Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2014, Article ID 138571, 12 pages
http://dx.doi.org/10.1155/2014/138571

Research Article

Taxonomy and Evaluations of Low-Power Listening Protocols for Machine-to-Machine Networks

Kwang-il Hwang and Sung-Hyun Yoon

Department of Embedded Systems Engineering, Incheon National University, Incheon 402-772, Republic of Korea

Correspondence should be addressed to Kwang-il Hwang; hkwangil@incheon.ac.kr

Received 2 April 2014; Accepted 4 June 2014; Published 8 July 2014

Academic Editor: Young-Sik Jeong

Copyright © 2014 K.-i. Hwang and S.-H. Yoon. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Even though a lot of research has made significant contributions to advances in sensor networks, sensor network protocols, which have different characteristics according to the target application, might confuse machine-to-machine (M2M) network designers when they choose the protocol most suitable for their specific applications. Therefore, this paper provides a well-defined taxonomy of low-power listening protocols by examining in detail the existing low-power sensor network protocols and evaluation results. It will also be very useful for helping M2M designers understand specific features of low-power media access control protocols as they design new M2M networks.

1. Introduction

Machine-to-machine (M2M) networks enable creation of the Internet of Things, which interconnects via the Internet physical things equipped with various sensors and actuators. Mitsui et al. [1] presented various M2M applications based on sensor network technologies. A typical M2M architecture is basically composed of three domains: the server, the Internet, and the sensors. In particular, the sensor domain is the most important, aggregating data from physical sensors and accessing the Internet via 3G or 4G M2M gateways. Like a sensor network, an M2M sensor domain requires well-structured and energy-efficient network protocols among distributed sensors using short range communications. Much research has already been conducted on sensor network protocols [2], making significant contributions towards advances in automated sensor networks [3–5]. However, having too many sensor network protocols causes confusion for M2M designers as they choose the protocol most suitable for their specific applications. Furthermore, most of the literature on sensor network protocols is too theoretical, requires a lot of specific assumptions, and is not easy to apply to practical M2M sensor domains.

Sensor media access control (MAC) protocols can be categorized into random-based, slot (schedule-) based, time division multiple access- (TDMA-) based, random/TDMA hybrids, and low-power listening (LPL) methods. In particular, LPL-based MAC protocols can be considered the most suitable type for M2M sensor domains, because they provide a low duty cycle and low implementation complexity. Therefore, there has been substantial research on LPL protocols. Each one shows different characteristics and operations, as described in Table 1. Therefore, this paper aims to provide a well-defined taxonomy of low-power listening protocols by examining in detail the existing low-power sensor network protocols, introducing an M2M communication model and then evaluating performance with respect to data aggregation time and energy consumption in terms of an M2M communication model.

The remainder of this paper is organized as follows. A taxonomy of LPL protocols is presented in Section 2. Section 3 analyzes each LPL protocol in terms of an M2M communications model. Section 4 summarizes numerical results and Section 5 provides concluding remarks.

2. A Taxonomy of Low-Power Listening Protocols

2.1. Trigger Source (Preamble versus Packet). The main idea of LPL is to asynchronously trigger a receiver that is alternating

Table 1: LPL MAC protocols.

Protocol	Features
B-MAC [6]	(i) Berkeley MAC (ii) LPL with check time, back-off window size, and power-down policy in the application level (iii) Advanced clear channel assessment (CCA) for dealing with random noise
Wise-MAC [7]	(i) Improved LPL, remembering neighbors' polling schedules (ii) Sends short preamble when the receiver wakes up
X-MAC [8]	(i) Upgraded B-MAC protocol (ii) Divides long preamble into two parts (micropreamble/receiver address) to solve overhearing
SpeckMAC [9]	(i) Consists of SpeckMAC-B, SpeckMAC-D (ii) Consecutive data frame, wake-up packet (iii) Sender accesses receiver with 3 bytes preamble in the packet frame
RI-MAC [10]	(i) Receiver-initiated MAC protocol (ii) Receiver sends periodic beacon frame and sender sends data frame if beacon frame is received
BoX-MAC [11]	(i) Cross-layer MAC protocol using PHY, link layer (ii) Consists of two parts (BoX-MAC-1/BoX-MAC-2) (iii) Goes into sleep state in back-off time (iv) Less wake-up time than X-MAC
MX-MAC [12]	(i) An LPL variant of CSMA-MPS (ii) Compatible with X-MAC and SpeckMAC (iii) Consecutive packet transmission instead of short preamble strobe in X-MAC (iv) Sends ACK when packet is received to solve X-MAC's early ACK problem
A-MAC [13]	(i) Receiver-initiated MAC protocol (ii) Using hardware-generated acknowledgment (HACK) for more efficient energy consumption (iii) Saves neighbors' LPL schedules (iv) Deals with hidden terminal problem

between wake-up and sleep states to detect a wake-up signal from a sender. Therefore, receivers can save much more energy by removing idle listening periods. Some protocols, such as B-MAC, WISE-MAC, and X-MAC, use a preamble as a trigger source. On the other hand, SpeckMAC, RI-MAC, BoX-MAC, MX-MAC, and A-MAC trigger receivers by transmitting a consecutive packet. More specifically, RI-MAC, MX-MAC, A-MAC, and SPEC-MAC-D utilize a data packet for the trigger, and SpeckMAC-B, BoX-MAC-1, and BoX-MAC-2 utilize short wake-up packets before data transmission.

2.2. Initiation Method (Receiver-Initiated versus Source-Initiated). LPL protocols can also be categorized into source-initiated and receiver-initiated methods, according to which one begins the transmission request. RI-MAC and A-MAC are receiver-initiated protocols but the rest of the protocols are source-initiated protocols.

2.3. Adaptivity (Adaptive versus Deterministic). B-MAC, SpeckMAC, RI-MAC, A-MAC, and BoX-MAC-1 always transmit triggering signals for predetermined fixed duration, but some protocols, such as WISE-MAC, X-MAC, MX-MAC, and BoX-MAC-2, transmit variable triggering signals depending on when a receiver is triggered.

2.4. Schedule (Schedule versus Nonschedule). To reduce data pending time more, some protocols, such as WISE-MAC and MX-MAC, use schedule-based triggering by exchanging wake-up time information among neighbors.

3. M2M Communication Model

In this section, an M2M communication model is presented, and then each LPL protocol is analyzed in terms of the M2M model.

