges of unreliable multihop routing in sensor networks. In *ACM Conference on Embedded Networked Sensor Systems (SenSys)*, 2003.

[9] J. Zheng and M. J. Lee. A comprehensive performance study of IEEE 802.15.4. *IEEE Press Book*, 2004.