C-MAC: An Energy-Efficient MAC Scheme Using Chinese-Remainder-Theorem for Wireless Sensor Networks

Yuh-Shyan Chen  
Department of Computer Science  
and Information Engineering  
National Taipei University  
Taipei, Taiwan, R.O.C.  
Email: yschen@csie.ntpu.edu.tw

Yun-Wei Lin  
Department of Computer Science  
and Information Engineering  
National Chung Cheng University  
Chia-Yi, Taiwan, R.O.C.  
Email: jyneda@giam.dynu.com

Abstract—This paper investigates a new TDMA-based MAC, called C-MAC (MAC using Chinese remainder theorem), protocol with low power consumption while maintaining low packet latency for large-scale cluster-based WSNs. To offer low power consumption, each node and coordinator to be active and sleep states based on the time-slot schedule of Chinese remainder theorem. To provide low packet latency, our C-MAC protocol provides an adaptive time-slot scheme to distributely and dynamically wake up time slots for the burst data transmission in a duty cycle. Finally, simulation results illustrate our performance achievements to verify that C-MAC performs better than existing TDMA-based MAC protocols, BMA and EMAC, in terms of power consumption and average packet latency.1

Keywords: Wireless sensor networks, MAC, Latency, Energy efficient operation

I. INTRODUCTION

Energy efficiency is the most important design issue for WSNs. To reduce energy consumption of the sensor nodes, many researches investigate the research issues of energy efficiency to design low-power communication protocols for WSNs. There are many researches about media access control (MAC) protocol designed for wireless sensor network. Existing results can be divided into contention-based MAC protocols [3] [7] and TDMA-based MAC protocols [6] [9]. High cost of idle listening and hidden terminal are the main design problems for the contention-based MAC protocols. This is mainly due to the idle listening [10]. In addition, TDMA-based MAC protocols is another main design stream for WSNs. Example of a cluster-based WSN is given in Fig. 1(a), three cluster-based WSNs. Let $CL_i$ denote the coordinator node of $CL_i$. Coordinator node $C_i$ discovers two hop away adjacency coordinator node to create the neighbor cluster table. Let $B_{CL_i} = \{CL_{i+1}, CL_{i+j}, CL_{i+k}\}$ denote the neighbor cluster table of $CL_i$. Let $\alpha_i$ is the number of neighbor clusters in $B_{CL_i}$. Example of a cluster-based WSN is given in Fig. 1(a), three cluster $CL_1$, $CL_2$, and $CL_3$ is existed. $B_{CL_1}$, $B_{CL_2}$, and $B_{CL_3}$ are $B_{CL_1} = \{CL_2, CL_3\}$, $B_{CL_2} = \{CL_1, CL_3\}$, and $B_{CL_3} = \{CL_1, CL_2\}$ respectively, and $\alpha_1 = \alpha_2 = \alpha_3 = 2$. Our work mainly compares with two other existing TDMA-based MAC protocols, BMA [6] and EMAC protocols [9]. The model used in this work is same as the models defined in [2] [4] [6] [8] [9]. A cluster $CL_i$ is composed of FFD (cluster coordinator) and RDF (cluster member) [5].

1This research was supported by the National Science Council of the R.O.C. under grant NSC-95-2221-E-305-008.
In the following, we point out the performance improvements of our scheme both in power consumption and packet latency.

Some notations are defined for the up-link communication. Let $N_i$ denote the total number of sensor node in cluster $C_L_i$. Let $S_{N_i,j}$ be an event request that sensor node $N_i,j$ intends to send data to coordinator node $C_i$ occurred at $x$-th duty cycle in cluster $C_L_i$, where $1 \leq j \leq N_i$. Let $S_{N_i,j}$ denote a time slot, where sensor node $N_i,j$ sends data to coordinator for an event request $S_{N_i,j}$. Let $S_{N_i,j}$ denote a time slot, where coordinator $C_i$ receives data from sensor node $N_i,j$ for request $S_{N_i,j}$. The key idea of C-MAC protocol allows that coordinator node $C_i$ can enter into the PS mode, a novel wake-up scheme is scheduled is this work for purpose of the low power consumption and low packet latency.

