A Real-Time and Efficient MAC Protocol for Smart Grid Wireless Communications

Qiang Liu, 1 Danqi Chen, 2 Fangping Gao, 2 and Guoli Pang 2

1 Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China
2 Department of Disaster Information Engineering, Institute of Disaster Prevention, Sanhe, Hebei 065201, China

Correspondence should be addressed to Fangping Gao; gaofangping@cidp.edu.cn

Received 2 March 2014; Revised 4 June 2014; Accepted 11 June 2014; Published 26 June 2014

Copyright © 2014 Qiang Liu et al. 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.

Communication and information technologies play crucial roles in the smart grid system. Wireless communications offer many unique features to utilities. Real-time capability and high efficiency under heavy load are vital capabilities for smart grid wireless communications. However, the existing medium access control (MAC) protocols for low-rate and short-range wireless transfer in the literature mainly aim to achieve the objectives of low energy consumption and self-configuration and rarely address these requirements under heavy traffic intensity. This paper presents a real-time, efficient, and lightweight MAC (RE-MAC) protocol to support smart grid applications, based on priority node polling and a hybrid scheme. The upper bounds of packet delay are determined using the embedded Markov chain method, and the simulation results demonstrate that the protocol can achieve predictable, real-time, and efficient performance.

1. Introduction

Over the last few years, the smart grid has gained wide attention for its potential to address the challenges in the traditional power grid, such as increasing load demands, quickly aging components, domino-effect failures, renewable energy sources, and improving grid security [1, 2]. Communication and information technologies play crucial roles in the smart grid system. Wireless communications offer many unique features to utilities [3, 4]. For example, wireless networks can accommodate condition-monitoring applications due to their ability to work in extreme environmental conditions.

Wireless technologies can be widely applied to the smart grid, including power generation, power delivery, and power utilization [5]. In contrast to wired networks, wireless networks support flexible addition and removal of devices and reduce installation costs. In addition, wireless nodes can be mounted on mobile devices and high-voltage equipment and can fulfill monitoring and control tasks [6]. In detailed scenarios, condition-monitoring of transformers, circuit breakers, and power lines in substations is a very important task, including monitoring temperature, voltage, and current.

Wireless networks make monitoring these parameters more easy and efficient. However, when domino-effect failures occur, a great deal of data emerge, and wireless networks must tolerate the heavy load and transfer information in real time.

However, many challenges must be overcome to use wireless communications in the smart grid, including network performance, suitability, and security [7]. The length of the communication delay is the most important requirement for supporting smart grid applications, and these delay requirements range from 8 ms to more than 1 s. It is difficult to satisfy the delay requirement over a wireless network even for less challenging delay requirements, such as 500 ms for phase measurement [8]. However, the efficiency of wireless communications over shared media is a key metric when processing the heavy and dynamic load in emergencies.

Medium access control (MAC) protocols play a vital role in message delay and communication efficiency characteristics. However, few MAC protocols allow for a real-time smart grid. In general, the MAC protocols of wireless networks for low-rate and short-distance applications consider the energy efficiency as an important aspect. However, higher energy efficiency typically results in a noticeable message delay.
MAC protocols for low-rate and short-distance applications can be categorized into contention-based, schedule-based, and hybrid schemes. In contention-based MAC protocols, nodes compete to acquire the channel for random access, which may result in unpredictable delays arising from collisions. To mitigate unpredictable delays, the RT-MAC protocol maximizes spatial channel reuse by avoiding the false blocking problem of RTS/CTS exchange within one source and sink node pair [9]; however, this protocol still suffers from the interference of multistream communications. The MaxMAC protocol utilizes additional wake-ups to achieve a low delay and high throughput according to the rate of incoming packets [10]. ENCO imposes an approximate value of the optimum contention window size to minimize the channel access delay [11]. QoS-MAC for IEEE 802.15.4 implements QoS support based on the IEEE 802.15.4 unslotted carrier sense multiple access with collision avoidance (CSMA/CA) scheme by utilizing differentiated service for data traffic with different priorities. The QoS-MAC protocol is designed for smart grid distribution monitoring [12]. DRX is another MAC protocol for smart grid applications, and it is based on delay-estimation and data-prioritization steps that are performed by the application layer, in addition to the MAC layer parameters responding to the delay requirements of the smart grid application and the network condition [13]. Although many protocols based on contention may mitigate delays, it remains difficult to eliminate the effect caused by collisions.

