
Sarwar Morshed  
University of Twente  
The Netherlands  
s.morshed@utwente.nl

Geert Heijenk  
University of Twente  
The Netherlands  
geert.heijenk@utwente.nl

ABSTRACT
The medium access control (MAC) protocol determines the energy consumption of a wireless sensor node by specifying the listening, transmitting or sleeping time. Therefore MAC protocols play an important role in minimizing the overall energy consumption in a typical wireless sensor network (WSN). Using transmitted reference (TR) modulation in the underlying physical layer opens up new possibilities and challenges to be investigated in the upper MAC layer. Hence this paper presents a new energy-efficient MAC protocol, called TR-MAC, to exploit all the benefits provided by the TR modulation in the physical layer while minimizing the drawbacks. TR-MAC enables both transmitter-driven and receiver-driven communication in the WSN, allows a pair of nodes to use individual frequency offsets for multiple access, and is capable of achieving fast synchronization in the receiver to reduce energy consumption. In short, TR-MAC is an energy-driven MAC layer communication protocol for asynchronous low data rate applications that enables nodes to adapt their duty cycle based on the available energy in the node.

Categories and Subject Descriptors
C.2.1 [Computer-communication Networks]: Network Architecture and Design—Wireless communication

Keywords
WSN; TR modulation; energy-efficiency; energy-driven; MAC protocol; TR-MAC

1. INTRODUCTION

The Medium Access Control (MAC) protocol in wireless sensor networks is responsible for creating a network infrastructure with addressing and providing a channel access mechanism for network nodes to effectively communicate in a shared wireless communication medium. The physical layer underneath the MAC layer deals with the modulation of the data signal to the reference signal and controls the channel access mechanism of the shared communication medium for multiple nodes. Transmitted Reference (TR) modulation [11] is a novel modulation technique in the physical layer that transmits the reference signal together with the modulated signal. Minimizing energy consumption in WSNs is always a big challenge in the MAC layer and this problem maximizes for a modulation technique like TR modulation that takes more energy burden in the transmitter side. Interestingly, TR modulation provides many lucrative advantages for the receiver and all together offers many opportunities to be exploited in the upper MAC layer.

The transmitter using TR modulation sends both the modulated and unmodulated signal with a known frequency offset [14], as presented in Figure 1. Thereby transmitter consumes more power than a traditional modulation technique but this approach enables the receiver some possibilities to reduce the power consumption in the network. Firstly, the receiver can restore the original signal quickly by correlating the received signal with a delayed version of itself using the same frequency offset since all multi-path components contain identically distorted pulses with consistent mutual delay. Secondly, the receiver can achieve faster synchronization consuming less energy without using rake receiver technique, channel state information or power-hungry stable oscillators. Thirdly, multiple frequency offsets can be used as link identifiers for implicit addressing and provide more flexibility to communicate simultaneously. Therefore, TR modulation can be a potential candidate for asynchronous low data rate communication in wireless sensor networks offering additional flexibility to the MAC layer.

In this paper we investigate the possibilities to optimally exploit the characteristics of TR modulation at the MAC layer for WSNs by extending a new energy-efficient MAC layer protocol called TR-MAC. The presented results show that TR-MAC protocol gives better energy consumption than two other reference MAC layer protocols. Also TR-MAC becomes energy-efficient by successfully minimizing the four major sources of energy wastage [26], namely collision, idle listening, overhearing, and control packet overhead.

The contributions of this paper are as follows: (1) we extend a new MAC protocol, TR-MAC, to exploit all the advantages provided by the TR modulation technique while minimizing its drawbacks; (2) we complete the mathematical
The asynchronous preamble sampling MAC protocols try to reduce the preamble duration mainly in three ways: protocols with packetization, schedule learning by piggybacking synchronization information, and adaptive duty cycles [5]. The protocols with packetization enables the transmitter to replace the long preamble by short preamble packet bursts with destination address. As a result, the target receiver successfully receives the data if it receives a single preamble packet, whereas a non-target receiver goes back to sleep after receiving a single preamble packet. Alternatively, the transmitter can send a preamble packet, listen for acknowledgement from the receiver and can continue to repeat this cycle to shorten its preamble length by an acknowledgement from the intended receiver when it wakes up and receives the preamble. However, preamble length adaptation for future transmissions and acknowledgement after successful data transmission are missing in these kind of protocols. X-MAC [4], SpeckMAC-B [24], ContikiMAC [7] are the most common packetized MAC protocols.