Before describing our scheme, we describe the main works of BMA and EMAC protocols [6] [9] as follows. Example is shown in Fig. 1(a) and $N_i = N_{i,2} = N_{i,3}$. The time slot structures of BMA and EMAC are given in Fig. 1(b). In EMAC protocol, radios of coordinator and sensor nodes are turned on in CR (communication request) period and in TC (traffic control) period to transmit/receive control messages. To prevent collision, only one cluster $C_L_i$ can active in a duty cycle. For an active duty cycle, all sensor nodes periodically wake up to listen control message from coordinator during TC period for every time slot. Not all sensor nodes have data to send or receive. Idle listening is occurred and useless control packet is increased. Example is given in Fig. 1(c), using BMA protocol, nodes $N_1, N_2,$ and $N_3$ send data to coordinator $C_1$ on time slots $S_{N_1,1}, S_{N_2,1},$ and $S_{N_3,1}$ at $x$-th duty cycle for requests $S_{N_1,1}^{x-2}, S_{N_2,1}^{x-2},$ and $S_{N_3,1}^{x-2}$. The operation of BMA includes cluster set-up and steady-state phases, where steady-state phase contains contention, data transmission, and idle periods. If a sensor node intends to send data, it sends a short control to coordinator in the contention period. Coordinator broadcasts the schedule for sensor nodes after contention period. If any data arrives after the contention period, a sensor nodes must waits and transmits data until next duty cycle. That is, if any node detects request of data transmission, it buffers the data and transmits it until next duty cycle. Therefore, BMA provides less packet latency than EMAC. Example is given in Fig. 1(c), using BMA protocol, nodes $N_{i,1}, N_{i,2},$ and $N_{i,3}$ send data to coordinator $C_1$ at time slots $S_{N_{i,1}}^{x-2}, S_{N_{i,2}}^{x-2},$ and $S_{N_{i,3}}^{x-2}$ at $(x-1)$-th duty cycle.

Our C-MAC protocol is developed to allow coordinator to some pre-scheduled time slots by Chinese Remainder Theorem (CRT), called C-MAC protocol, for wireless sensor networks.

III. C-MAC PROTOCOL

To effectively reduce the power consumption and packet latency, we present a new medium access control protocol using Chinese Remainder Theorem (CRT), called C-MAC protocol, for wireless sensor networks.
In C-MAC protocol, each sensor node $N_{ij}$ is responsible for periodically wakes up to transmit sensed data to the coordinator node $C_i$ in a cluster $CL_i$, where $1 \leq j \leq N_i$. Let $WT_{P_{ij}}$ denote the wake-up time period of sensor node $N_{ij}$, where $WT_{P_{ij}}=\tilde{S}_{N_{ij}}-\tilde{S}_{N_{ij}}^{-1}$. Each sensor node $N_{ij}$ has different wake-up time period $WT_{P_{ij}}$ to avoid the collision of data transmission. C-MAC protocol performs a wake-up time period schedule to generate distinct wake-up time period $WT_{P_{ij}}$ for each sensor node $N_{ij}$ in a cluster $CL_i$. This wake-up time period schedule is executed by a coordinator node $C_i$ based on a prime and remainder sequences (defined later) from the Chinese Remainder Theorem [1]. Each coordinator node $C_i$ has distinct prime and remainder sequences, each sensor node $N_{ij}$ can choose another remainder sequences to change the time slot $\tilde{S}_{N_{ij}}$. Sensor node $N_{ij}$ can choose another remainder sequences to change the $WT_{P_{ij}}$ for different sensing task. This result can effectively avoid the time slot collision for data transmission. The main goal of the wake-up time period schedule is to generate $WT_{P_{ij}}$ for each sensor node $N_{ij}$. Sensor node $N_{ij}$ wakes up and transmits data to the coordinator node $C_i$ if the sensor node $N_{ij}$ has data transmission event before the wake-up time slot $\tilde{S}_{N_{ij}}$

