Optimization and Verification of the TR-MAC Protocol for Wireless Sensor Networks

Sarwar Morshed, Geert Heijenk

▶ To cite this version:

Sarwar Morshed, Geert Heijenk. Optimization and Verification of the TR-MAC Protocol for Wireless Sensor Networks. 13th International Conference on Wired/Wireless Internet Communication (WWIC), May 2015, Malaga, Spain. pp.396-410, 10.1007/978-3-319-22572-2_29 . hal-01728822

HAL Id: hal-01728822

https://inria.hal.science/hal-01728822

Submitted on 12 Mar 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

Optimization and Verification of the TR-MAC Protocol for Wireless Sensor Networks

Sarwar Morshed and Geert Heijenk

University of Twente, The Netherlands
{s.morshed, geert.heijenk}@utwente.nl

Abstract. Energy-efficiency is an important requirement in the design of communication protocols for wireless sensor networks (WSN). TR-MAC is an energy-efficient medium access control (MAC) layer protocol for low power WSN that exploits transmitted-reference (TR) modulation in the physical layer. The underlying TR modulation in TR-MAC provides faster synchronization and signal acquisition without requiring channel estimation and complex rake receiver in the receiver side. TR modulation also enables multiple access for a pair of nodes using different frequency offsets. This paper introduces an explicit expression that allows the TR-MAC protocol to minimize its energy consumption, depending on the experienced traffic load. Furthermore, an implementation of the protocol in the OMNeT++ simulator with MiXiM simulation framework is introduced, and analytical results introduced in [13] are verified by simulation results obtained using the simulator.

Keywords: Energy-efficiency, MAC protocol, energy-driven, TR modulation, TR-MAC

1 Introduction

Low-power wireless sensor devices have gained popularity in past decade. These low-power devices have to deploy an efficient medium access control protocol together with an efficient modulation technique in the underlying physical layer to enable low power operation. TR modulation is such a low-power spread-spectrum technique [7] where the transmitter sends the unmodulated carrier signal together with the modulated signal in the wireless medium separated by a frequency offset already known to the receiver. The receiver restores the original signal using the known frequency offset by performing self-correlation with the frequency shifted version of the same signal [11]. Hence the receiver enjoys faster synchronization with reduced signal acquisition time without the need of a complex rake receiver or channel estimation or power-hungry stable oscillators. Moreover, multiple frequency offsets using TR modulation can provide implicit addressing as link identifiers. As a result, the TR modulation with its simplified receiver architecture enabling low power, low data rate and low duty cycle operations provides more flexibility to the upper MAC layer for WSNs.

The radio transceiver dominates the energy consumption of a node in a WSN. To offer low power consumption, TR-MAC employs duty cycling by an asynchronous preamble sampling strategy where each node is allowed to switch off its transceiver as much as possible and switch it on periodically to sense channel activity. After the first time communication, the nodes can remember other nodes next wake up time and can communicate at that time to reduce energy consumption, which is suitable for a low duty-cycle based protocol operation. Even though the receiver architecture is simplified significantly with many attractive advantages to exploit in MAC layer, the transmitter consumes more power to transmit individual bits since the reference signal is also sent. This motivated the authors to design a new energy-efficient MAC protocol for this context as minimizing energy consumption is always a big challenge in WSNs. This paper optimizes the energy-efficient low-power TR-MAC protocol introduced in [12],[13], which exploits the TR modulation characteristics in the MAC layer for WSNs. In addition, this paper provides an implementation of the TR-MAC protocol using OMNeT++ simulator and MiXiM simulation framework described in [9] and also provides a validation of the analytical models.

The contributions of this paper are as follows: (1) we provide an explicit optimization for the previously introduced TR-MAC protocol depending on the experienced traffic load; (2) we introduce the detailed design of TR-MAC by means of a finite state machine and implement the protocol in the OMNeT++ simulator using the MiXiM simulation framework; (3) we provide a verification of the analytical models of TR-MAC protocol introduced in [13] to analyze the energy consumption using simulated results obtained from the simulator.

This paper is structured as follows: related work in Section 2 is followed by TR-MAC protocol design in Section 3. Later on, Section 4 presents the TR-MAC modeling and optimization and Section 5 details the implementation in simulator. Finally, Section 6 gives the results and analysis followed by the conclusions and future work in Section 7.