3.1. System Model. Generally, M2M is composed of a concentrator, which is a centralized device to connect the M2M sensor domain to the Internet, and M2M devices, which are equipped with various sensors or actuators. In an M2M sensor domain, devices form either a star or a peer-to-peer topology for multihop communications. Data from each device are aggregated in the concentrator and transmitted to a corresponding server via the Internet. To consider a practical M2M system, each protocol and algorithm should be able to execute their tasks with off-the-shelf radio frequency (RF) modems (TI CC430, CC2420, RadioPulse MG2400, etc.) and MCUs.

3.2. Data Model. The most popular data models for M2M are the *periodic report model* and the *request-oriented model*. In the periodic report model, each device transmits data to a concentrator periodically, and the model is generally used for unidirectional data aggregation. By contrast, the request-oriented model allows bidirectional communication between the concentrator and devices. In the data model, a server (user) can request a concentrator to aggregate real-time sensor data in the sensor domain. The concentrator also triggers and transmits server requests to the devices. Each device replies to the concentrator, and the responses from

Table 2: Notations.

Notation	Description	Value
T_{PS}	Preamble sensing time	15.6 milliseconds
T_{WS}	Wake-up sensing time	183 milliseconds
T_{FS}	Frame sensing time	183 milliseconds
T_{BS}	Beacon sending time	90 milliseconds
T_S	Sleep duration	Variable milliseconds
T_P	Preamble transmission time for device trigger	(i) *Long preamble (B-MAC)* (ii) *Short preamble (X-MAC)* (iii) *Variable (Wise-MAC)*
T_{WP}	Wake-up packet time for trigger	90 milliseconds
T_{DP}	Data packet time for trigger	150 milliseconds
T_{BL}	Beacon listen time	Random
T_B	Back-off period	1 milliseconds
T_L	Listen period	Variable
T_{DT}	Data transmission time	200 milliseconds
T_{ACK}	ACK transmission time	90 milliseconds
T_{DPT}	Data pending time	Random
T_{DC}	Duty-cycle time	$= T_{\text{PS}} + T_S$ (preamble-based) $= T_{\text{WS}} + T_S$ (wake-up packet-based) $= T_{\text{FS}} + T_S$ (packet-based) = beacon interval (receiver-initiated)
$T_{\text{tot-active}}$	Total active duration	$T_{A\text{-PPS}} + T_{A\text{-aggre}}$ (i)
$T_{\text{tot-sleep}}$	Total sleep duration	$T_{\text{interval}} - T_{\text{tot-active}}$ (ii)
T_{interval}	Request interval	1 hour (=3600 seconds = 3600000 milliseconds)
I_A	Current consumption in active state	0.0061944 mA
I_S	Current consumption in active state	0.0000083 mA
$T_{A\text{-aggre}}$	Active duration during data aggregation time	
$T_{A\text{-PPS}}$	Active duration in a duty cycle	
T_{AW}	Acknowledgement waiting period	100 milliseconds
V	Supply voltage	3.5 V

devices are aggregated in the concentrator and transmitted to the server.

3.3. Energy Model. For M2 M networks, energy conservation is one of the most critical challenges, as it is in sensor networks. It is important to note that in order to save energy, each device should remain active only for required duration, and the rest of the time should go to sleep. Therefore, when calculating the energy consumption of each device, we need to know the total active duration, $T_{\text{tot-active}}$ (i) and the total sleep duration, $T_{\text{tot-sleep}}$ (ii) in a request interval, T_{interval}. By using (i) and (ii), the energy consumption for each device can be expressed as follows:

4. Numerical Analysis

Now, we numerically analyze each LPL protocol in terms of M2M communication models. In particular, we focus on data aggregation time, which is the total time required to aggregate data from all devices with respect to a request. Table 2 presents notations used for our numerical analysis.

4.1. B-MAC. B-MAC is a representative LPL protocol utilizing a preamble for the receiver trigger. As shown in Figure 1, each device repeats a short time wake-up for T_{PS} to detect the preamble transmission and then sleeps for T_s, per T_{DC}. A sender that wants to send data first transmits a long preamble for T_P to trigger the receiver that is performing periodic preamble sensing (PPS) before data transmission. Each preamble transmission can be detected by all devices within communication range of a sender, and all nodes that detect the preamble transmission, as well as the intended receiver, have to remain active for T_L, until the preamble transmission ends.

4.1.1. Periodic Report. Since each device should send its data to the concentrator on the predetermined schedule, the report time of each device is as follows:

$$T_{\text{resp}} = T_B + T_P + T_{\text{DT}} + T_{\text{ACK}}. \quad (1)$$

Therefore, the total report time of n nodes is

$$T_{\text{B-MAC}(P)} = n * T_{\text{resp}} = n * \left(T_B + T_P + T_{\text{DT}} + T_{\text{ACK}}\right). \quad (2)$$

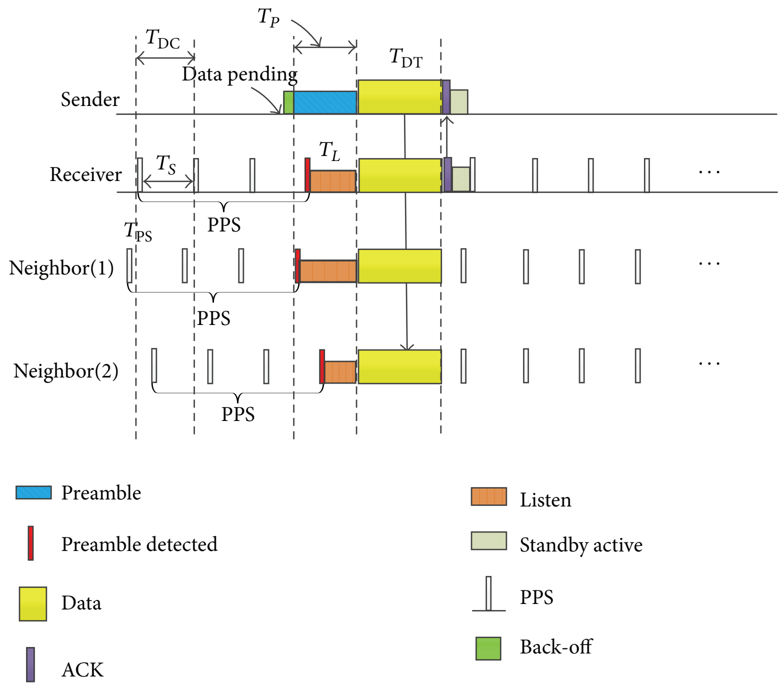


Figure 1: B-MAC.