C-MAC provides two transmission capability, regular data transmission and burst data transmission mechanism. The regular data transmission mechanism is used for periodically transmitting sensed data. In the regular data transmission, sensor node $N_{ij}$ transmits sensed data at time-slot $\tilde{S}_{N_{ij}}$ which is determined by prime and remainder sequences. As shown in Fig. 3(a), sensor node $N_{ij}$ transmits sensed data to coordinator node $C_i$ at a specific time-slot $\tilde{S}_{N_{ij}}$. The burst data transmission mechanism is used for transmitting the burst data. In the burst data transmission mechanism, sensor node $N_{ij}$ transmits burst data at a specific time-slot which is determined by coordinator node $C_i$. Let $B_{N_{ij}}$ be a burst data request that sensor node $N_{ij}$ occurred at $x$-th duty cycle in cluster $CL_i$. Let $B_{N_{ij}}$ denote a time slot, where sensor node $N_{ij}$ sends data to coordinator for a burst request $B_{N_{ij}}$. As shown in Fig. 3(b), a burst data request is happened at $B_{N_{ij}}^{-1}$, and the burst data is transmitted at $\tilde{B}_{N_{ij}}$. Using burst data transmission mechanism, the time delay of this burst data transmission is $T_1$, and using regular data transmission to transmit the burst data, the time delay of burst data transmission is $T_2$, where $T_1 < T_2$, as shown in Fig. 3(b). It is worth to develop the burst data transmission mechanism for the burst data transmission. The regular data transmission and burst data transmission mechanisms of C-MAC are described as follows.

A. Regular Data Transmission Mechanism

In the regular data transmission, C-MAC protocol mainly calculates a duty cycle length $L_i$ for each coordinator node $C_i$ in the cluster $CL_i$, where $L_i$ is the duty cycle length for cluster $CL_i$. Based on calculated $L_i$, the regular data transmission mechanism is executed as follows.

Sensor node $N_{ij}$ has different wake-up time period $WT_{P_{ij}}$ for different sensing task. Due to different sensing task, each sensor nodes $N_{ij}$ waits for different wake-up time period $WT_{P_{ij}}$ and wake up at different wake-up time slot $\tilde{S}_{N_{ij}}$ to transmit data to coordinator node $C_i$ if sensor nodes $N_{ij}$ has buffered data. The wake-up time slot $\tilde{S}_{N_{ij}}$ is assigned by coordinator node $C_i$ for existing TDMA-based protocols. Only sensor node $N_{ij}$ knows its required wake-up time period $WT_{P_{ij}}$ due to different sensing task. To assign the wake-up time slot $\tilde{S}_{N_{ij}}$, the assignment of required wake-up time period $WT_{P_{ij}}$ is not achieved by the coordinator node $C_i$. Therefore, in our C-MAC protocol, each sensor node $N_{ij}$ determines its own the wake-up time slot $\tilde{S}_{N_{ij}}$. The wake-up time slot $\tilde{S}_{N_{ij}}$ is determined by prime and remainder sequences from Chinese Remainder Theorem. When a sensor node $N_{ij}$ registers to the coordinator node $C_i$, the coordinator node $C_i$ informs the sensor node $N_{ij}$ prime and remainder sequences. Sensor node $N_{ij}$ then chooses a remainder sequences to determine the wake-up time slot $\tilde{S}_{N_{ij}}$. Before describing how to determine the $\tilde{S}_{N_{ij}}$ from prime and remainder sequences, the Chinese Remainder Theorem [1] is formally described below.

Theorem 1: [1] Let prime sequence $p = \{p_1, p_2, ..., p_v\}$ denote a pairwise relatively prime integer sequence, where $1 \leq t \leq v$. Let a remainder sequence $r = \{r_1, r_2, ..., r_t\}$ denote the remainder integer sequence form $p$, where $0 \leq r < p_t$. Then

- There exists an integer $\lambda$ such $I = r_0(\text{mod} p_t)$, and
- If $I' = r_t(\text{mod} p_t)$, then $I' \equiv I(\text{mod} p_1 p_2 ... p_v)$

For example, there exists an integer $I = 23$, and the prime sequence $p = \{3, 5, 7\}$ and the remainder sequence $r = \{2, 3, 2\}$, since $23 = 2(\text{mod} 3)$, $23 = 3(\text{mod} 5)$. If another integer $I' = 128$ exists, then $128 = 2(\text{mod} 3)$, $128 = 3(\text{mod} 5)$, $128 = 2(\text{mod} 7)$ and $128 = 23(\text{mod} p_1 p_2 p_3) = 128 = 23(\text{mod} 105)$.