Alternatively, some preamble sampling MAC protocols learn the schedule of the next wake up time of another node using piggybacking synchronization information, hence reduce the preamble length using receiver-driven approaches. WiseMAC [8], CSMA-MPS [13], TrawMAC [27], SyncWUF [20] are examples of this category. However, receiver-driven protocols are unsuitable for broadcast traffic because one transmitter might have to adapt its preamble multiple times for its multiple neighbors. Furthermore, these protocols have to transmit the longest possible length of preamble for the first time communication.

Finally, some preamble sampling protocols adapt their duty cycle based on requests from the neighborhood, e.g., BEAM [2], traffic load, e.g., MaxMAC [16], or topology information, e.g., EA-ALPL [12]. However, these duty cycle adaptive protocols do not adapt their network operation based on energy availability on individual nodes and target only specific application scenarios.

MAC protocols with sleep and wake up schedules often need to manage clock drift and maintain timing synchronization to align transmitter and receiver wake up time. Regular communication ensures global timing synchronization within the network, however, low duty cycle WSNs allow nodes to have long sleeping period with less communication [25]. Thus the clocks in the transmitter and receiver may drift apart in long absence of communication and that depends mostly on the time since last communication. Many methods are proposed to maintain synchronization, for example using frequent resynchronization by transmitting more packets [10], using guard times [21], and using packets arrival time for reference broadcast synchronization [9]. However, these approaches consume a significant amount of energy.

3. TR-MAC PROTOCOL DESIGN

We propose a new energy-efficient protocol, TR-MAC, that will exploit the opportunities provided by TR modulation and at the same time will mitigate the transmit power penalty of TR modulation. The TR-MAC transmitter packetizes the preamble since transmitting is costly with TR modulation due to the fact of transmitting both the modulated and reference signal. The transmitter using TR modulation in the early state transmits a small preamble, then listens for a potential acknowledgement from the receiver.
and continue this cycle until it receives and acknowledgement from the receiver. As a result the transmitter has the opportunity to shorten its preamble length from the maximum duration of two consecutive sampling intervals of the receiver, hence minimizing the energy at the transmitter.

TR-MAC saves energy on the receiving side by sleeping most of the time, only waking up to detect any activity in the channel to shorten its idle listening. Furthermore, TR-MAC receiver can detect very small preamble since TR modulation has fast synchronization capability. TR-MAC takes this advantage and appends small data packets ranging from very few bytes with the preamble, thus efficiently reducing the control packet overhead. Therefore preamble-listen cycles will be referred to as data-listen cycles from now on in this paper. The transmitter will indicate the receiver to continue listening using a single bit if larger data packets follow the initial small ones, as in WiseMAC [8]. Also the TR-MAC protocol solves the critical multiple access issue using individual frequency offsets for a pair of nodes, a key advantage offered by the underlying TR modulation. Thus collision can be avoided as future communication will take place in different virtual channels using frequency offsets, hence reduces the expense of costly retransmissions.

The newly proposed TR-MAC protocol has three states to have better capability to adapt to the situation, namely (1) first time communication; (2) unsynchronized state; and (3) synchronized state, as shown in Figure 2. In the beginning of operation, a node knows nothing about the system and its neighbors, this state is referred as first time communication state. In this state, a node transmits data-listen cycles in the default frequency offset if it has data to transmit. Other nodes in the system sleep most of the time and listen to the default frequency offset periodically for any data transmission, like other preamble sampling protocols. If the intended receiver receives a single data packet, then it responds with an acknowledgement indicating a successful transmission. In this state, one pair of nodes discover each other, exchange the full MAC address, and establish a link identifier by agreeing upon the frequency offset to be used for future communications, thus proceed to the next unsynchronized state.