2 Related Work

Energy-efficiency in MAC protocols for WSNs has been studied extensively in the past decade by researchers, see e.g., [1], [3]. As the transmitter using TR modulation has a power penalty, we focus on energy efficient MAC protocols for our research. Contention-less TDMA-like MAC protocols, such as [8], are good for high traffic load but energy inefficient for low data rate WSNs because of its requirement of network-wide time-slot synchronization. Protocols with common active period, like [4],[15], are good for periodic traffic but they also require a certain amount of network-wide active period schedule synchronization. Asynchronous preamble sampling protocols allow the node to sleep most of the time without the need of any network-wide synchronization, thus they are the most energy efficient category of MAC layer protocols, as confirmed in [3].

The simple preamble sampling protocol B-MAC [14] makes the transmitter to send a long preamble covering two consecutive channel sampling of the receiver, and the receiver listens the rest of the preamble before receiving the data packet. WiseMAC [6] takes this basic preamble sampling technique to operate in unsynchronized state and adds a synchronized state using receiver-driven communication strategy where the transmitter adapts the preamble length by remembering the receiver's next periodic wake up time. However, per packet overhead in low traffic increases the energy consumption in unsynchronized link state since the potential receiver and all the overhearers has to listen the complete preamble before receiving the data packet. Furthermore, WiseMAC lacks a transmitter-driven strategy in synchronized state and has to use long preambles for broadcasting instead of short ones in unsynchronized state since the protocol does not adapt duty cycles depending on the changing traffic pattern.

The packetized preamble sampling protocols, like X-MAC [2], enables the transmitter to send a packetized preamble and listen for an acknowledgement and repeat this process till an acknowledgement is received from the receiver. However, X-MAC does not adapt its preamble-listen duration by taking any advantage of the previous communication to shorten its preamble-listen iterations length. Also X-MAC does not send any acknowledgement after successful data packet transmission. CSMA-MPS [10] and ContikiMAC [5] also packetizes the preamble and iterates with listen cycles to optimize the unsynchronized link communication. Furthermore, both these protocols communicate in receiver-driven way by remembering the receiver's next wake up time but they lack the transmitter-driven strategy.

X-MAC and WiseMAC are very common protocols for asynchronous preamble sampling category of protocols. Thus TR-MAC protocol is compared with these two protocols.

3 TR-MAC Protocol Design

Our proposed TR-MAC protocol uses the asynchronous preamble sampling technique to exploit TR modulation characteristics in the underlying physical layer, to mitigate the power penalty of the transmitter side and to ensure energy-efficiency. The TR-MAC protocol has three basic states: (1) first time communication; (2) unsynchronized link state; and (3) synchronized link state, as given in Fig. 1. The first time communication is presented as a separate link state although communication takes place in unsynchronized fashion; because the protocol initializes in this state by performing neighbor discovery, exchanging MAC address and establishing link identifiers using frequency offsets.

As a preamble sampling protocol, the nodes in TR-MAC unsynchronized link state sleep most of the time and have periodic duty cycling to sense activity in the channel. The receiver using TR modulation saves energy by shortening its idle listening with the capability of detecting a very small preamble since it requires small signal acquisition time and enjoys faster synchronization. Thus we packetize the preamble and add a small data packet with the preamble and refer to it as preamble-data for the rest of this paper. Big data packets will be segmented and sent with an indication to the receiver to continue listening, but at

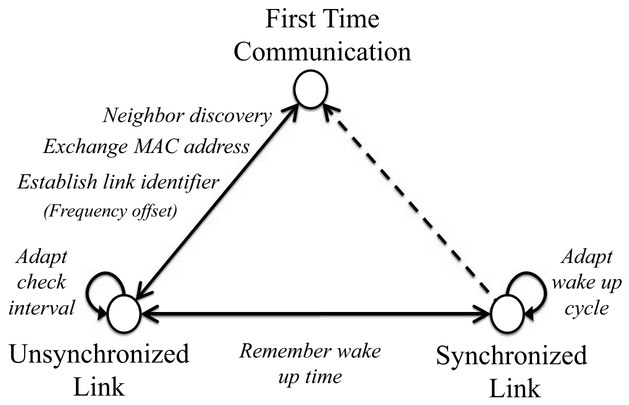

Fig. 1. TR-MAC: Three link states

this moment are out of scope of this paper. The transmitter using TR modulation transmits a preamble-data using the default frequency offset, then waits for the receiver acknowledgement and repeats this cycle until an acknowledgement is received. Receiving an acknowledgement from the receiver marks a successful communication. Thus the transmitter saves energy by shortening its preamble length. Potential overhearers can return to sleep after detecting one preamble-data packet and decoding the link identifier based on the type of preamble and the used frequency offset. The protocol operation is presented in Fig. 2.