4.1.2. Request-Oriented. The required time for a concentrator to transmit its request to devices is

$$T_{\text{req}} = T_B + T_P + T_{\text{DT}}. \tag{3}$$

In particular, since B-MAC is capable of triggering all nodes with a single preamble transmission, all the devices can listen to the request message following the preamble. Therefore, the total aggregation time of n nodes per request is

$$\begin{aligned} T_{\text{B-MAC}(R)} &= T_{\text{req}} + n * T_{\text{resp}} \\ &= T_B + T_P + T_{\text{DT}} + n * (T_B + T_{\text{DC}} + T_{\text{DT}} + T_{\text{ACK}}). \end{aligned} \tag{4}$$

4.2. WISE-MAC. Unlike B-MAC, which transmits a long preamble, WISE-MAC aims to save more energy by transmitting a shorter preamble. To achieve this, each sender manages a schedule table in which all its neighbors' PPS schedules are stored. Therefore, if a sender does not know the PPS schedule of a receiver, the sender must transmit a long preamble, as in B-MAC, but otherwise, the sender can transmit a minimum preamble for T_{MP} to trigger the receiver, as shown in Figure 2.

4.2.1. Periodic Report. The report time of a WISE-MAC device is calculated as follows:

$$T_{\text{resp}} = T_B + T_P + T_{\text{DT}} + T_{\text{ACK}}. \tag{5}$$

Therefore, the total report time of n nodes is

$$T_{\text{WISE-MAC}(P)} = n * T_{\text{resp}} = n * (T_B + T_P + T_{\text{DT}} + T_{\text{ACK}}). \tag{6}$$

4.2.2. Request-Oriented. The required time for a concentrator to transmit its request to devices is as follows:

$$T_{\text{req}} = T_B + T_P + T_{\text{DT}}. \tag{7}$$

And, like B-MAC, the WISE-MAC sender can trigger all devices with a single long preamble, so the total aggregation time of n nodes per request is

$$\begin{aligned} T_{\text{WISE-MAC}(R)} &= T_{\text{req}} + n * T_{\text{resp}} \\ &= T_B + T_P + T_{\text{DT}} + n * (T_B + T_{\text{DC}} + T_{\text{DT}} + T_{\text{ACK}}). \end{aligned} \tag{8}$$

4.3. X-MAC. As shown in Figure 3, X-MAC utilizes an early acknowledgment (ACK) and short preamble to reduce energy waste from transmitting a long preamble, as in B-MAC and WISE-MAC in the worst case scenario. In addition, the short preamble represents a destination ID, and a gap, T_{AW}, between short preambles is used to receive the ACK of the receiver.

4.3.1. Periodic Report. The report time of an X-MAC device is as follows:

$$T_{\text{resp}} = T_B + m * (T_{\text{MP}} + T_{\text{AW}}) + T_{\text{DT}} + T_{\text{ACK}}, \tag{9}$$

where m is a maximum time to trigger the receiver and $m \leq T_{\text{DC}}$.

Therefore, the total report time of n nodes is

$$\begin{aligned} T_{\text{X-MAC}(P)} &= n * T_{\text{resp}} \\ &= n * (T_B + m * (T_{\text{MP}} + T_{\text{AW}}) + T_{\text{DT}} + T_{\text{ACK}}). \end{aligned} \tag{10}$$

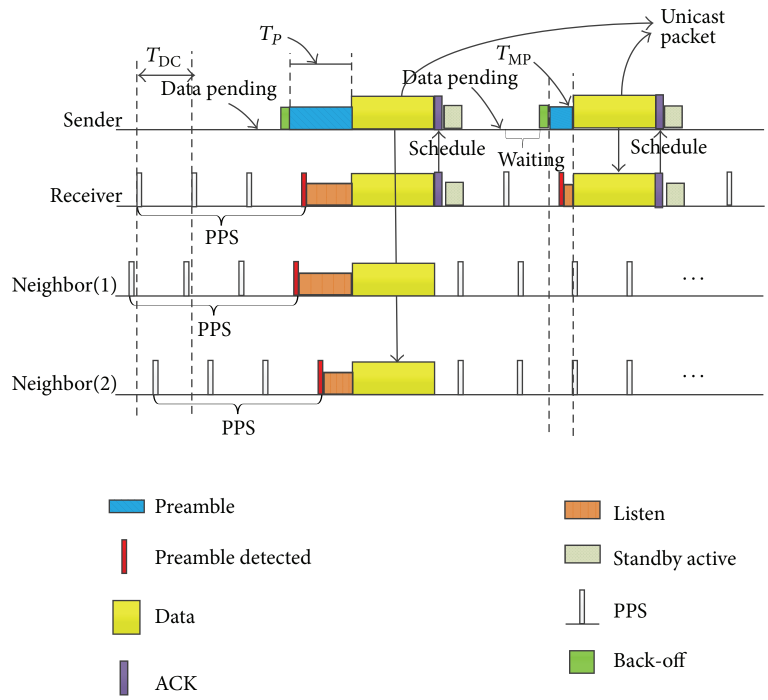


Figure 2: WISE-MAC.

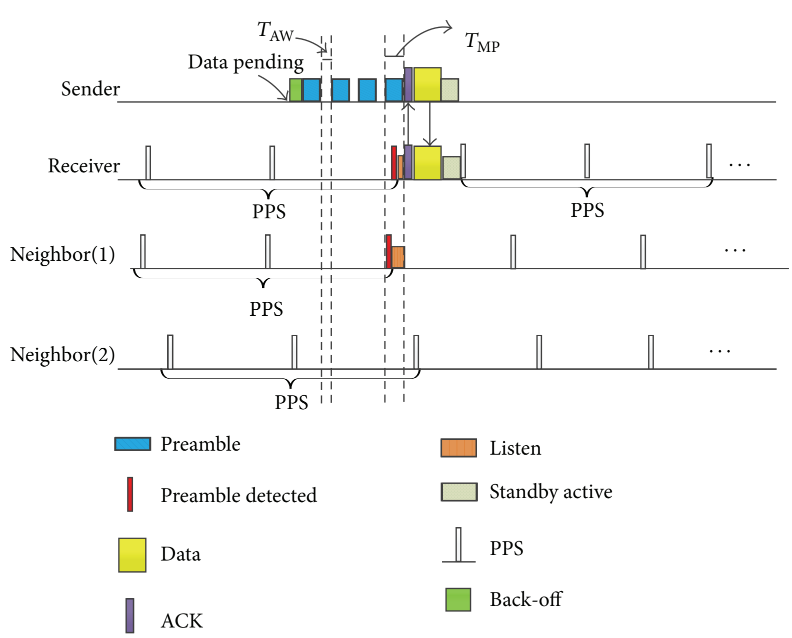


Figure 3: X-MAC.