In unsynchronized state the transmitter transmits data-listen cycles at the previously agreed upon frequency offset until it receives an acknowledgement from the receiver as shown in Figure 3. The receiver can derive the link identifier from the frequency offset and preamble part of the data packet. Potential overhearers can return to sleep just after detecting the preamble part of the data packet and decoding the link identifier. The acknowledgement packet sent from the receiver contains the receiver’s next desired wake up time indicating whether its future check interval will be a default one, or a half or double of the previous one. The transmitter will transmit the same time back to the receiver if it agrees with the proposed time, meaning that the transmitter will follow the receiver. We refer this approach of communication as receiver-driven for this time instance. However, the transmitter proposes a new time if it does not agree with the previously proposed time. The receiver transmits the same time back in next acknowledgement if it agrees with the transmitter. We refer this approach as transmitter-driven communication as the receiver now follows the transmitter. Thus the nodes themselves decide whether future communication will be transmitter-driven or receiver-driven, hence providing enormous flexibilities to exploit in the upper layers. In addition, one node having less energy can request other node to follow its lead. A request to follow is considered urgent if the node sends the same time twice without agreeing with other node’s proposal. Afterwards the protocol advances to its final synchronized state.

In synchronized state, the transmitter and receiver pair wake up to communicate in the previously agreed time instance in a known frequency offset. Thus the transmitter can optimally minimize the data-listen cycle length to minimum as possible and the receiver can minimize its listening duration, as shown in Figure 3. Consequently the nodes can communicate in a very energy-efficient manner in this state. However, at low data rates the energy consumption between a pair of nodes sometimes may be more than expected because of the potential clock drifts between the

**Figure 2: TR-MAC: Three states**

**Figure 3: TR-MAC with two reference protocols**
pair of nodes and the last time of communication. We also provide an optimization technique in the synchronized state for TR-MAC to minimize the effect of clock drift that will be explained later in Section 4. But there might be situations when operating in synchronized state would be costly from an energy perspective, then a pair of nodes can roll back to the unsynchronized state to minimize energy consumption.

Two similar MAC protocols, X-MAC [4] and WiseMAC [8] are also presented in Figure 3 with their behavior in corresponding states as TR-MAC. The X-MAC protocol always works in the same way where the transmitter sends packetized preamble and listens waiting for the acknowledgement from the receiver. X-MAC has high control packet overhead since it does not adapt its preamble-listen duration, thus it does not have synchronized state. Also X-MAC does not send any acknowledgement after successful packet transmission.

Alternatively, WiseMAC protocol has unsynchronized and synchronized states where the transmitter adapts the preamble length in the later state depending on remembering the receiver’s next periodic wake up time. In synchronized state WiseMAC efficiently manages clock drifts and minimizes collision using small preambles. However, the potential receiver and the overhearers has to listen the complete preamble for the first time communication before receiving any data, which is costly for per packet overhead in low traffic condition. Furthermore, WiseMAC have to use long preambles for broadcasting instead of short ones and does not adapt duty cycles depending on the changing traffic pattern. The receiver-driven approach of WiseMAC makes a potential receiver collision-prone if many transmitter follows its next wake up and start transmitting at the same time, then WiseMAC has to add a medium reservation preamble in front of the wake up preamble [8].

The possibility of communication using transmitter-driven or receiver-driven technique provides some interesting opportunities to realize energy-efficient multi-hop communications at the network level. For instance, TR-MAC can offer an energy-efficient broadcasting approach by creating a ripple effect, where the transmitter instructs its first hop neighbors to follow its lead and those in turn instruct their respective neighbors to follow them: thus saving energy, increasing throughput and decreasing delay for broadcasting. Alternatively, a receiving node can instruct many transmitters to follow its duty cycle. Furthermore, TR-MAC enables the nodes in the system to adapt their duty cycles based on available energy on the nodes, or traffic load or any other application requirement like increasing throughput or decreasing latency. For example, one node with less energy may request to sleep more to increase overall network lifetime, thus effectively shifting the energy burden to other nodes with more energy. As a result the newly proposed TR-MAC protocol is not only an energy-efficient protocol, rather effectively an energy-driven protocol.