Let $p_t = \{p_{t1}, p_{t2}, ..., p_{ts} \}$, where $p_{ts}$ is a prime and remainder sequences. Sensor node $N_{ij}$ from $r_{set}$, where $r_{ji} \in r_{set}$. For instance, $r_{set1} = \{0, 1, 0, 0\}$, and $r_{set2} = \{1, 0, 1, 1\}$, respectively if prime sequence $p = \{3, 5\}$, $p_2 = 3, 5$, $p_3 = 3, 5$.

The prime sequence $p_t = \{p_{t1}, p_{t2}, ..., p_{ts} \}$ and remainder sequence $r_{ji} = \{r_1, r_2, ..., r_{ts}\}$ are used to calculate the duty cycle length $L_i$ and wake-up time slot $\tilde{S}_{N_{ij}}$, respectively. Let $B_{CL_i}^{j}$ denote the $j$-th neighbor cluster $CL_j$ in $B_{CL_i}$. Let $N(B_{CL_i}^{j}) = N_i$ denote the number of sensor nodes in neighbor cluster $CL_j$.

Given a $N_i$ for a cluster $CL_i$, coordinator node $C_i$ can determine the duty cycle length $L_i$. Duty cycle is composed of many time-slots, and sensor node $N_{ij}$ wakes up at wake-up time slot $\tilde{S}_{N_{ij}}$. Given a sensor network with several cluster $CL_i$. A coordinator node $C_i$ and sensor node $N_{ij}$ in a cluster $CL_i$ perform regular data transmission mechanism as follows.

Step 1. For a cluster $CL_i$, each sensor node $N_{ij}$ initially registers to the coordinator node $C_i$, such that coordinator node $C_i$ knows the number of sensor nodes $N_i$ in the cluster $CL_i$. Each sensor node $N_{ij}$ registers to coordinator node $C_i$.

Step 2. Coordinator node $C_i$ communicates with other neighbor coordinator nodes to obtain the total number of sensor nodes among one-hop neighbor clusters $B_{CL_i}$. Let $N$ denote the number of sensor nodes in neighbor clusters $B_{CL_i}$. The $N$ is computed by

$$N = \sum_{j=1}^{\alpha_i} N(B_{CL_i}^{j}).$$
Step 3. The coordinator node $C_i$ generates a prime sequence $p_i = \{p_{i,1}, p_{i,2}, \ldots, p_{i,s}, \ldots, p_{i,k}\}$ to calculate the duty cycle length $L_i$, where $p_{i,1}$ is a smallest prime integer to satisfy $p_{i,1} > \alpha_i$. The duty cycle length $L_i$ is calculated by

$$L_i = \min(k\sum p_{i,s}), \text{ where } L_i \geq (B_i + N_i).$$

The coordinator node $C_i$ announces the duty cycle length $L_i$, the prime sequence $p_i = \{p_{i,1}, p_{i,2}, \ldots, p_{i,s}, \ldots, p_{i,k}\}$, and the available set of remainder sequence $r_{seq}$, to all sensor nodes $N_{i,j}$, where $1 \leq j \leq N_i$. Each sensor node $N_{i,j}$ receives the duty cycle length $L_i$, the prime sequence $p_i = \{p_{i,1}, p_{i,2}, \ldots, p_{i,s}, \ldots, p_{i,k}\}$, and the available set of remainder sequence $r_{seq}$ from coordinator node $C_i$.

Step 4. A sensor node $N_{i,j}$ chooses a remainder sequence $r_{i,j} = \{i-1, r_2, \ldots, r_s, \ldots, r_k\}$ from $r_{seq}$, and replies the selected remainder sequence $r_{i,j}$ to the coordinator node $C_i$. The coordinator node $C_i$ removes $r_{i,j}$ from $r_{seq}$. Next, coordinator node $C_i$ announces the new $r_{seq}$ and revives the remainder sequence $r_{k}$ in $\{-1, r_2, \ldots, r_s, \ldots, r_k\}$ from the next sensor node $N_{i,k}$ untill coordinator node $C_i$ revives remainder sequences from all sensor nodes in $C_i$.