Currently, schedule-based MAC protocols generally support deterministic delays. WRT-Ring is a distributed real-time MAC protocol and operates in the slotted virtual ring network [14]. Based on a control signal that circulates into the virtual ring and CDMA, WRT-Ring supports real-time and generic applications. Because the control signal distributes while traveling, addressing the urgent alarm transmission is complicated. Point coordination function (PCF) is one of the two medium access mechanisms in IEEE 802.11, and PCF is a polling scheme that provides a shorter delay in medium access applications than the distributed coordination function (DCF) mechanism, even under heavy loads [15]. Time-division multiple access (TDMA) is an important schedule-based MAC protocol. A tree-based TDMA protocol, which is based on a tree topology and the TDMA scheme [16], has been designed for home area networks in the smart grid; however, TDMA-based protocols lack the flexibility to respond to fluctuations in the traffic load.

Hybrid schemes are developed to overcome the drawbacks of a single scheme by combining multiple schemes. IEEE 802.15.4 works as a hybrid of CSMA/CA and TDMA when using a superframe structure and guaranteed time slot. Because of the limited number of available slots and congestion of the contention access period under intensive data exchanges, this scheme does not appear to be applicable for high-performance, time-critical smart grid applications [17]. WirelessHART is also based on a CSMA/CA and TDMA-based hybrid scheme regarded as a paradigm shifter in the process control industry [18]. WirelessHART satisfies the strict timing requirements and high security concerns for industrial control using time synchronization, channel hopping, and mesh networking. However, CSMA/CA and TDMA suffer from low efficiency under heavy loads and load changes, respectively. EQ-MAC provides QoS support by combining a hybrid medium access scheme with service differentiation for cluster-based single-hop sensor networks [19]; however, it still suffers from congestion because it uses contention-based medium access for control messages.

Applications for smart grid wireless communications, such as condition monitoring, must exhibit high real-time capability, high efficiency under heavy loads, and low complexity. Furthermore, the gateway nodes of such applications must have an unlimited power supply. We study the wireless medium access in this scenario and present a real-time, efficient, and lightweight MAC protocol: RE-MAC.

2. Lightweight, Real-Time Hybrid Scheme

A gateway node must communicate with several device nodes (the sensor and actuator nodes) for condition-monitoring applications in the smart grid. The sensor nodes send regular and urgent data to the gateway node, which then transmits control commands to the actuator nodes. These applications require real-time ability, even under heavy loads, and the gateway node is powered by an unlimited power supply. Tailored to this type of scenario, the lightweight polling scheme, priority node scheme, and hybrid MAC method form a promising hybrid scheme.

RE-MAC is a hybrid scheme of the polling scheme and TDMA. The polling scheme provides high efficiency even under high traffic, and TDMA is used to transmit emergency data during the sleeping period. First, the nodes are divided into clusters. Each cluster has one gateway node and several sensor and actuator nodes in a star topology. The gateway node has an unlimited power supply; thus, the energy consumption of wireless communications can be ignored for the gateway node. The nodes in a cluster utilize the hybrid scheme to share the channel, as shown in Figure 1. Considering the importance of the control commands and configuration data transmitted by the gateway node, the gateway node is prioritized over the other nodes and transmits data first during each polling period. Figure 2 shows a flow chart of the RE-MAC protocol. Initially, all of the device nodes turn on their radios to listen, and the gateway node begins to transmit its data, if any exist. Then, the gateway node sends a polling
All of device nodes turn on to listen.

Gateway node has data to transmit.

Gateway node sends a polling packet to device node $i$.

Is this device node $i$?

Yes

Device node $i$ sends the reply for the polling packet.

Device node $i$ has data to transmit.