### 4. TR-MAC PROTOCOL MODELING

We present mathematical modeling of TR-MAC for the total system with overhearers for both unsynchronized and synchronized states in terms of energy consumption per second. Moreover, we also present similar analytical models for X-MAC and WiseMAC for the previously mentioned scenarios to compare them with TR-MAC. The comparison results and analysis are presented in Section 5. We use comma separated subscript $T$, $X$ and $W$ to denote a symbol specific for TR-MAC, X-MAC and WiseMAC respectively. If the subscript is omitted, the symbol applies to multiple or all three MAC protocols.

#### A. Unsynchronized state:

Let’s consider there are $n$ nodes where one node will transmit, one node will listen and at most $(n - 2)$ other nodes can be overhearing the communication. Also all the $n$ nodes will have periodic listening. Therefore the general equation to compute energy consumption per second in unsynchronized state for all three protocols for the total system is given by

$$E_{\text{unsync}} = \lambda(E_{\text{unsync}} + E_{\text{unsync}} + (n - 2)E_{\text{OH}}) + nE_{\text{PL}}$$  \hspace{1cm} (1)$$

where $E_{\text{unsync}}$ represents energy consumption of the total system, $\lambda$ being the packet arrival rate, $E_{\text{unsync}}$ is energy to transmit a packet, $E_{\text{unsync}}$ is energy to receive a packet, $E_{\text{OH}}$ is energy spent by the overhearers to receive the preamble and $E_{\text{PL}}$ is energy for periodic listening. We model the energy to receive a packet by extending the periodic listen duration. The overhearers receive one iteration of the preamble and data part only, then sleep without sending any acknowledgement. The energy consumption in unsynchronized state for transmitting a packet, for receiving a packet, for periodic listening, and for overhearing are represented by Eq. 2, Eq. 3, Eq. 4 and Eq. 5 respectively

$$E_{\text{tx,unsync}} = \frac{1}{2}(T_{S,T} + T_{T,T})^2 + (T_{P,R} + T_{R,T}) + T_{A},$$  \hspace{1cm} (2)$$

$$E_{\text{rx,unsync}} = P_{\text{tx}}(T_{R,T} + T_{T,T}) + P_{\text{rx}}T_{A},$$  \hspace{1cm} (3)$$

$$E_{\text{PL}} = P_{\text{tx}}T_{S} + P_{\text{tx}}T_{S},$$  \hspace{1cm} (4)$$

$$E_{\text{OH}} = P_{\text{tx}}(T_{R,T} + T_{T,T}).$$  \hspace{1cm} (5)$$

Here the TR-MAC data packet, $T_{D,T}$, consists of 8 bits of preamble, $T_{P,T}$, 16 bits of header, $T_{H}$, followed by 32 bits of data, $T_{D ata}$, thus having 56 bits. Also the sleeping time and power are represented by $T_{S}$ and $P_{S}$ respectively. Furthermore, $T_{R,T}$ represents the expected extending listening duration. The check interval, $T_{Q}$, here includes the sleeping time between two consecutive sampling intervals and one periodic listen cycle. The symbols and values are given in Table 1. The energy consumption and time for the transmitter-receiver turnarounds and vice-versa are much smaller compared to other values, thus are omitted from our modeling.

<table>
<thead>
<tr>
<th>Description</th>
<th>TR-MAC</th>
<th>X-MAC</th>
<th>WiseMAC</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_{P}$</td>
<td>Preamble</td>
<td>8 bits</td>
<td>65 bits</td>
</tr>
<tr>
<td>$T_{H}$</td>
<td>Header</td>
<td>16 bits</td>
<td>16 bits</td>
</tr>
<tr>
<td>$T_{D}$</td>
<td>Data</td>
<td>32 bits</td>
<td>32 bits</td>
</tr>
<tr>
<td>$T_{A}$</td>
<td>Ack</td>
<td>24 bits</td>
<td>68 bits</td>
</tr>
<tr>
<td>$P_{tx}$</td>
<td>Tx power</td>
<td>2 mW</td>
<td>1 mW</td>
</tr>
<tr>
<td>$P_{rx}$</td>
<td>Rx power</td>
<td>1 mW</td>
<td>1 mW</td>
</tr>
</tbody>
</table>
These equations for unsynchronized state are explained in [15] and are valid for at most one packet arrival per check interval duration.