Step 5. Each sensor node $N_{i,j}$ determines the wake-up time-slot $\hat{S}_{N_{i,j}}^x$, where $\hat{S}_{N_{i,j}}^x = (\sum t_{i,s} p_{i,s}r_{i,j}^{s-th}) mod L_i$, and $r_{i,j}^{s-th}$ is the $s$-th prime number in the $r_{i,j}$. Denote $P_{i} = \{P_{i,1}, P_{i,2}, \ldots, P_{i,s}, \ldots, P_{i,k}\}$ as a combine number sequence of $CL_i$, and $P_{i,s} = \prod_{w=1, u \neq s}^{k} p_{i,w}, 1 \leq s \leq k$.

Denote $t_i = \{t_{i,1}, t_{i,2}, \ldots, t_{i,s}, \ldots, t_{i,k}\}$ as a base number sequence of $CL_i$, and $t_{i,s}$ is a smallest positive integer to satisfy $t_{i,s}P_{i,s}^s \equiv 1 (mod p_{i,s})$. Each sensor node $N_{i,j}$ uses the selected remainder sequence $r_{i,j}$ to determine the wake-up time-slot $\hat{S}_{N_{i,j}}^x$, where

$$\hat{S}_{N_{i,j}}^x = (\sum t_{i,s} p_{i,s}r_{i,j}^{s-th}) mod L_i.$$

The wake-up time period $WTP_{i,j}$ of sensor node $N_{i,j}$ is $\hat{S}_{N_{i,j}}^x - \hat{S}_{N_{i,j}}^{x-1}$. To save energy, sensor node $N_{i,j}$ keeps in sleep mode if the time-slot is not $\hat{S}_{N_{i,j}}^x$. Each sensor node $N_{i,j}$ obeys the transmission schedule to send regular data to coordinator node $C_i$.

Step 6. All wake-up time slots $R_{i,j}$ of coordinator node $C_i$ are determined by $\hat{S}_{N_{i,j}}^x \cup \hat{S}_{N_{i,j+1}}^x \cup \ldots \cup \hat{S}_{N_{i,N}}^x$.

As shown in Fig. 4, $B_{CL_i}$ is $\{CL_2, CL_3\}$ and $\alpha_i = 2$. The smallest prime integer greater than $\alpha_i$ is $p_2 = \{3, 5\}$ and $p_3 = \{3, 5\}$ respectively. The available set of remainder sequence in $CL_1$, $CL_2$, and $CL_3$ are $r_{seq1} = \{\{0, 0\}, \{0, 1\}, \{0, 2\}, \{0, 3\}, \{0, 4\}\}$, $r_{seq2} = \{\{1, 0\}, \{1, 1\}, \{1, 2\}, \{1, 3\}, \{1, 4\}\}$, and $r_{seq3} = \{\{2, 0\}, \{2, 1\}, \{2, 2\}, \{2, 3\}, \{2, 4\}\}$ respectively. Sensor node $N_{i,1}$ gets a remainder sequence $r_{1,1} = \{0, 2\}$, and $\hat{S}_{N_{i,1}}^x = \{(1,1) \times P_{1,1} \times r_{1,1}^{1-th} + (t_{1,2} \times P_{1,2} \times r_{1,2}^{1-th})\} mod 15 = (2 \times 5 \times 0 + (2 \times 3 \times 2)) mod 15 = 12$. Sensor node $N_{i,2}$ gets a remainder sequence $r_{1,2} = \{0, 4\}$ and $\hat{S}_{N_{i,2}}^x = 9$. Sensor node $N_{i,3}$ gets a remainder sequence $r_{1,3} = \{0, 0\}$ and $\hat{S}_{N_{i,3}}^x = 0$. To confirm each $\hat{S}_{N_{i,j}}^x$ is unique time slot for preventing collision, we explain the proof as follows.

Theorem 2: Assume $p_s$ and $p_a$ are co-prime and let $r_s$ and $r_a$ are arbitrary, where $p_s, p_a \in p, r_s, r_a \in r$, and $1 \leq r_s \leq p_s, 1 \leq r_s \leq p_a$. The pair of equations $\hat{S}_{N_{i,j}}^x = r_s (mod p_s)$ and $\hat{S}_{N_{i,j}}^x = r_a (mod p_a)$ have an unique solution for $\hat{S}_{N_{i,j}}^x (mod p_s, p_a)$.