Device node $i$ transmits the data.

Device node $i$ goes to sleep.

The device node sleeps until the slot for sending urgent device data for this node is available.

The device node transmits urgent device data if any exists.

The device node sleeps until the polling for next device node and $i = i + 1$.

No

Gateway node transmits the data.

Other device nodes receive the reply of device node $i$.

Device node $i$ has data to transmit.

Device node $i$ transmits the data.

Device node $i$ goes to sleep.

Other device nodes receive the reply of device node $i$.

Is this device node $i$?

Figure 2: Flow chart of the RE-MAC protocol.

Algorithm 1: Gateway node processing of RE-MAC.

(1) $i = 0$; $N =$ Number of device nodes;

(2) If has data to transmit then

(3) Transmit the data;

(4) Send a polling packet containing the length of the sleeping duration $s_i$ to device node $i$;

(5) Receive the reply packet of device node $i$;

(6) If the reply packet indicates device node $i$ has data to transmit then

(7) Receive the data of device node $i$;

(8) Begin the sleeping duration;

(9) Receive urgent data in the slot for device node $i$;

(10) Receive urgent data in the slot for device node $2$;

(11) ...;

(12) Receive urgent data in the slot for device node $N$;

(13) $i = (i + 1)\%N$;

(14) Goto Step 2;

Algorithm 2: Device node processing of RE-MAC.

(1) Turn on the radio to listen;

(2) Receive the polling packet with the sleeping duration $s_i$;

(3) If the polling packet is for this device node then

(4) Send the reply for the polling packet;

(5) If this device node has data to transmit then

(6) Transmit the data;

(7) Else

(8) Wait to receive the reply for the polling packet;

(9) Get the data length $l_i$ from the reply packet;

(10) Set the length of sleeping for this node with $s_i$ and $l_i$;

(11) Go to sleep;

(12) If this is slot for urgent data of this node then

(13) Transmit urgent data;

(14) End of the sleeping duration;

(15) Goto Step 1;

Each device node owns two queues, one for regular device data and the other for urgent device data. When a device node is polled to transmit data, it will send regular device data. During the sleeping period, the device node can wake up and send urgent device data to the active gateway by the TDMA method as shown in Figures 1 and 2. Each device node has one slot for transmitting urgent device data in each sleeping period, and these slots follow the duration of regular device data one by one, as shown in Figure 1. The device node sleeps until the slot for sending urgent device data for this node is available, and then that device node sends the urgent data, if any exist, and goes to sleep. If the device node has no urgent device data, it goes to sleep directly.

Algorithm 1 describes the gateway node processing of RE-MAC in detail. During one polling period, the gateway node will send its data first and then poll device nodes to receive the data; finally, the gateway node will receive the urgent device data by the TDMA method.

Device node processing of RE-MAC is shown as Algorithm 2. The device node will receive the polling packet and obtain the sleeping duration for the polling packet. If this
node is the polled node, it will transmit its data. Otherwise, it will wait to receive the reply for the polling packet from the polled node and obtain the transmitted data length from the reply packet. Finally, the device node will send urgent data by the TDMA method.

We illustrate the RE-MAC protocol with an example. There are 10 device nodes from DN 1 to DN 10 and one gateway node, GN, in a cluster. At the beginning, all 10 of the device nodes turn on their radios to listen. GN transmits its data if any exists. Then, GN sends a polling packet containing the length of the sleeping duration $s_i$ to DN 1. DN 1 receives the polling packet and returns a reply with the length of data $l_i$ to be sent. If DN 1 has data to be sent, it transmits the data to GN at once. Then, DN 1 goes to sleep for $s_i$. The other device nodes receive the polling packet to DN 1 and the reply packet of DN 1, set the sleeping period referring to $s_i$ and $l_i$, and then go to sleep. During the sleeping period, DN 1 to DN 10 send urgent device data to GN one by one using TDMA slots at the beginning of the sleeping period. At the end of the sleeping period, all of the device nodes wake up to listen. GN begins polling DN 2, with the same protocol as for polling DN 1. After polling DN 10, GN polls DN 1 to access the wireless channel.