B. Synchronized state:

In the synchronized state, a transmitter and receiver pair wake up at a previously agreed upon time to listen to a particular frequency offset. Even though the two communicating nodes try to wake up at a particular frequency offset upon time, yet there can be timing mismatch in waking up with respect to each other because of the respective clocks of those two nodes being drifted from the ideal depending upon the time since last communication. The total energy of the system in the synchronized state is calculated for a particular link without any overhearers and periodic listening since nodes are now synchronized and follow each others wake up time. Hence the total energy per second in synchronized state is given by

\[ E_{sync} = \lambda (E_{Tx,T}^{sync} + E_{Rx,T}^{sync}) + nE_{PL} \]  

where \( E_{Tx,T}^{sync} \) represents the expected energy to transmit a packet given by equation 7 and \( E_{Rx,T}^{sync} \) represents the expected energy to receive a packet given by equation 8 respectively.

\[ E_{Tx,T}^{sync} = \int_{d=d_{min}}^{d_{max}} P(D = d) E_{Tx,T}(D = d) \, dd \]  

\[ E_{Rx,T}^{sync} = \int_{d=d_{min}}^{d_{max}} P(D = d) E_{Rx,T}(D = d) \, dd \]  

The probability part of the previous two equations is derived from the possible clock difference between the transmitter and receiver’s clock, represented by the random variable \( D \) with individual realization \( d \). And the energy part calculates the amount of energy needed to transmit or receive a packet for individual clock difference. The equations to calculate the energy will be given in later part.

The main reason for difference between clocks of transmitter and receiver is clock drift. The drift of both transmitter and receiver is clock drift. We assume that the clock drift of both transmitter and receiver’s clock, represented by the random variable \( d \), is uniformly distributed, i.e.,

\[ P(C_T = c) = P(C_R = c) = \begin{cases} \frac{1}{\theta}, & -\theta l \leq c \leq \theta l \\ 0, & \text{otherwise} \end{cases} \]  

where \( \theta \) represents the frequency tolerance of the clock, \( l \) represents the time since last communication and \( c \) represents the individual realizations of \( C_T \) and \( C_R \). For example, the receiver might be \( \theta l \) early or \( \theta l \) late compared to the transmitter based on the clock drift, hence the term \( 2\theta l \). In contrast, the transmitter can also be \( \theta l \) early or \( \theta l \) late with respect to the receiver. We are interested in the difference of the two clocks \( C_T - C_R \), that is, how much the sender and receiver clocks are apart from each other. The difference between two uniformly distributed clock drifts results in a convolution between them that further produces a triangular distribution that will eventually determine the probability of the transmitter and receiver being awake to communicate and is given by

\[ P(D = d) = \begin{cases} \frac{2\theta l}{\theta l}, & -2\theta l \leq d \leq 0 \\ \frac{2\theta l}{\theta l}, & 0 < d \leq 2\theta l \\ 0, & \text{otherwise} \end{cases} \]  

The receiver in the synchronized state spends energy to receive the data, later transmits an acknowledgement after successful reception. The receiver has to listen more than the ideal duration if it wakes up either early or late with respect to the transmitter. If the receiver is early, then it has to listen continuously till it receives a data packet. Alternatively if the receiver is late, then it has to listen continuously more than one iteration of data-listen cycles before transmitting an acknowledgement, thus the transmitter has to transmit more iterations of data-listen cycles. Therefore energy spent by the receiver to receive a preamble with data is given by

\[ E_{Rx,T} = PL(T_{PD,T} + T_{PA,T}) + PL_{sync} \]  

We also derived analytical modeling for the two reference protocols we consider, namely X-MAC and WiseMAC. X-MAC does not have a synchronized state, but wiseMAC has a synchronized form in the protocol since the receiver node include the next waking up time in the acknowledgement so that the transmitter node can follow that time to transmit the data in future. Hence we modeled WiseMAC for syn-
chronous link considering similar equations mentioned for TR-MAC in Eq. 7 and 8. Here the probability part of the equation and the resulting triangular distribution mentioned in Eq. 10 will remain same, only the energy part will differ. Thus we model the energy part of WiseMAC.