4.3.2. Request-Oriented. The required time for a concentrator to transmit its request to devices is

$$T_{\text{req}} = T_B + m * (T_{\text{MP}} + T_{\text{AW}}) + T_{\text{DT}} + T_{\text{ACK}}. \quad (11)$$

And, unlike B-MAC and WISE-MAC, X-MAC cannot trigger all devices with a single preamble transmission. Therefore, the request of the concentrator must be transmitted as many times as the number of devices. So the total aggregation time of n nodes per request is

$$\begin{aligned} T_{\text{X-MAC}(R)} &= n * (T_{\text{req}} + T_{\text{resp}}) = 2 * n * T_{\text{req}} \\ &= 2 * n * (T_B + m * (T_{\text{MP}} + T_{\text{AW}}) + T_{\text{DT}} + T_{\text{ACK}}). \end{aligned} \quad (12)$$

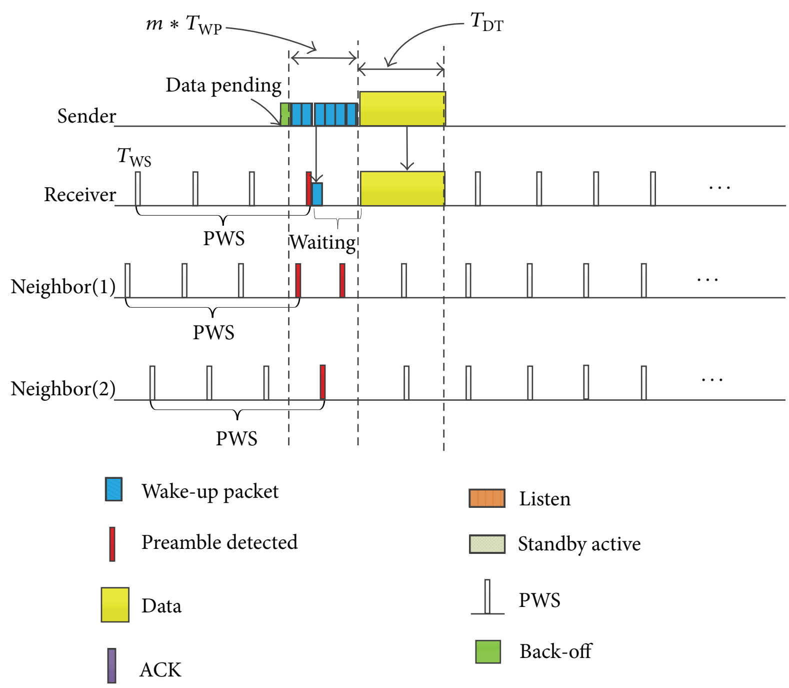


Figure 4: SpeckMAC-B.

4.4. SpeckMAC-B. Instead of a preamble transmission, SpeckMAC-B transmits consecutive wake-up packets to trigger devices performing periodic wake-up-signal sensing (PWS), as shown in Figure 4. A wake-up packet contains a destination ID and a time stamp, which represents data packet transmission time information. Therefore, a device that listens to a wake-up packet during PWS goes to sleep until the beginning of data transmission wakes up and then receives data from the sender. Devices that listen to a wake-up packet but that are not the intended receiver go to sleep and continue to perform PWS.

4.4.1. Periodic Report. The report time of a SpeckMAC-B device is as follows:

$$T_{\text{resp}} = T_B + m * T_{\text{WP}} + T_{\text{DT}}, \quad (13)$$

where m is a maximum time to trigger the receiver and $m \leq T_{\text{DC}}$.

Therefore, the total report time of n nodes is

$$T_{\text{SPECK-MAC-B}(P)} = n * T_{\text{resp}} = n * \left(T_B + m * T_{\text{WP}} + T_{\text{DT}}\right). \quad (14)$$

4.4.2. Request-Oriented. The required time for a concentrator to transmit its request to devices is

$$T_{\text{req}} = T_B + m * T_{\text{WP}} + T_{\text{DT}}. \quad (15)$$

And, like B-MAC or WISE-MAC, a single request packet can trigger all devices, so the total aggregation time of n nodes per request is

$$\begin{aligned} T_{\text{SPECK-MAC-B}(R)} &= T_{\text{req}} + n * T_{\text{resp}} = (n+1) * T_{\text{req}} \\ &= (n+1) * \left(T_B + m * T_{\text{WP}} + T_{\text{DT}}\right). \end{aligned} \quad (16)$$

4.5. SpeckMAC-D. Instead of the wake-up packet transmission used in SpeckMAC-B, SpeckMAC-D enables fast data reception by utilizing consecutive data packets. Each SpeckMAC-D device performs periodic frame sensing (PFS) for T_{FS} to receive a data frame, as shown in Figure 5.

4.5.1. Periodic Report. The report time of a SpeckMAC-D device is as follows:

$$T_{\text{resp}} = T_B + T_{\text{DP}} * m, \quad (17)$$

where m is a maximum time to trigger the receiver and $m \leq T_{\text{DC}}$.

Therefore, the total report time of n nodes is

$$T_{\text{SPECK-MAC-D}(P)} = n * T_{\text{resp}} = n * \left(T_B + T_{\text{DP}} * m\right). \quad (18)$$

4.5.2. Request-Oriented. The required time for a concentrator to transmit its request to devices is

$$T_{\text{req}} = T_B + T_{\text{DP}} * m. \quad (19)$$

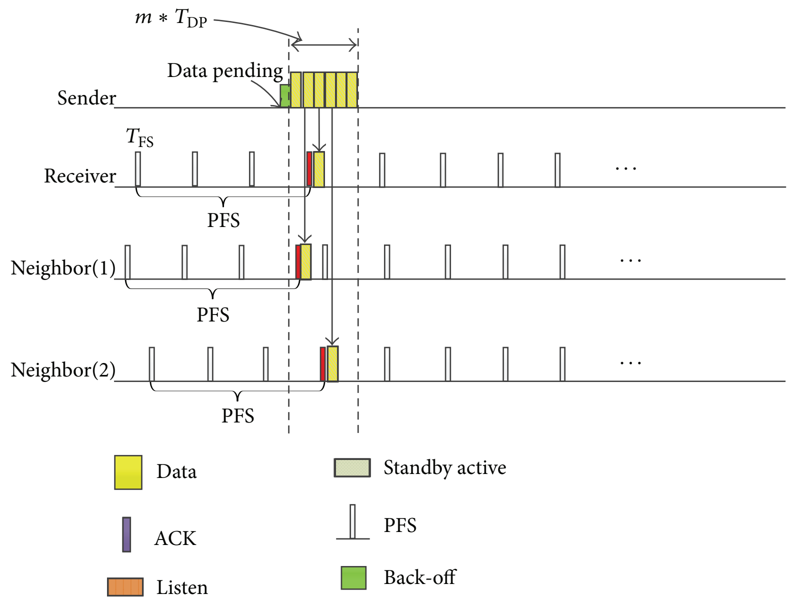


Figure 5: SpeckMAC-D.