In summary, there are three types of data for this RE-MAC hybrid scheme: gateway data, urgent device data, and regular device data. We utilize the priority node scheme and TDMA scheme in each sleeping period to provide real-time processing capability for the gateway node and device nodes, respectively. RE-MAC is also highly efficient for device nodes. The device nodes work under a polling scheme and TDMA; therefore, device nodes using the RE-MAC protocol gain better transmission efficiency and energy efficiency than CSMA/CA protocols, which suffer contention, especially under a heavy load. In addition, device nodes will go to sleep when there is no work to do within the polling duration, which will greatly reduce the energy consumption of device nodes and extend the life time of device nodes.

### 3. Theoretical Analysis

We analyze the performance of priority node polling for RE-MAC and then verify the theoretical results and investigate the entire protocol using simulations. We assume that a cluster includes $N + 1$ nodes, where one node is the gateway node and the other $N$ nodes are device (sensor and actuator) nodes. The gateway node polls device nodes in cyclic order 1, 2, ..., $N$, such that node $N$ is followed by node 1. Each node has two queues of the first-come-first-serve type, and the capacity of each queue is unlimited. The service of all nodes is of the gating type, namely, the node sends only those packets that were waiting in the queue when the transmission began. For the device nodes, the arriving packets of node $i$ ($i = 1, 2, ..., N$) follow a Poisson process with an arrival rate $\lambda_i$; the service time for each node to transmit packets is independent of the other nodes with a general distribution, and the distribution function is $H_i(x)$. $G$ expresses the queue of the gateway node; the input is a Poisson process with an arrival rate $\lambda_G$; the service time has a general distribution; and the distribution function is $H_G(x)$. From node $i$ to node $i + 1$, the polling time has a general distribution and the distribution function is $\mu_i(x)$; the sleeping time has a negative exponential distribution; and the distribution function is $s_i(x)$. $\nu_1(t)$ and $\nu_2(t)$ are the number of packets of node $i$ and gateway arriving within $t$ duration, respectively. In the equilibrium case, when the polling arrives at node $i$, the probability of node $k (k = 1, 2, ..., N, G)$ having $j_k$ packets waiting is $g_i(j_1, j_2, \ldots, j_N, j_G)$. We define the generation function as follows:

$$G_i(x_1, x_2, \ldots, x_N, x_G) = \sum_{j_1,j_2,\ldots,j_N,j_G=0}^{\infty} x_1^{j_1} x_2^{j_2} \cdots x_N^{j_N} x_G^{j_G} g_i(j_1, j_2, \ldots, j_N, j_G).$$

\hspace{1cm} (1)

In the device nodes, assume that the instant of the start of service at one of the device nodes is $t_1, t_2, \ldots$, as shown in Figure 1; then $t_1 < t_2 < \cdots < T$ is defined as the sequence $\{t_n\}$. Define random variables $e_n(i)$ as the number of packets in node $i$ at instant $t_n$ and $\delta_n$ as the node identifier of the polling arriving at instant $t_n$. Thus, at instant $t_n$, the system state is $(\delta_n, s_1, e_n(1), s_2, e_n(2), \ldots, s_N, e_n(N), e_n(G))$ and the state space of the system is $I = \{(i, k_1, k_2, \ldots, k_N, k_G) : i = 1, 2, \ldots, N, k_1 = 0, 1, 2, \ldots, j = 1, 2, \ldots, N, G\}$. Thus, the transition probability of the state $(\delta_n, s_1, e_n(1), s_2, e_n(2), \ldots, s_N, e_n(N), e_n(G))$ becomes an aperiodic, irreducible finite Markov chain. Therefore, we have the following limiting probability:

$$\lim_{n \to \infty} \Pr \{\delta_n = i, e_n(k) = j_k : k = 1, 2, \ldots, N, G\} = g_i(j_1, j_2, \ldots, j_N, j_G).$$

\hspace{1cm} (2)

The necessary and sufficient condition for existence of the equilibrium state is given by

$$\sum_{i=1}^{N} \rho_i + \rho_G < 1, \quad \rho_i = \lambda_i h_i, \quad \rho_G = \lambda_G h_G,$$

\hspace{1cm} (3)

where $h_i$ and $h_G$ are the mean service time of a packet in the device node $i$ and in the gateway node, respectively. $\rho_i$ and $\rho_G$ are the traffic intensity of node $i$ and gateway, respectively. The above equation yields the equilibrium condition that the total traffic intensity must be less than one.