Proof: Assume that $\hat{S}_{N_{i,j}}^x$ and $\hat{S}_{N_{i,j}}^x'$ are two solutions. From the first equation it follows that

$$\hat{S}_{N_{i,j}}^x - \hat{S}_{N_{i,j}}^x = K_1 p_s$$

That is the difference between the solutions must be a multiple of $p_s$. From the second equation it follows that

$$\hat{S}_{N_{i,j}}^x - \hat{S}_{N_{i,j}}^x = K_2 p_a$$

From above equations, it must be that $\hat{S}_{N_{i,j}}^x - \hat{S}_{N_{i,j}}^x$ is a multiple of both $p_s$ and $p_a$. By assumption, $p_s$ and $p_a$ are co-prime that means gcd$(p_s, p_a) = 1$. The least common multiple of $p_s$, and $p_a$ is $p_s \times p_a$, which means that $\hat{S}_{N_{i,j}}^x = \hat{S}_{N_{i,j}}^x (mod p_s \times p_a)$. This proves uniqueness.

B. Burst Data Transmission Mechanism

The main objective in designing the burst data transmission mechanism is to improve the latency of transmitting the burst data. Let $r_{seq used}$ denote a remainder sequence set which
is those remainder sequences \( r_{i,j} \) chose by sensor node \( N_{i,j} \). Based on \( r_{setused} \), sensor node \( N_{i,j} \) can inform a burst data request \( B_{N_{i,j}}^x \) to coordinator node \( C_i \) via the designated time slot \( \hat{R}_{N_{i,j}}^x \), where \( N_{i,k} \neq N_{i,j} \). This structure of \( \hat{R}_{N_{i,j}}^x \) consists of three sections, TR, Data and B section as illustrated in Fig. 5. TR section is used for coordinator node \( C_i \) to receive the burst data request \( B_{N_{i,j}}^x \) from sensor node \( N_{i,j} \). DATA section is used for coordinator node \( C_i \) to receive data from sensor node \( N_{i,j} \) in arranged schedule. B section is used for coordinator node \( C_i \) to announce temporary time slots. The burst data transmission mechanism operates as follows. Multiple sensor nodes \( N_{i,k}, N_{i,k+1}, \ldots, N_{i,k+n} \) send burst data requests \( B_{N_{i,k}}^x, B_{N_{i,k+1}}^x, \ldots, B_{N_{i,k+n}}^x \) to coordinator node \( C_i \) at TR section, where \( 0 \leq n < N_i \). When the coordinator node \( C_i \) detects a collision, the coordinator node \( C_i \) allocates two temporary time slots. Let \( \hat{B}_{C_i}^{1st} \) and \( \hat{B}_{C_i}^{2nd} \) denote the first and second allocated temporary time slots respectively by coordinator node \( C_i \). At the temporary time slot \( \hat{B}_{C_i}^{1st} \) and \( \hat{B}_{C_i}^{2nd} \), each sensor node \( N_{i,k} \) can send burst data after a random backoff time \( R_{i,k} \) if the physical media is free.

Given a coordinator node \( C_i \) and sensor nodes \( N_{i,k}, N_{i,k+1}, \ldots, N_{i,k+n} \) in a cluster \( CL_i \), the burst data transmission mechanism is performed as follows.

**Step 1:** Sensor nodes \( N_{i,k}, N_{i,k+1}, \ldots, N_{i,k+n} \) send burst data requests \( B_{N_{i,k}}^x, B_{N_{i,k+1}}^x, \ldots, B_{N_{i,k+n}}^x \) to coordinator node \( C_i \) at TR section of time slot \( \hat{R}_{N_{i,j}}^x \). Coordinator node \( C_i \) judges the collision condition. Coordinator node \( C_i \) judges that multiple sensor nodes \( N_{i,k}, N_{i,k+1}, \ldots, N_{i,k+n} \) send burst data requests \( B_{N_{i,k}}^x, B_{N_{i,k+1}}^x, \ldots, B_{N_{i,k+n}}^x \) due to the collision condition.