In order to move to synchronized state, the transmitter and receiver pair has to agree on future communication time instance and frequency offset. This can happen in two ways: using a receiver-driven or a transmitter-driven strategy. For the receiver-driven case, the receiver sends its future periodic channel sampling time based on optimizing the check interval for a given traffic load in the acknowledgement packet. Thus the transmitter will delay the next packet transmission till the receiver's next periodic channel sampling time, meaning that the transmitter will follow the receiver; hence the term receiver-driven communication. For transmitter-driven case, the transmitter proposes a time instance and receivers adds an extra channel sampling time. Thus the receiver follows the transmitter and hence the term transmitter-driven communication.

The MAC layer enjoys tremendous flexibilities because of the options to communicate in either receiver-driven or transmitter-driven strategy for both link layer and multi-hop communication. For example, one node having less energy can transfer energy burden by requesting other node to follow its lead and can effectively operate like an energy-driven protocol. Efficient broadcasting can take place where one node makes its first-hop neighbors to follow its lead. Furthermore, TR-MAC protocol provides an efficient multiple access scheme to avoid collisions and costly retransmissions by enabling multiple pairs of nodes to use different frequency offsets for their corresponding synchronized link states.

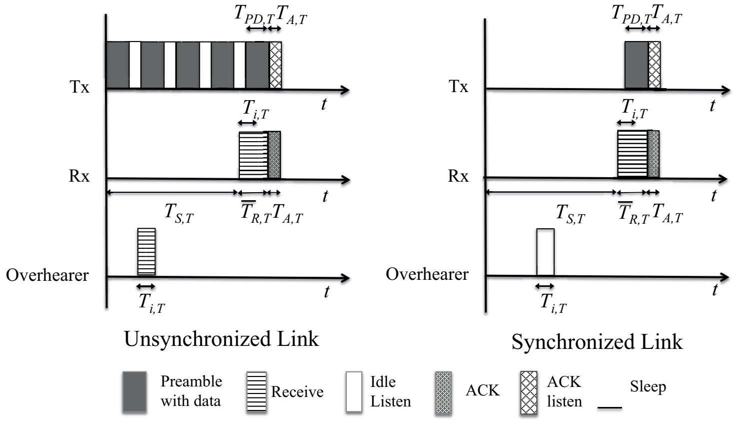

Fig. 2. TR-MAC protocol operation

4 TR-MAC Protocol Modeling and Optimization

The mathematical models of TR-MAC for both unsynchronized and synchronized link states were initially presented in [12],[13] together with the analytical models for X-MAC and WiseMAC to compare with TR-MAC. In this section, we present an explicit mathematical optimization to minimize the energy consumption for the TR-MAC protocol unsynchronized link state by adapting the check interval based on traffic load. A symbol specific for TR-MAC, X-MAC and WiseMAC protocol are represented by comma separated subscript T, X and W respectively. A symbol without a subscript represents all three protocols.

The TR-MAC data packet $T_{PD,T}$, consists of 8 bits of preamble $T_{P,T}$, 16 bits of header $T_{H,T}$, 32 bits of data $T_{Data,T}$; thus having 56 bits. Data rate of 25 kbps is chosen for measurements. The sleeping time and power are represented by $T_{S,T}$ and $P_{S,T}$ respectively. The sampling interval or check interval $T_{W,T}$ includes the sleeping time between two consecutive periodic listening intervals of a node and one periodic listen cycle $T_{i,T}$; hence $T_{W,T} = T_{S,T} - T_{i,T}$. The switching time and power consumption to switch the transceiver among sleeping, sending and receiving states are much smaller compared to other values; thus are neglected in our modeling. The symbols and values for TR-MAC, X-MAC and WiseMAC are given in Table 1. Fig. 2 also depicts the relevant notations.