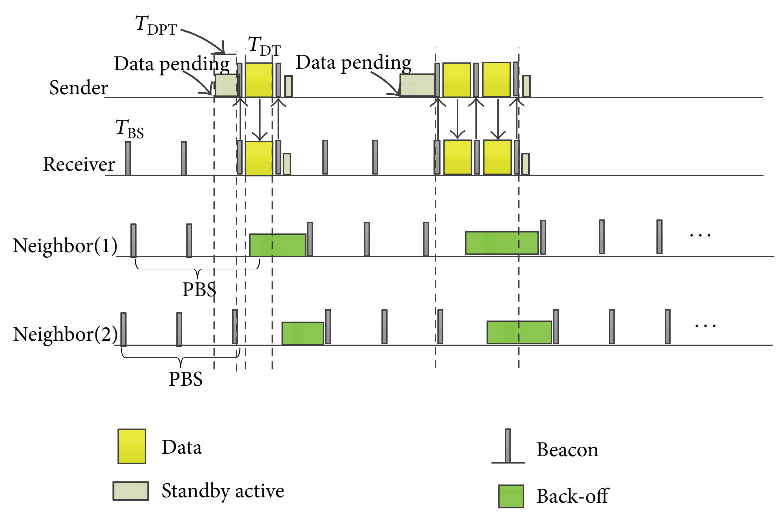


Figure 6: RI-MAC.

And T_{req} is equal to T_{resp}. So the total aggregation time of n nodes per request is

$$T_{SPECK-MAC-D(R)} = T_{req} + n * T_{resp} = (n+1) * T_{req} = (n+1) * (T_B + T_{DP} * m). \quad (20)$$

4.6. RI-MAC. RI-MAC is a representative receiver-initiated LPL protocol. Each RI-MAC device basically performs periodic beacon sending (PBS). A sender first switches to reception (RX) mode and waits to receive the beacon of a corresponding receiver. As soon as a corresponding beacon is received, the sender transmits data and then goes back to PBS. The receiver of the data acknowledges a beacon, as shown in Figure 6.

4.6.1. Periodic Report. The report time of an RI-MAC device is as follows:

$$T_{resp} = T_{DT} + 2 * T_{BS}. \quad (21)$$

Therefore, the total report time of n nodes is

$$T_{RI-MAC(P)} = n * T_{resp} = n * (T_{DT} + 2 * T_{BS}). \quad (22)$$

4.6.2. Request-Oriented. The required time for a concentrator to transmit its request to devices is

$$T_{req} = T_{DT} + 2 * T_{BS}. \quad (23)$$

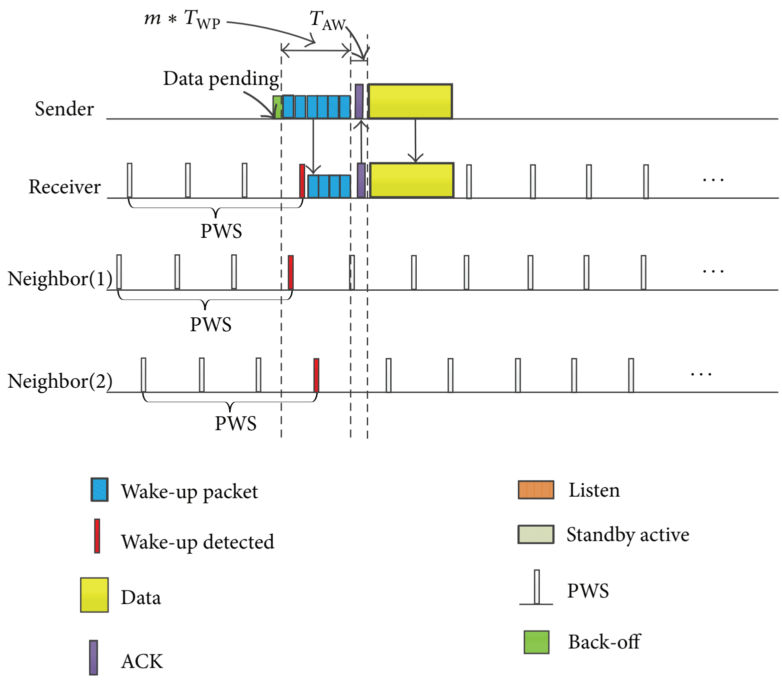


Figure 7: BoX-MAC-1.

And T_{req} is equal to T_{resp}. So the total aggregation time of n nodes per request is

$$T_{\text{RI-MAC}(R)} = n * \left(T_{req} + T_{resp}\right) = 2 * n * T_{req} = 2 * n * \left(T_{DT} + 2 * T_{BS}\right). \quad (24)$$

4.7. BoX-MAC-1. As shown in Figure 7, BoX-MAC-1 is one of the packet-based LPL protocols, and T_{AW} (to wait for ACK of the receiver) is followed by consecutive wake-up packets. Then, on reception of the ACK, the data are transmitted.

4.7.1. Periodic Report. The report time of a BoX-MAC-1 device is as follows:

$$T_{resp} = T_B + m * T_{WP} + T_{ACK} + T_{DT}, \quad (25)$$

where m is a maximum time to trigger the receiver and $m \leq T_{DC}$.

Therefore, the total report time of n nodes is

$$T_{\text{BOX-MAC-1}(P)} = n * T_{resp} = n * \left(T_B + m * T_{WP} + T_{ACK} + T_{DT}\right). \quad (26)$$

4.7.2. Request-Oriented. The required time for a concentrator to transmit its request to devices is

$$T_{req} = T_B + m * T_{WP} + T_{ACK} + T_{DT}. \quad (27)$$

And since data transmission is started only when an ACK is received, the request of the concentrator must be transmitted as many times as the number of devices. So the total aggregation time of n nodes per request is as follows:

$$T_{\text{BOX-MAC-1}(R)} = n * \left(T_{req} + T_{resp}\right) = 2 * n * T_{req} = 2 * n * \left(T_B + m * T_{WP} + T_{ACK} + T_{DT}\right). \quad (28)$$

4.8. BoX-MAC-2. BoX-MAC-2 is one of the wake-up, packet-based LPL protocols. However, unlike BoX-MAC-1 or SpeckMAC-B utilizing consecutive wake-packet transmissions, a sender waits for ACK from the receiver for T_{AW}, per wake-up transmission, as shown in Figure 8. Therefore, a sender repeats wake-up packet transmission and RX for ACK until receiving the ACK.