**Step 2:** Coordinator node \( C_i \) announces two temporary time slots \( \hat{B}_{C_i}^{1st} \) and \( \hat{B}_{C_i}^{2nd} \) for sensor nodes \( N_{i,k}, N_{i,k+1}, \ldots, N_{i,k+n} \) at B section of time slot \( R_{N_{i,j}}^x \). The two temporary time slots \( \hat{B}_{C_i}^{1st} \) and \( \hat{B}_{C_i}^{2nd} \) are

\[
\hat{B}_{C_i}^{1st} = (\hat{R}_{N_{i,j}}^x + min(q)) \mod L_i \neq \left( \sum_{i=1}^{k} t_i p_i r_{setused} \right) \mod L_i
\]

\[
\hat{B}_{C_i}^{2nd} = (\hat{B}_{C_i}^{BS} + min(q)) \mod L_i \neq \left( \sum_{i=1}^{k} t_i p_i r_{setused} \right) \mod L_i
\]

The entire temporary time slot \( \hat{B}_{C_i}^{1st} \) is used for coordinator node \( C_i \) to receive burst data from sensor node \( N_{i,k} \). The temporary time slot \( \hat{B}_{C_i}^{1st} \) does not contain TR and B section. At the temporary time slot \( \hat{B}_{C_i}^{1st} \), sensor nodes \( N_{i,k}, N_{i,k+1}, \ldots, N_{i,k+n} \) can send burst data after a random backoff time \( R_{i,k}, R_{i,k+1}, \ldots, R_{i,k+n} \) respectively if the physical media is free.

**Step 3:** The temporary time slot \( \hat{B}_{C_i}^{2nd} \) contains TR and B section. Sensor nodes \( N_{i,k+1}, \ldots, N_{i,k+n} \) send burst data requests \( B_{N_{i,k+1}}^{x}, B_{N_{i,k+2}}^{x}, \ldots, B_{N_{i,k+n}}^{x} \) to coordinator node \( C_i \) at TR section of time slot \( \hat{B}_{C_i}^{2nd} \). Sensor nodes \( N_{i,k+1}, \ldots, N_{i,k+n} \) can send burst data after a random backoff time \( R_{i,k+1}, \ldots, R_{i,k+n} \) respectively if the physical media is free. If coordinator node \( C_i \) detects collision condition again at TR section of time slot \( \hat{B}_{C_i}^{2nd} \), goto step 2.

In step 1, if coordinator node \( C_i \) exactly receives a burst data requests \( B_{N_{i,k}}^x \) and does not detect any collision, coordinator node \( C_i \) only allocates \( \hat{B}_{C_i}^{1st} \) and sensor node \( N_{i,k} \) sets the \( R_{i,k} = 0 \). Example is illustrated in Fig. 6, sensor nodes \( N_{i,k} \) and \( N_{i,k+1} \) use \( r_{setused} \) to compute the next wake-up time slot \( \hat{R}_{N_{i,j}}^x \) of coordinator node \( C_i \). Sensor nodes \( N_{i,k} \) and \( N_{i,k+1} \) send burst data requests \( B_{N_{i,k}}^x \) and \( B_{N_{i,k+1}}^x \) respectively at TR section of time slot \( \hat{R}_{N_{i,j}}^x \). Coordinator node \( C_i \) detects a collision at TR section of time slot \( \hat{R}_{N_{i,j}}^x \). Coordinator node \( C_i \) creates two temporary time slots \( \hat{B}_{C_i}^{1st} \) and \( \hat{B}_{C_i}^{2nd} \) at B section of time slot \( \hat{R}_{N_{i,j}}^x \). Sensor node \( N_{i,k} \) sends burst data first at temporary time slot \( \hat{B}_{C_i}^{1st} \), and does not detect any collision, coordinator node \( C_i \) at TR section of temporary time slot \( \hat{B}_{C_i}^{2nd} \). Coordinator node \( C_i \) exactly receives the burst data request \( B_{N_{i,k}}^x \) and sensor node \( N_{i,k+1} \) transmits burst data at DATA section of time slot \( \hat{B}_{C_i}^{2nd} \).