A. Unsynchronized link state:

The energy consumption for the TR-MAC protocol in unsynchronized link state is given by

$$E_T^{unsync} = \lambda(E_{Tx,T}^{unsync} + E_{Rx,T}^{\text{unsync}} + (n-2)E_{OH,T}) + nE_{PL,T} \qquad (1)$$

where E_T^{unsync} represents energy consumption of the total system that uses TR-MAC protocol, λ being the packet arrival rate, $E_{Tx,T}^{unsync}$ is energy to transmit a packet, $E_{Rx,T}^{unsync}$ is energy to receive a packet, $E_{OH,T}$ is overhearing energy

Table 1. System parameters

Symbol	Description	TR-MAC	X-MAC	WiseMAC
T_P	Preamble	8 bits	65 bits	T_{W}
T_{Data}	Data	32 bits	32 bits	32 bits
T_H	Header	16 bits	16 bits	16 bits
T_A	Acknowledgement	24 bits	65 bits	80 bits
T_i	Periodic listen	40 bits	195 bits	8 bits
P_{Tx}	Tx power	2 mW	1 mW	1 mW
P_{Rx}	Rx power	1 mW	1 mW	1 mW

and $E_{PL,T}$ is energy for periodic listening. We assume n nodes where one node transmits, another node listens and $(n-2)$ other nodes act as overhearers. The energy spent for packet transmission is characterized by the number of times the packet has to be transmitted until the receiver is awake and acknowledges it, multiplied with the energy spent for a transmission. The energy spent for packet reception is characterized by the additional time the receiver is listening beyond the periodic listening together with the energy for transmitting an acknowledgment. An overhearer receives one iteration of the preamble-data only, then sleeps without sending acknowledgement after realizing it was not the target node. The energy to send a packet, receive a packet, periodic listening and overhearing are given respectively by Eq. 2, Eq. 3, Eq. 4 and Eq. 5

$$\begin{aligned} E_{Tx,T}^{unsync} = & \left(\frac{1}{2}\frac{\left(T_{S,T}+T_{P,T}\right)^2}{\left(T_{i,T}+T_{S,T}\right)\left(T_{PD,T}+T_{A,T}\right)}+1\right) \\ & * \left(P_{Tx,T}T_{PD,T}+P_{Rx,T}T_{A,T}\right), \end{aligned} \tag{2}$$

$$E_{Rx,T}^{unsync} = P_{Rx,T}\left(\overline{T}_{R,T}-T_{i,T}\right)+P_{Tx,T}T_{A,T}\ , \tag{3}$$

$$E_{PL,T} = \frac{P_{Rx,T}T_{i,T}+P_{S,T}T_{S,T}}{T_{S,T}+T_{i,T}}, \tag{4}$$

$$E_{OH,T} = P_{Rx,T}\left(\overline{T}_{R,T}-T_{i,T}\right). \tag{5}$$

Here $\overline{T}_{R,T}$ represents the expected value of extended listening duration for the receiver. All these equations for TR-MAC unsynchronized link state are explained together with the expressions for X-MAC and WiseMAC protocols in [12], [13] and are valid for at most one packet arrival per check interval duration.

B. Optimization of unsynchronized link state:

The energy consumption for unsynchronized link state of TR-MAC protocol depends on the chosen check interval. We observed that the energy consumption reaches its minimum value if the protocol can optimize its check interval. Therefore we derive a mathematical expression to find the optimum check interval for the TR-MAC protocol. We use the well known technique to find out a minimum value by first differentiating the equation with respect to the respected variable, then evaluating the differentiated result for that variable. Thus we take the mathematical expressions of Eq. 1 and differentiate with respect to the check interval $T_{\mathrm{W,T}}$, which is given by Eq. 6

$$\begin{aligned}\frac{\mathrm{d}E^{unsync}_{Tx,T}}{\mathrm{d}T_{W,T}} = & \frac{nP_{S,T}}{T_{W,T}} - \frac{n(P_{Rx,T}T_{i,T} - P_{S,T}(T_{i,T} - T_{W,T}))}{T^2_{W,T}} \\ & - \left(\frac{(T_{P,T} - T_{i,T} + T_{W,T})^2}{2T^2_{W,T}(T_{A,T} + T_{PD,T})} - \frac{2(T_{P,T} - T_{i,T} + T_{W,T})}{2T_{W,T}(T_{A,T} + T_{PD,T})}\right) \\ & *\lambda\left(P_{Rx,T}T_{A,T} + P_{Tx,T}T_{PD,T}\right).\end{aligned} \tag{6}$$