In (2), $g_i(j_1, j_2, \ldots, j_N, j_G)$ expresses the probability that $j_k$ packets are waiting at device node $k (k = 1, 2, \ldots, N)$ when the gateway node polls device node $i$ in equilibrium state, namely, considering the state space $I = \{(i, k_1, k_2, \ldots, k_N, k_G) : i = 1, 2, \ldots, N, k_1 = 0, 1, 2, \ldots, j = 1, 2, \ldots, N, G\}$. In addition, $g_i(j_1, j_2, \ldots, j_N, j_G)$ is the transition probability from state $(i - 1, e_i(1), e_{i-1}(2), \ldots, e_{i-1}(N), e_{i-1}(G))$ to state $(i, e_i(1), e_i(2), \ldots, e_i(N), e_i(G))$. There are two cases for the state transition, for all $j_k \geq 0$, the first case is $g_i(j_1, j_2, \ldots, j_N, j_G) > 0 (i = 1, 2, \ldots, N)$ and the second case is $g_i(j_1, j_2, \ldots, j_N, j_G) = 0 (i = 1, 2, \ldots, N)$. In the first case, the Markov chain is ergodic and the system can be assumed to be in statistical equilibrium. In the second case, all the states are either transient or recurrent null.
We define $F_G(x)$ as the service time of the gateway node data at any one instant $t_i$, and there are $x$ packets in the gateway node. During the period from $t_i$ to $t_{i+1}$, the service time for the gateway node is $F_G'(x)$. Therefore, at instant $t_{i+1}$, the gateway node and device nodes own the following numbers of packets:

$$
\epsilon_{i+1} (n) = v_G (\mu_i (n) + \tau_i (n) + s_i (n)) + v_G (F_G (\epsilon_i (n)))
$$

$$
\epsilon_{i+1} (i) = v_i (\tau_i (n) + s_i (n)) + v_G (F_G (\epsilon_i (n)))
$$

$$
\epsilon_{i+1} (j) = \epsilon_i (j) + v_j (\mu_j (n) + \tau_j (n) + s_j (n)) + v_j (F_G (\epsilon_j (n)))
$$

(4)

where $\mu_i(n)$, $\tau_i(n)$, and $s_i(n)$ are the polling time, service time, and sleeping time for device node $i$ from $t_i$ to $t_{i+1}$, respectively. When the system attains the state of statistical equilibrium, the probability distribution generation function of $Pr(i + 1, \epsilon_{i+1}(1), \epsilon_{i+1}(2), \ldots, \epsilon_{i+1}(N), \epsilon_{i+1}(1))$ is

$$
G_{i+1} (x_1, x_2, \ldots, x_N, x_G) = E \left[ \sum_{k=1}^{N} x_k^{\epsilon_k (1)} \cdot x_{\epsilon_k (G)}^{\epsilon_k (G)} \right]
$$

$$
= U_i^* (A) \cdot S_i^* (B) \cdot G_i
$$

(5)

$$
\times (x_1, x_2, \ldots, H_i^* (B), \ldots, x_N, H_G^* (A)).
$$

In the above equation, $U_i^* (s)$ and $S_i^* (s)$ ($i = 1, 2, \ldots, N$) are the Laplace-Stieltjes transform of the polling time and sleeping time probability distribution, respectively, when the gateway node polls from device node $i$ to $i + 1$; $H_i^* (s)$ is the busy period probability distribution Laplace-Stieltjes transform of the device node queue with a Poisson input; and $H_G^* (s)$ is the busy period probability distribution Laplace-Stieltjes transform of the gateway node queue with a Poisson input. $A$ and $B$ are as follows:

$$
A = \sum_{k \neq i} \lambda_k (1 - x_k) + \lambda_G (1 - x_G),
$$

$$
B = \sum_{k=1}^{N} \lambda_k (1 - x_k) + \lambda_G (1 - x_G).
$$

(6)