**IV. PERFORMANCE ANALYSIS**

To evaluate the performance of BMA [6], EMAC [9], and our C-MAC for power consumption and packet latency, we consider simulation scenarios under with upstream/no-upstream data transmission as follows. The system parameters are given below. To discuss the effect of the number of nodes in WSNs, the number of sensor nodes is assumed from 1 to 10 nodes. The simulation environment in this study is a cluster-based network, the coordinator is in the simulation area and all sensor nodes are randomly deployed. The topology of network is fixed and unchangeable. The other system parameters, refered from [8] [9]. The radio propagation range is fixed at 10 meters. Adopting the same power consumption model [9], the power consumption of sensor node in transmitting, receiving, and standby states are 21mW, 14.4mW, and 15 \( \mu \)W, respectively. In the following, the simulated results in our simulation are represented as "Ideal EMAC", "Ideal BMA" and "Ideal C-MAC" for EMA, BMAC, and C-MAC protocols, respectively. When a sensor node is idle listing, the sensor node stay in standby mode. The performance metrics to be observed are:

- **Power consumption:** The power consumption of both coordinator nodes and sensor nodes are consumed in the WSNs. Coordinator nodes and sensor nodes consume variable energy in different radio operational mode.

- **Packet latency:** The time is spent between event generated and packet transmitted. The packet latency time includes upstream and downstream.
An efficient MAC protocol in a WSN is achieved with a low power consumption and packet latency. To illustrate the performance achievements, power consumption and packet latency for EMA, BMAC, and C-MAC protocols are compared as follows.

A. Power consumption

Figs. 7(a)(b) are simulation result of upstream. Let $US_{data}$ denote that sensor node $Ni_j$ sends upstream data to coordinator node $Ci$ and $US_{none}$ denote that no upstream data is sent from sensor node $Ni_j$ to coordinator node $Ci$. Fig. 7(a) gives the average simulation result of $US_{data}$. In BMA, coordinator node $Ci$ receives the requests from each sensor node and broadcasts schedule in the contention period. In EMAC, sensor nodes send requests to contend the transmission time slot. After the contention, cluster node broadcasts the transmission schedule to each sensor node. Both BMA and EMAC consume more energy due to its heavy control overhead. In C-AMC, transmission schedule is chosen by sensor nodes when sensor nodes register to coordinator node $Ci$. Therefore, the minimum control overhead is used. Fig. 7(b) shows the average simulation result of $US_{none}$. EMAC consumes more power than BMA and C-MAC. In EMAC, even sensor node has no data transmission, sensor node still need to receive the control message from coordinator node $Ci$. The energy is wasted on this operation. In BMA, sensor nodes keep sleep in contention period because of no data transmission. In our C-MAC, only coordinator node $Ci$ wake up in the assigned time slot, sensor nodes keep sleep for power saving purpose.

B. Packet latency

Fig. 8 shows the simulation results of packet latency for EMA, BMAC, and C-MAC protocols. Fig. 8(a) and (b) are the average simulation result of upstream and downstream data transmissions. To prevent the collision in EMAC protocol, adjacent clusters active by turns. When some events are occurred, data packets must be stored in the buffer until turn to active duty cycle. This causes the higher latency. In BMA protocol, if burst data detected after the contention period, sensor node buffered the data and transmit the data until the next duty cycle. The worst case of BMA is event occur after the contention period, the packet latency is almost a duty cycle. In our C-MAC MAC protocol, sensor node can request additional time slot when coordinator is wake up to serve other sensor nodes. The result is given in Fig. 8. As a summary, our C-MAC has the better packet latency, compared to all other protocols.

V. CONCLUSION

This paper investigates a new TDMA-based MAC protocol in cluster-based wireless WSNs with low power consumption and low packet latency. We propose an energy-efficient MAC scheme using Chinese Remainder Theorem. To provide low packet latency, our C-MAC protocol provides an adaptive time-slot scheme to distributively and dynamically wake up time slots for the burst data transmission in a duty cycle, where the wake-up time slots are based on the Chinese remainder theorem to avoid the time slot collision between different clusters. Finally, simulation result illustrates our performance achievements to verify that C-MAC performs better than existing TDMA-based MAC protocols, BMA and EMAC, in terms of power consumption and average packet latency.

REFERENCES