Afterwards, we find out the optimized check interval $T^*_{W,T}$ by evaluating Eq. 6 and the result is given by Eq. 7

$$\begin{aligned}T^*_{W,T} = & [\lambda(P_{Rx,T}T_{A,T} + P_{Tx,T}T_{PD,T}) * \{\lambda(P_{Rx,T}T_{A,T} + P_{Tx,T}T_{PD,T}) \\ & *(T^2_{i,T} + T^2_{P,T}) + 2n(T_{A,T}T_{i,T} + T_{PD,T}T_{i,T})(P_{Rx,T} - P_{S,T}) \\ & -2\lambda T_{P,T}T_{i,T}(P_{Rx,T}T_{A,T} + P_{Tx,T}T_{PD,T})\}]^{1/2} \\ & /\lambda(P_{Rx,T}T_{A,T} + P_{Tx,T}T_{PD,T}).\end{aligned} \tag{7}$$

This expression can be evaluated by supplying individual input values for respective variables to optimize the check interval. We also derived similar expressions for X-MAC and WiseMAC for optimizing the check interval but left the derivations for the lack of space.

C. Synchronized link state:

In the synchronized link state of the TR-MAC protocol, a transmitter and receiver pair communicates at a previously agreed time and frequency offset. Using a known time for future communication allows the transmitter to shorten its data-listen iterations. And using a different frequency offset will avoid the overhearers. However, the transmitter in the synchronized link state might have more than one data-listen iterations depending on the precise wake-up time of either the transmitter or the receiver. Because of the potential clock drifts of the nodes, the receiver might wake up earlier than the transmitter or vice-versa, as represented in Fig. 3. The total energy of the system in the synchronized link state for TR-MAC protocol is given by

$$E^{sync}_{T} = \lambda(\mathbb{E}[E^{sync}_{Tx,T}] + \mathbb{E}[E^{sync}_{Rx,T}]) + nE_{PL,T} \tag{8}$$

where $E^{sync}_{Tx,T}$ represents the expected energy to transmit a packet (Eq. 9) and $E^{sync}_{Rx,T}$ represents the expected energy to receive a packet (Eq. 10) respectively

$$\mathbb{E}[E^{sync}_{Tx,T}] = \int_{d=d_{min}}^{d=d_{max}} P(D=d)\, E^{sync}_{Tx,T}|(D=d)\,\mathrm{d}(d), \tag{9}$$

$$\mathbb{E}[E^{sync}_{Rx,T}] = \int_{d=d_{min}}^{d=d_{max}} P(D=d)\, E^{sync}_{Rx,T}|(D=d)\,\mathrm{d}(d). \tag{10}$$

The equations to calculate energy consumption for the transmitter or receiver of Eq. 9 and 10 has an energy part and a probability part. The energy part calculates the energy consumption to transmit or receive a packet for individual clock difference and the probability part represents the possible clock difference of the communicating nodes. The respective clock differences are modeled with random variable D with individual realization d. We assumed uniform distributions for the clock differences for both the transmitter and the receiver. The difference between two uniformly distributed clock drifts results in a convolution that further produces a triangular distribution. This triangular distribution will eventually determine the probability of the transmitter and receiver being awake to communicate at a previously agreed time.

The transmitter does not need to transmit any extra data-listen iterations if the receiver wakes up earlier than the transmitter or at the same time. However, if the receiver wakes up late than the transmitter, then the transmitter needs to send more than one data-listen iterations and the receiver needs to extend its receiving; which results in much energy consumption for both the transmitter and the receiver. This motivated us to experiment with an intentional clock offset. The measurement results showed that the energy consumption is not minimized for zero clock offset, rather at a point when the receiver is little early. Therefore we optimized the synchronized link state energy consumption by waking the receiver little early than the transmitter. The detailed equations and experiment results with the optimization are available in [13] together with the mathematical modeling for the synchronized state of WiseMAC protocol with which TR-MAC was compared.

5 Implementation in Simulator

Simulation is an important technique to evaluate the mathematical model since the system can be realized and tested in a controllable manner in a simulator. Thus we implement TR-MAC protocol in OMNeT++ simulator with MiXiM

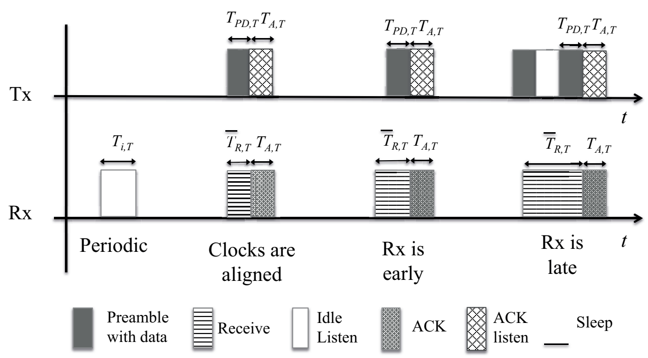

Fig. 3. TR-MAC synchronized link state

simulation framework [9]. OMNeT++ is a discrete event simulator that helps to analyze the wired and wireless protocols by means of events occurring at discrete time instances. An event can be a certain action, for example, generating a packet following a certain distribution, or transmitting a packet at a certain time. The simulated protocol causes the node to change its states between sleeping, transmitting or receiving at discrete point of time depending on the events. All these changes can be analyzed based on the progression of time together with the relative ordering of the events.