4.8.1. Periodic Report. The report time of a BoX-MAC-2 device is as follows:

$$T_{resp} = T_B + m * \left(T_{WP} + T_{AW}\right) + T_{DT} + 2 * T_{ACK}, \quad (29)$$

where m is a maximum time to trigger the receiver and $m \leq T_{DC}$.

Therefore, the total report time of n nodes is

$$T_{\text{BOX-MAC-2}(P)} = n * T_{resp} = n * \left(T_B + m * \left(T_{WP} + T_{AW}\right) + T_{DT} + 2 * T_{ACK}\right). \quad (30)$$

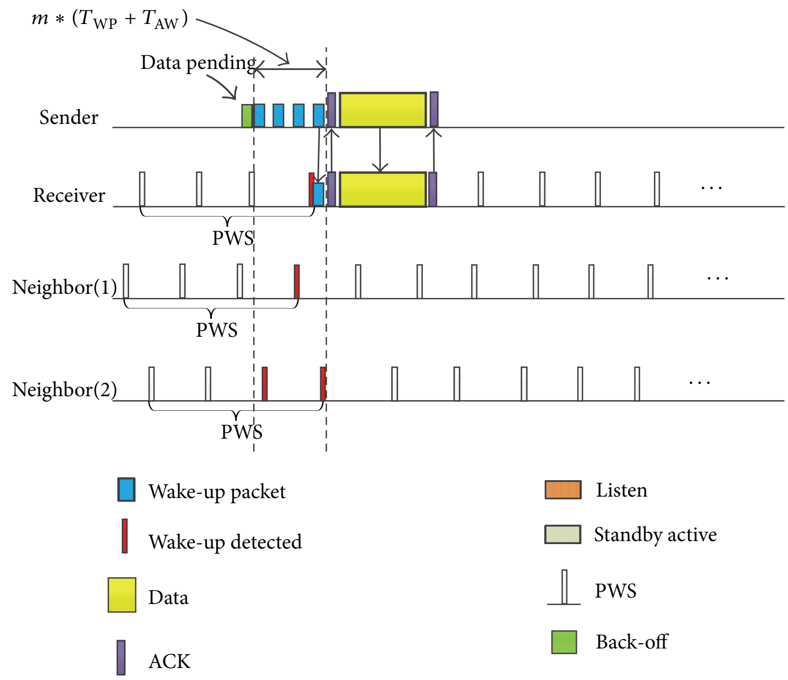


Figure 8: BOX-MAC-2.

4.8.2. Request-Oriented. The required time for a concentrator to transmit its request to devices is

$$T_{\text{req}} = T_B + m * (T_{\text{WP}} + T_{\text{AW}}) + T_{\text{DT}} + 2 * T_{\text{ACK}}. \quad (31)$$

And as in BoX-MAC-1, since data transmission is started only when ACK is received, the request of the concentrator must be transmitted as many times as the number of devices. So the total aggregation time of n nodes per request is

$$\begin{aligned} &T_{\text{BOX-MAC-2}(R)} \\ &\quad = n * (T_{\text{req}} + T_{\text{resp}}) = 2 * n * T_{\text{req}} \\ &\quad = 2 * n * (T_B + m * (T_{\text{WP}} + T_{\text{AW}}) + T_{\text{ACK}} + T_{\text{DT}}). \end{aligned} \quad (32)$$

4.9. MX-MAC. MX-MAC is one of the packet-based LPL protocols and aims to reduce additional handshakes to trigger the receiver. Each MX-MAC device performs PFS, and a sender transmits data after waiting for T_{DP} and then switches to RX to listen for ACK for T_{AW}, as shown in Figure 9.

4.9.1. Periodic Report. The report time of an MX-MAC device is as follows:

$$T_{\text{resp}} = T_B + m * (T_{\text{DP}} + T_{\text{AW}}) + T_{\text{ACK}}, \quad (33)$$

where m is a maximum time to trigger the receiver and $m \le T_{\text{DC}}$.

Therefore, the total report time of n nodes is

$$\begin{aligned} T_{\text{MX-MAC}(P)} &= n * T_{\text{resp}} \\ &= n * (T_B + m * (T_{\text{DP}} + T_{\text{AW}}) + T_{\text{ACK}}). \end{aligned} \quad (34)$$

4.9.2. Request-Oriented. The required time for a concentrator to transmit its request to devices is

$$T_{\text{req}} = T_B + m * (T_{\text{DP}} + T_{\text{AW}}) + T_{\text{ACK}}. \quad (35)$$

T_{req} is equal to T_{resp}, and each request is paired with each response. So the total aggregation time of n nodes per request is

$$\begin{aligned} T_{\text{MX-MAC}(R)} &= n * (T_{\text{req}} + T_{\text{resp}}) \\ &= 2 * n * T_{\text{req}} \\ &= 2 * n * (T_B + m * (T_{\text{DP}} + T_{\text{AW}}) + T_{\text{ACK}}). \end{aligned} \quad (36)$$

4.10. A-MAC. A-MAC is one of the receiver-initiated LPL protocols, like RI-MAC. However, it enhances data reliability by adding early ACK transmission before data transmission. A sender goes into RX to wait and listen for a beacon from the receiver. Upon reception of the corresponding beacon, the sender transmits an early ACK to notify the receiver that data transmission follows, as shown in Figure 10.

4.10.1. Periodic Report. The report time of an MX-MAC device is as follows:

$$T_{\text{resp}} = T_{\text{ACK}} + T_{\text{DT}} + 2 * T_{\text{BS}}. \quad (37)$$

Therefore, the total report time of n nodes is

$$T_{\text{A-MAC}(P)} = n * T_{\text{resp}} = n * (T_{\text{ACK}} + T_{\text{DT}} + 2 * T_{\text{BS}}). \quad (38)$$

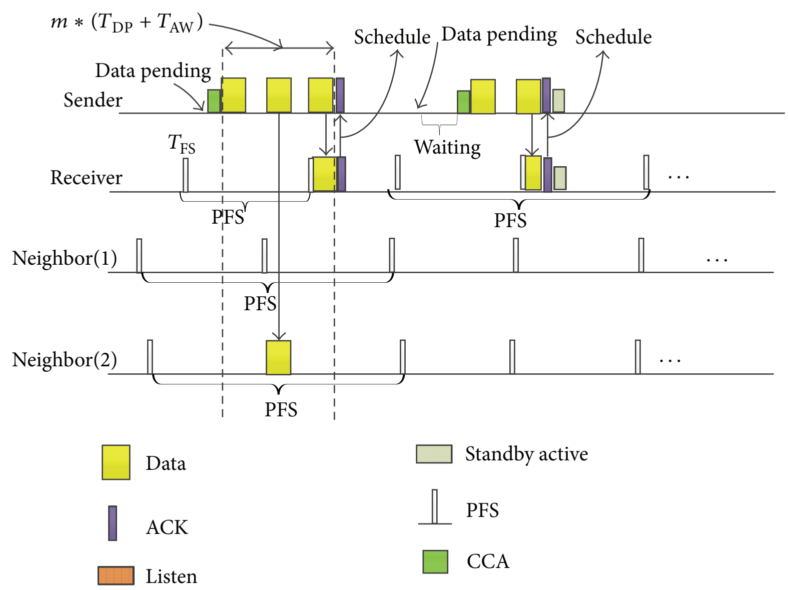


Figure 9: MX-MAC.