Since the WiseMAC transmitter knows the receiver’s next wake up time, it can transmit a preamble of duration $T_{PS,W} = \min\{4\theta, T_W\}$ where $T_W$ is the periodic check interval. This value of the preamble is enough for the possible clock drifts of the transmitter and receiver, and the maximum value of $T_W$ ensures that the next periodic channel sampling will receive the preamble. The energy for WiseMAC transmitter is given by

$$E_{Tx,W}^{sync} = P_{Tx,W}T_{PS,W} + P_{Tx,W}T_{D,W} + P_{Rx,W}T_{A,W},$$

where $P_{Tx,W}$ and $P_{Rx,W}$ represents the power to transmit and receive a packet respectively, $T_{D,W}$ represents the data packet duration and $T_{A,W}$ represents the acknowledgement duration. The WiseMAC receiver spends energy to detect the preamble, to receive the data packet and to transmit the acknowledgement back to the transmitter. The energy for WiseMAC receiver is given by

$$E_{Rx,W}^{sync} = P_{Rx,W}T_{P,W} + P_{Rx,W}T_{D,W} + P_{Tx,W}T_{A,W},$$

where preamble receiving duration, $T_{P,W}$ is considered to have the same value as the minimum duration to detect communication of TR-MAC protocol, $T_{P,T}$.

5. PERFORMANCE EVALUATION

We evaluate the analytical models of TR-MAC, X-MAC and WiseMAC for unsynchronized and synchronized states in Matlab to compare their respective energy consumption per second. For unsynchronized links, the total energy consumption includes the energy to transmit or receive a packet and for periodic listening. The interesting parameters used for analyzing the protocols are check interval duration, $T_W$, and packet arrival rate, $\lambda$. The symbols and corresponding values are given in Section 4 and Table 1. The transceiver power level is considered 1 mW [6], however, the transmitter using TR-MAC uses twice power than other MAC protocols. We considered a data rate of 25 kbps and correspondingly derived the duration of different parts of the packet.

The energy consumption of the overall system is presented in Figure 5 for all three protocols in unsynchronized state for varying check intervals with packet arrival rate of $\lambda = 0.1$ packets/s. We observe TR-MAC performs better than X-MAC but worse than WiseMAC for small check interval whereas the behavior reverses for larger check intervals. Thus we calculate the optimum check interval that minimizes the energy consumption for various packet arrival rate.

The optimized check interval values for a range of packet arrival rates for all three protocols is illustrated in Figure 6 in logarithmic scale for all three protocols. As observed, WiseMAC can consume less energy by making the check interval really small as it can have very short periodic listen only to detect a preamble in the medium. Contrary to that, TR-MAC and X-MAC have longer periodic listen duration in order to spread over the listen part of the data-listen cycles, hence consumes less energy for longer check intervals.

Afterwards we calculate the overall energy consumption in unsynchronized state using the optimized check interval calculated for each packet arrival rate for 12 nodes in the system; where one node transmits, one node listens and...
other 10 nodes are potential overhearers. Figure 7 represents the results and interestingly TR-MAC has better energy consumption than both of WiseMAC and X-MAC in the unsynchronized state even though TR-MAC transmitter has a two times power penalty over other protocols. The reason is TR-MAC overhearers can go back to sleep after receiving the preamble part of data-listen cycles if the same frequency offset is used or the overhearers do not listen to the transmission if a different frequency offset is used. However, WiseMAC overhearers have to listen till the end of the preamble in order to listen to the data packet, that essentially give a rise to the overall energy consumption of the system. Figure 8 represents the comparison of energy consumption for three protocols for different number of overhearers in the system.