Thus, $G_i(x_1, x_2, \ldots, x_N, x_G)$ is described in functional equation form using the recursive formula of (5); however, the explicit representation of $G_i(x_1, x_2, \ldots, x_N, x_G)$ cannot be derived. The mean queue length in node $j$ at the start-of-service instant at node $i$ is denoted by $g_j(i)$; then, we have

$$
g_j(i) = \lim_{x_1, x_2, \ldots, x_N \to 1} \frac{\partial G_i (x_1, x_2, \ldots, x_N, x_G)}{\partial x_j} / G_i (1, 1, \ldots, 1, 1).
$$

(7)

From (5), we obtain $G_i(1, 1, \ldots, 1) = k$ as a constant when $i = 1, 2, \ldots, N$. Therefore, for (5), we can perform differential operations with respect to $x_j$, $x_i$, and $x_G$. When $x_i \to 1$ ($l = 1, 2, \ldots, N, G$), the following expressions are obtained:

$$
g_{i+1} (j) = \lambda \cdot \mu_i + \lambda \cdot \rho_i + g_i (j) + \lambda \cdot h_i \cdot g_i (i) + \lambda \cdot h_G \cdot g_i (G),
$$

$$
g_{i+1} (i) = \lambda \cdot s_i + \lambda \cdot h_i \cdot g_i (i),
$$

$$
g_{i+1} (G) = \lambda \cdot C_G \cdot \mu_G + \lambda \cdot C_G \cdot s_i + \lambda \cdot h_i \cdot g_i (i) + \lambda \cdot h_G \cdot g_i (G),
$$

(8)

where $\mu_i$ and $s_i$ are the mean polling time and mean sleeping time for device node $i$, respectively. With the third equation in equation set (8), we obtain the expression for $g_j(G)$ because $g_{i+1}(G) = g_i(G)$ in the equilibrium state. Then, we substitute the expression for $g_j(G)$ into the first equation of equation set (8) and accumulate the first equation with respect to $i = j + 1, \ldots, i-1, i$ and the second equation with respect to $i = j$. When device nodes are symmetric, namely, $\lambda_i = \lambda, \mu_i = \mu, \rho_i = \rho, g_i(G) = g(G)$, and $g_i(i) = g$, we obtain the mean queue length of the device nodes as follows:

$$
g = \frac{N \lambda \mu + N \lambda s - \lambda \rho G - \lambda \mu}{1 - \rho G - N \rho G}.
$$

(9)

The mean queue length of gateway node is

$$
g (G) = \frac{(1 - \rho_G - \rho + \rho \rho_G) \cdot \lambda_G \mu + (1 - \rho_G) \cdot \lambda_G s}{(1 - \rho_G) \cdot (1 - \rho_G - N \rho + \rho \rho_G)}.
$$

(10)

Then, we can obtain the mean cyclic time of polling as

$$
T_{cyclic} = N \cdot \left[ \mu + s + h_G + h_C g (G) \right]
$$

$$
= \frac{\mu + s + \mu \rho}{1 - \rho_G - N \rho + \rho \rho_G} \cdot N.
$$

(11)

We ignore the radio propagation delay here because the nodes are relatively close to each other. A device node will wait for one cycle time to transmit a packet in the worst case, and thus, the mean upper bound of the device node packet delay is given by

$$
T_{upper\_bound\_dn} = \frac{\mu + s + \mu \rho}{1 - \rho_G - N \rho + \rho \rho_G} \cdot N + h.
$$

(12)

The gateway node has a higher priority than the device node, and it will have the opportunity to send packets at every service start instant. Therefore, the mean upper bound of the gateway node data packet delay is given by

$$
T_{upper\_bound\_gn} = \frac{\mu + s + \mu \rho}{1 - \rho_G - N \rho + \rho \rho_G} + h_G.
$$

(13)

4. Simulation Results

In this section, we present simulation results to verify the theoretical results and illustrate the advantage of the REMAC algorithm compared to wirelessHART and DRYX.