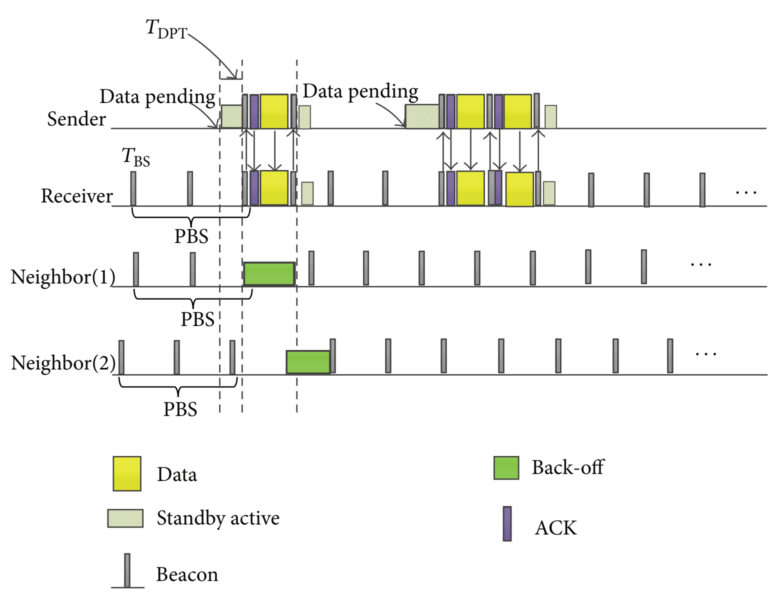


Figure 10: A-MAC.

4.10.2. Request-Oriented. The required time for a concentrator to transmit its request to devices is

$$T_{\text{req}} = T_{\text{ACK}} + T_{\text{DT}} + 2 * T_{\text{BS}}. \quad (39)$$

T_{req} is equal to T_{resp}, and each request is paired with each response. So the total aggregation time of n nodes per request is

$$\begin{aligned} T_{\text{A-MAC}(R)} &= n * \left(T_{\text{req}} + T_{\text{resp}}\right) \\ &= 2 * n * T_{\text{req}} \\ &= 2 * n * \left(T_{\text{ACK}} + T_{\text{DT}} + 2 * T_{\text{BS}}\right). \end{aligned} \quad (40)$$

5. Summary and Concluding Remarks

Based on the taxonomy presented in Section 2 and the numerical analysis presented in Section 4, Table 3 presents a summary of the taxonomy and evaluation results regarding data aggregation time and energy consumption in terms of M2M communication models. In addition, protocol complexity is evaluated in terms of time synchronization requirements, memory usage, and ability to be implemented with an off-the-shelf RF modem and MCU.

First, in terms of data aggregation time, without regarding wake-up source type, while adaptive and schedule-based protocols such as WISE-MAC and MX-MAC show fast data aggregation time, preamble-based or receiver-initiated

Table 3: Summary of taxonomy and evaluations.

Protocol	Wake-up		Initiation		Adaptivity		Schedule		Data aggregation time		Energy consumption			Protocol complexity
	Preamble	Packet	Receiver	Source	Adaptive	Deterministic	Scheduled	Nonschedule	Periodic report	Request-oriented	Concentrator	Device	Network	
B-MAC	√			√		√		√	○	○	●	●	●	○
WISE-MAC	√			√	√		√			●	●	●	●	
X-MAC	√			√	√			√			●	●	●	●
SPECK-MAC(B)		√		√		√	√		○					
SPECK-MAC(D)		√		√		√		√			○		○	○
RI-MAC		√	√			√		√		○				○
BOX-MAC(1)		√		√		√		√	○		○	○	○	○
BOX-MAC(2)		√		√	√			√		○	○	○	○	
MX-MAC		√		√	√		√		●			○	○	
A-MAC		√	√			√		√		○				○

●: A : B : C ○: D.

protocols like B-MAC, RI-MAC, and A-MAC present long aggregation times. This is because adaptive LPL protocols are capable of coping with a receiver's reaction through feedback during wake-up duration, compared with deterministic protocols utilizing fixed-size wake-up duration without regard to the receiver's reaction.

In terms of energy efficiency, while preamble-based protocols, such as B-MAC, WISE-MAC, and X-MAC present superior energy efficiency, data packet-based LPL protocols like BoX-MAC and MX-MAC present high energy consumption. Since preamble detection duration is considerably shorter than the data reception duration, the preamble-based protocols can operate with a very short duty cycle.

In terms of protocol complexity, deterministic or receiver-initiated protocols have relatively low complexity, whereas adaptive and schedule-based protocols, such as WISE-MAC, SpeckMAC-B, and MX-MAC, have high complexity because they require tight time synchronization and management for neighbors' PPS times. In addition, X-MAC (which transmits a short preamble in which ID information is contained) is not possible to implement with an off-the-shelf RF modem.

Lastly, we expect the summarized taxonomy will provide a useful guideline for understanding the specific features of LPL protocols and for designing a new M2M network.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) and funded by the Ministry of Education, Science and Technology (2012R1A1A2041271).

References

[1] H. Mitsui, H. Kambe, and H. Koizumi, "Student experiments for learning basic M2M technologies by implementing sensor network systems," in *Proceedings of the 9th International Conference on Information Technology Based Higher Education and Training (ITHET '10)*, pp. 268–275, Cappadocia, Turkey, April-May 2010.

[2] P. Huang, L. Xiao, S. Soltani, M. W. Mutka, and N. Xi, "The evolution of MAC protocols in wireless sensor networks: a survey," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 101–120, 2013.

[3] J. Liu and S. H. Chung, "An efficient load balancing scheme for multi-gateways in wireless mesh networks," *Journal of Information Processing Systems*, vol. 9, no. 3, pp. 365–378, 2013.