The TR-MAC protocol in synchronized state effectively minimizes its preamble in transmitter side as either the transmitter or the receiver follows each others next wake up time. The energy consumption in the synchronized state depends on the relative wake up time of the transmitter and receiver clocks as presented in Section 4 Figure 4. The individual energy consumption of transmitter and receiver depending
on their relative wake up time difference can be observed in Figure 9. If the clocks are aligned, then minimum energy is spent by both the transmitter and receiver. The receiver waking up earlier than the transmitter will be costly in terms of energy consumption. However, the receiver being later than the transmitter will cause the transmitter to transmit more iterations of data-listen cycles, which is more costly than the receiver being early.

The results are presented in logarithmic scale, thus a small difference in the energy consumption. We modeled WiseMAC protocol for synchronized and unsynchronized states. Figure 11 illustrates the comparison where the optimized TR-MAC protocol outperforms WiseMAC with respect to packet arrival rate as illustrated in Figure 10. A clock offset represents the receiver being early on average, and a negative clock drift represents the transmitter being early on average. Interestingly, the total energy consumption does not reach its minimum value when the clock offset is zero, rather at a point with some positive or negative offset difference, which is when the receiver is a bit earlier than the transmitter results in minimum energy consumption. In order to exploit this phenomenon the optimized clock offset is computed by finding the minimum total energy for various packet inter-arrival time. Afterwards the energy consumption in synchronized state is computed by varying the packet arrival rate with the ideal scenario with no clock offset and optimized clock offsets for minimum energy consumption. We modeled WiseMAC protocol for synchronized state, but omitted the X-MAC protocol since it does not have any synchronized state. Figure 11 illustrates the comparison where the optimized TR-MAC protocol outperforms WiseMAC protocol in the synchronized state.

Finally, we combine the overall energy consumption of the system for both unsynchronized and synchronized state with respect to packet arrival rate as illustrated in Figure 12. The TR-MAC protocol will switch between its two available states based on the minimum energy consumption. We see from Figure 12 that TR-MAC outperforms the WiseMAC protocol in the both unsynchronized and synchronized state. The results are presented in logarithmic scale, thus a small difference in this result has more gain in the order of magnitude.

6. CONCLUSIONS AND FUTURE WORK

The TR-MAC with TR modulation underneath is an energy efficient MAC protocol suitable for short-range, low data rate applications that utilizes the advantages of TR modulation and at the same time minimizing its drawbacks. We analytically modeled TR-MAC and compared the unsynchronized and synchronized states of TR-MAC with X-MAC and WiseMAC. It turns out that TR-MAC has a very low energy consumption for both unsynchronized and synchronized states for a system of a realistic number of nodes. TR-MAC periodic listening is not affected by overhearing transmission for other receivers, as in the case of WiseMAC. Furthermore, similar to X-MAC but contrary to WiseMAC, TR-MAC needs very little energy to receive a packet. Finally, transmitting a packet is more costly than in X-MAC, especially due to the characteristics of TR modulation, but this can be compensated by choosing a shorter check interval. Also TR-MAC successfully mitigates the energy wastage by idle listening, overhearing, control packet overhead and collisions. Overall, TR-MAC is very promising for energy-efficient communications in noisy environments where only a limited amount of data is transmitted between a single pair of nodes.

As our future work, we will extend TR-MAC for network level multi-hop communication and evaluate for traffic adaptivity and scalability. We also expect to have better performance for TR-MAC in different scenario with related protocols that use data instead of preamble or use different frequency channels. Finally, energy harvesting will be incorporated in future by letting transmitters and receivers adapt their duty cycle based on locally available energy.

7. ACKNOWLEDGEMENT

This research is supported by the Dutch Technology Foundation STW, which is part of the Netherlands Organization for Scientific Research (NWO), and partly funded by the Ministry of Economic Affairs in the context of STW project 11317 - Walnut: Hard to Crack - Wireless Ad-hoc Links using robust Noise-based Ultra-wide band Transmission.

8. REFERENCES