MiXiM is a simulation framework that uses OMNeT++ simulator for simulating wireless communication scenarios [9]. MiXiM provides some example protocol stacks for all the layers of OSI model with battery elements for computing energy consumption. Since we are interested to see the TR-MAC protocol behavior, thus we remodeled the MAC layer according to TR-MAC protocol and used the default implementation for the rest of the layers. The individual battery parameters are configured in MiXiM together with power levels where the transmitter uses double power than the receiver. The default network layer is used and the sensor application layer is configured with Poisson packet arrival with exponential inter-arrival times between packets. The physical layer uses simple path loss models without any specific channel effects or propagation delays.

We present a finite state machine for the TR-MAC protocol implementation in Fig. 4 with the possible states and events causing the node to change between its states. Here CCA represents clear channel assessment, CI represents check

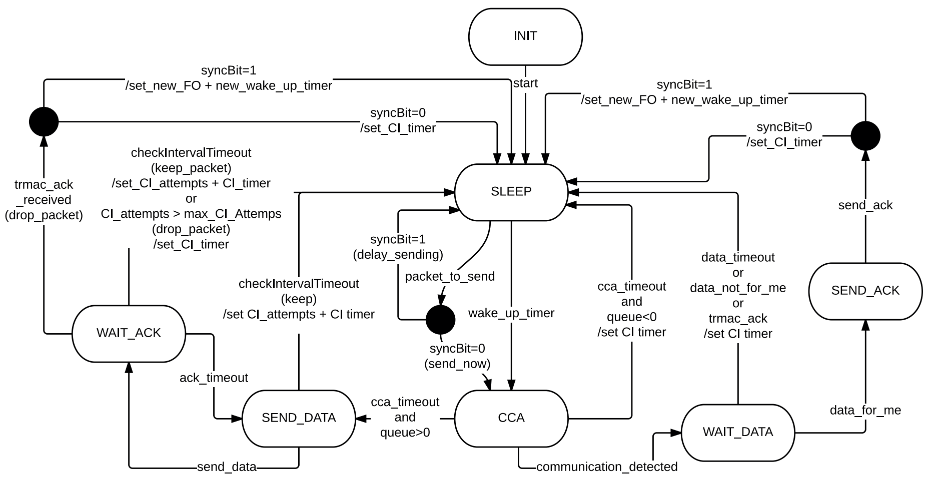

Fig. 4. Finite state machine of TR-MAC protocol implementation

interval and FO represents frequency offset. After initialization in INIT, the node starts in unsynchronized link state and goes to SLEEP. First we explain the finite state machine from the point of receiver. The node wakes up to sense the channel for periodic listening by moving from SLEEP state to CCA state. If the node detects communication during CCA, then it goes to WAIT_DATA state; otherwise goes back to SLEEP. The node receives rest of the data packet in WAIT_DATA state and sends an acknowledgement back to the sender node if the received packet is a data packet and is meant for this node. The node also decides whether to move to synchronized link state by setting syncBit to one, or remain in unsynchronized link state by setting syncBit to zero in the acknowledgement packet before going to SLEEP. Next time this node will wake up for periodic listening or for an agreed time instance for synchronized case or if it has a packet to send; depending on which comes first.

Now a transmitter node having a packet to send in the finite state machine will first check whether it is operating in unsynchronized or synchronized link state. For unsynchronized link, the node moves to CCA state from SLEEP state and further moves on to send the packet right away if no communication is detected during CCA. For synchronized link state, the node continues to SLEEP delaying its wake up to the previously agreed time. If the desired time is reached for the synchronized case, then the node wakes up again and performs CCA. The node will move to SEND_DATA if no communication is detected during CCA, then to WAIT_ACK and iterate in these two states until it receives an acknowledgement or until the default check interval time is reached. After receiving an acknowledgement, the node will again decide about going to synchronized state or not by setting the syncBit to one or zero respectively. The node will drop the packet after a maximum number of failing attempts when it is unable to receive an acknowledgement for several check interval durations.