The mean upper bounds of regular device data packet delay and gateway data packet delay are obtained from (11)–(13). To verify these analytical results, we established
the RE-MAC simulation scenario based on ns-2 as follows. Systems with five or 10 device nodes were applied. The total traffic intensity was set to the range of 0.05 to 0.80, and the step was 0.05. The data inputs of the device nodes and gateway node have symmetrical distributions, the data packet lengths are all 400 bits, the polling packet length is 32 bits, the link rate is 250 Kbps, the mean value of the sleeping time is 0.2 s, and the simulation running time is 200 s. Figure 3 illustrates that the simulation results of the mean cyclic time agree with the theoretical analysis, thus verifying that the theoretical analysis is correct. The simulation results indicate that at least 99.0063% of the packet delays satisfied the mean upper bounds of the packet delay.

We developed a wirelessHART (wHART) simulation scenario and added the urgent device data to RE-MAC and wirelessHART. The regular device data and the urgent device data utilize the contention access period slots and contention-free period slots, respectively, and the gateway data use the contention-free period for the wirelessHART simulation. In addition, we developed a DRX simulation scenario and set the delay threshold to 0.4 second for the regular device data and the gateway data. Figure 4 illustrates that the gateway data delays are all less than 200 ms when the total traffic intensity is less than 0.35, and RE-MAC, DRX, and wirelessHART all present good real-time performance. However, the gateway data delay of DRX and wirelessHART increases rapidly when the total traffic intensity is greater than 0.4 and 0.7, respectively. Figure 5 shows that the urgent device data delays are all less than 200 ms when the total traffic intensity is less than 0.35, and RE-MAC and wirelessHART both present good real-time performance. With the growth of the total traffic intensity, the urgent device data delays of RE-MAC show a slow rise; however, the data delays of DRX increase quickly when the total traffic intensity is greater than 0.4.

According to the regular device data delay results shown in Figure 6, the RE-MAC data delay slowly increases with total traffic intensity growth, and the data delay of DRX and wirelessHART rapidly rises when the total traffic intensity is more than 0.4 and 0.5 for $N = 5$ and $N = 10$, respectively. RE-MAC is based on a polling scheme, and the polling scheme could provide this traffic-adaptive ability. In addition, using the priority node method and TDMA, RE-MAC exhibits better efficiency than DRX and wirelessHART under heavy load conditions.
5. Conclusion

This paper presented a real-time, efficient, and lightweight MAC protocol, RE-MAC, based on priority node polling and a hybrid scheme. The mean upper bounds of the gateway data packet delay and regular device data packet delay were determined. The gateway data and regular device data present outstanding real-time performance, even under heavy traffic. Additionally, the urgent messages of the device nodes can be transmitted in real time using TDMA scheduling.

Disclosure

This paper is original and has been written by the stated authors who are all aware of its content and approve its submission. It is not under consideration for publication elsewhere.

Conflict of Interests

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

Acknowledgments

The authors would like to thank the anonymous reviewers for several valuable comments and suggestions, which enabled the authors to significantly improve this paper. This work was supported in part by the Science and Technology Program Project of Hebei Province under Grant no. 12270325 and the Special Fund of Fundamental Scientific Research Business Expense for Higher School of Central Government (projects for creation teams) under Grant no. ZY20120104.