[4] A. Sinha and D. K. Lobiyal, "Performance evaluation of data aggregation for cluster-based wireless sensor network," *Human-Centric Computing and Information Sciences*, vol. 3, no. 13, pp. 1–17, 2013.

[5] M. Yoon, Y. K. Kim, and J. W. Chang, "An energy-efficient routing protocol using message success rate in wireless sensor networks," *Journal of Convergence*, vol. 4, no. 1, 2013.

[6] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys '04)*, pp. 95–107, New York, NY, USA, November 2004.

[7] A. El-Hoiydi and J. Decotignie, "WiseMAC: an ultra low power MAC protocol for the downlink of infrastructure wireless sensor networks," in *Proceedings of the 9th International Symposium on Computers and Communications (ISCC '04)*, pp. 244–251, June-July 2004.

[8] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems (SenSys '06)*, pp. 307–320, New York, NY, USA, November 2006.

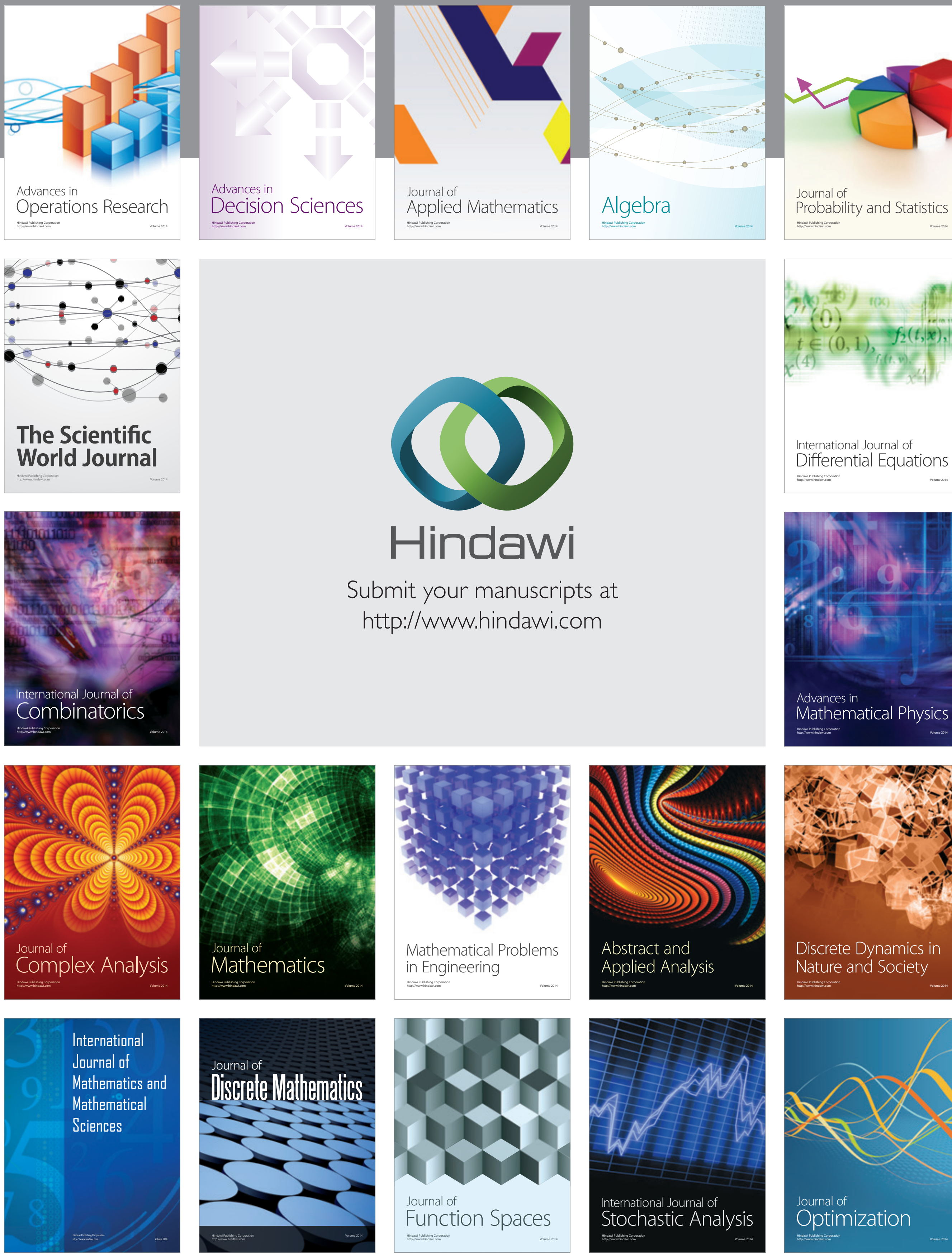
[9] K.-J. Wong and D. Arvind, "SpeckMAC: low-power decentralised MAC protocols for low data rate transmissions in specknets," in *Proceedings of the 2nd International Workshop on Multi-hop Ad Hoc Networks: From Theory to Reality (REALMAN '06)*, pp. 71–78, New York, NY, USA, May 2006.

[10] Y. Sun, O. Gurewitz, and D. B. Johnson, "RI-MAC: a receiver-initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks," in *Proceedings of the 6th ACM Conference on Embedded Networked Sensor Systems (SenSys '08)*, pp. 1–14, New York, NY, USA, November 2008.

[11] D. Moss and P. Levis, "BoX-MAC: exploiting physical and link layer boundaries in low-power networking," Tech. Rep. SING-08-00, 2008.

[12] C. J. Merlin and W. B. Heinzelman, "Schedule adaptation of low-power-listening protocols for wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 5, pp. 672–685, 2010.

[13] P. Dutta, S. Dawson-Haggerty, Y. Chen, C. M. Liang, and A. Terzis, "Design and evaluation of a versatile and efficient receiver-initiated link layer for low-power wireless," *ACM Transactions on Sensor Networks*, vol. 8, no. 3, 2012.



Advances in Operations Research
Advances in Decision Sciences
Journal of Applied Mathematics
Algebra
Journal of Probability and Statistics
The Scientific World Journal
International Journal of Differential Equations
Hindawi
Submit your manuscripts at http://www.hindawi.com
International Journal of Combinatorics
Advances in Mathematical Physics
Journal of Complex Analysis
Journal of Mathematics
Mathematical Problems in Engineering
Abstract and Applied Analysis
Discrete Dynamics in Nature and Society
International Journal of Mathematics and Mathematical Sciences
Journal of Discrete Mathematics
Journal of Function Spaces
International Journal of Stochastic Analysis
Journal of Optimization