6 Results and Analysis

In this section, we evaluate the analytical models together with the simulation results for TR-MAC, X-MAC and WiseMAC protocol from the context of energy consumption. We simulate TR-MAC protocol using MiXiM framework in OMNeT++ and present the simulation results with 95% confidence interval. For generating simulation results, we run each simulation to generate and transmit approximately 100 packets and the consumed energy is averaged for the number of packets. Afterwards 100 such simulation runs are used to average the result. We present the energy consumption results with respect to the following parameters, namely the check interval and the packet arrival rate. We consider Poisson packet arrival with exponential inter-arrival times between packets. The parameters and their corresponding representing symbols and values are given in Section 4 and in Table 1.

The total energy consumption for unsynchronized link state includes the energy to transmit or receive a packet, for periodic listening and for overhearing. At this moment, we focus on 1-hop link with a network of twelve nodes where one node is transmitting, another one node is receiving and ten other nodes are overhearing. We present the energy consumption in unsynchronized link state for all three protocols with respect to varying sampling interval for a traffic load

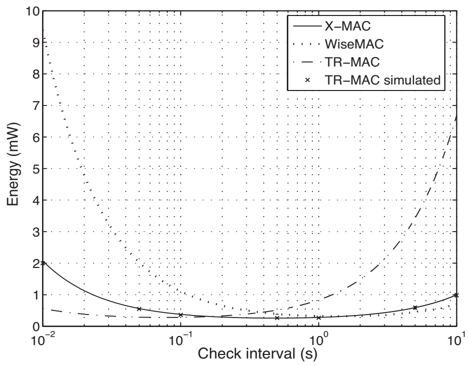

Fig. 5: Unsynchronized state: Energy consumption for packet arrival rate=0.1 packets/s

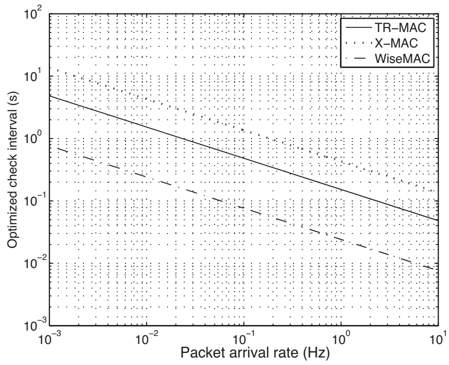

Fig. 6: Unsynchronized state: Optimized check interval for varying packet arrival rates

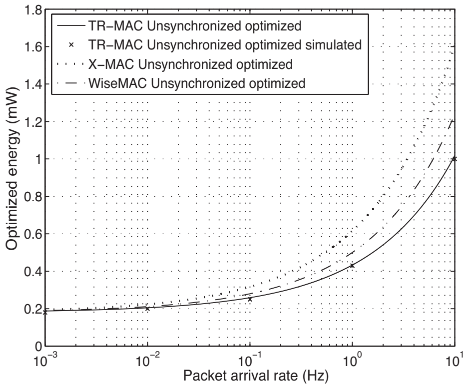

Fig. 7: Unsynchronized state: Energy caclculated using optimized check interval for varying packet arrival rates