References

[1] X. Fang, S. Misra, G. Xue, and D. Yang, “Smart grid—the new and improved power grid: a survey,” IEEE Communications Surveys and Tutorials, vol. 14, no. 4, pp. 944–980, 2012.
[2] V. C. Gungor, D. Sahin, T. Kocak et al., “A survey on smart grid potential applications and communication requirements,” IEEE Transactions on Industrial Informatics, vol. 9, no. 1, pp. 28–42, 2013.
[3] H. Farhangi, “The path of the smart grid,” IEEE Power and Energy Magazine, vol. 8, no. 1, pp. 18–28, 2010.
[4] Z. Fan, P. Kulkarni, S. Gormus et al., “Smart grid communications: overview of research challenges, solutions, and standardization activities,” IEEE Communications Surveys and Tutorials, vol. 15, no. 1, pp. 21–38, 2013.
[5] V. C. Gungor, D. Sahin, T. Kocak et al., “A Survey on Smart Grid Potential Applications and Communication Requirements,” IEEE Transactions on Industrial Informatics, vol. 9, no. 1, pp. 28–42, 2013.
[6] Y. Liu, “Wireless sensor network applications in smart grid: recent trends and challenges,” International Journal of Distributed Sensor Networks, vol. 2012, Article ID 492819, 8 pages, 2012.
[7] C. Gentile, D. Griffith, and M. Souryal, “Wireless network deployment in the smart grid: design and evaluation issues,” IEEE Network, vol. 26, no. 6, pp. 48–53, 2012.
[8] K. C. Budka, J. G. Deshpande, T. L. Douni, M. Madden, and T. Mew, “Communication network architecture and design principles for smart grids,” Bell Labs Technical Journal, vol. 15, no. 2, pp. 205–228, 2010.
[9] B. K. Singh and K. E. Tepe, “Feedback based real-time MAC (RT-MAC) protocol for wireless sensor networks,” in Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM ’09), pp. 1–6, Honolulu, Hawaii, USA, December 2009.
[10] P. Hurni and T. Braun, “MaxMAC: a maximally traffic-adaptive MAC protocol for wireless sensor networks,” in Proceedings of the European Conference on Wireless Sensor Networks (EWSN ’10), pp. 289–305, Coimbra, Portugal, February 2010.
[11] I. Demirkol and C. Ersoy, “Energy and delay optimized contention for wireless sensor networks,” Computer Networks, vol. 53, no. 12, pp. 2106–2119, 2009.
[12] W. Sun, X. J. Yuan, J. P. Wang, D. Han, and C. W. Zhang, “Quality of service networking for smart grid distribution monitoring,” in Proceedings of the IEEE International Conference on Smart Grid Communications (SmartGridComm ’10), pp. 373–378, 2010.
[13] I. Al-Anbagi, M. Erol-Kantarci, and H. T. Mouftah, “Priority-and delay-aware medium access for wireless sensor networks in the smart grid,” IEEE Systems Journal, vol. 8, no. 2, pp. 608–618, 2013.
[14] L. Bononi, L. Donatiello, and M. Furini, “Real-time traffic in ad-hoc sensor networks,” in Proceedings of the IEEE International Conference on Communications (ICC ’09), pp. 1–5, Dresden, Germany, June 2009.
[15] M. A. R. Siddique and J. Kamruzzaman, “Performance analysis of PCF based WLANs with imperfect channel and failure retries,” in Proceedings of the 53rd IEEE Global Communications Conference (GLOBECOM ’10), pp. 1–6, December 2010.
[16] M. Kim, J. Kim, J. Kim, and Y. Yoo, “Design and implementation of MAC protocol for SmartGrid HAN environment,” in Proceedings of the IEEE 11th International Conference on Computer and Information Technology (CIT ’11), pp. 212–217, Paphos, Cyprus, August-September 2011.

[17] S. Ullo, A. Vaccaro, and G. Velotto, “Performance analysis of IEEE 802.15.4 based sensor networks for smart grids communications,” Journal of Electrical Engineering: Theory and Application, vol. 1, no. 3, pp. 129–134, 2010.

[18] J. Song, S. Han, A. K. Mok et al., “WirelessHART: applying wireless technology in real-time industrial process control,” in Proceedings of the IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS ’08), pp. 377–386, St. Louis, Mo, USA, April 2008.

[19] B. Yahya and J. Ben-Othman, “Energy efficient and QoS aware medium access control for wireless sensor networks,” Concurrency Computation: Practice and Experience, vol. 22, no. 10, pp. 1252–1266, 2010.