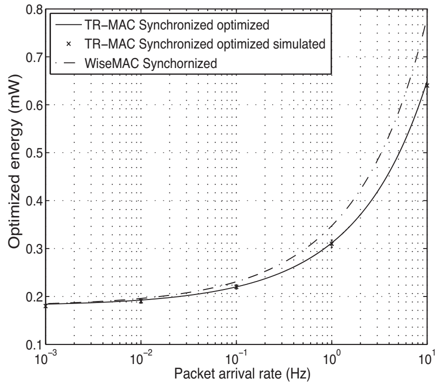

Fig. 8: Synchronized state: Energy consumption for varying packet arrival rates

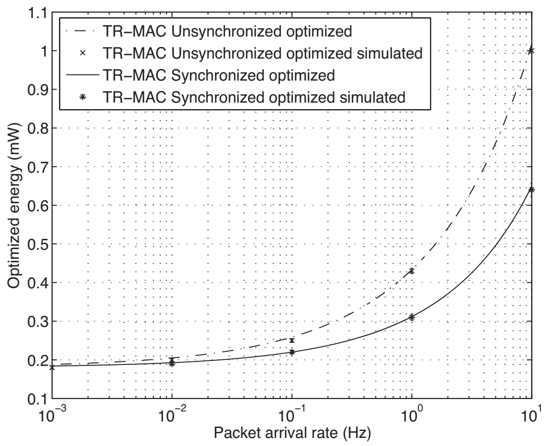

Fig. 9: TR-MAC: Unsynchronized and synchronized state comparison

of $\lambda = 0.1$ packets/s in Fig. 5. We present and explain the energy consumption results for a particular packet arrival rate since we examined similar system behavior for given packet arrival rates. The TR-MAC simulation results, presented on top of the analytical results with 95% confidence interval, matches to the analytical ones. The standard deviation from the mean value is very small, thus the simulation results coincide with the analytical results. We also see that periodic listening dominates the energy consumption for smaller check interval whereas packet transmission dominates the energy consumption for larger check interval. The results shows that the energy consumption for TR-MAC is lower than X-MAC but higher than WiseMAC for a small check interval. But the scenario turns around when the check interval increases. Therefore the energy consumption for TR-MAC protocol can be minimized by optimizing the check interval for a given traffic load.

In Section 4, we present a mathematical expression to find out the optimum check interval to minimize the unsynchronized link state energy consumption for TR-MAC protocol for a given packet arrival rate. We also derived the optimized check interval values for X-MAC and WiseMAC protocol but left out the derivations in this paper. Hence Fig. 6 presents the optimized check interval values calculated using mathematical expressions for all three protocols for a given traffic load. We see that WiseMAC can have smaller check interval because the receiver node only needs to detect the preamble. On the other hand, TR-MAC and X-MAC need to have longer check intervals than WiseMAC.

Next we calculate the overall energy consumption in unsynchronized link state for a range of traffic loads where the protocols use the previously calculated optimized check interval. The results are given in Fig. 7 where the simulation results for TR-MAC protocol with 95% confidence interval verifies the analytical results. We see that the TR-MAC protocol in unsynchronized link state performs better than the reference protocols even though the transmitter using TR-MAC uses double power than other protocols. TR-MAC overhearers can return to sleep after receiving one preamble-data packet, effectively minimizing its periodic listening. On the contrary, WiseMAC has more energy consumption because the overhearers have to listen till the end of the preamble before deciding whether to receive the data packet or return to sleep. The difference between TR-MAC and WiseMAC protocol decreases with small number of overhearers and increases with large number of overhearers.

Afterwards, Fig. 8 presents the energy consumption calculated from both analytical models and simulation measurements for the optimized TR-MAC synchronized link state for varying packet arrival rates. We see that the simulation results confirm the analytical results for TR-MAC protocol. In the same figure, we also present the analytical results for WiseMAC protocol for comparison. We find that optimized TR-MAC performs better than WiseMAC protocol. We leave out X-MAC here since it does not have any synchronized state.

Finally, the overall energy consumption of the TR-MAC protocol for both unsynchronized and synchronized link state with respect to packet arrival rate is illustrated in Fig. 9. The protocol uses optimized check interval for individual traffic loads also in synchronized link state. Here we see the TR-MAC protocol synchronized link state has better energy consumption than unsynchronized link state. Thus the protocol will try to switch to synchronized state if the offered traffic is known. However, the protocol will operate in unsynchronized link state if the packet arrival rate is unknown or a priority packet has to be sent with shorter waiting time. The reason is the TR-MAC protocol in unsynchronized state will wake up more often if per packet delay needs to be minimized.

7 Conclusions and Future Work

This paper presents an implementation of TR-MAC protocol by applying a finite state machine using MiXiM simulation framework and OMNeT++ discrete event driven simulator. Furthermore, an enhancement to minimize the energy consumption for unsynchronized link operation of the TR-MAC protocol has been proposed. We found an explicit expression to optimize the check interval of preamble sampling TR-MAC protocol that will contribute to lower energy consumption. Simulation results with 95% confidence intervals correspond to the analytical results generated for the similar scenario for both the unsynchronized and synchronized link states. The TR-MAC protocol has both transmitter-driven and receiver-driven communication possibilities together with efficient multiple access mechanism using different frequency offsets for different pair of nodes. The analytical and simulation results compared with X-MAC and WiseMAC protocols show that TR-MAC protocol is energy-efficient and suitable for low-power and low data rate WSNs.

For our future work, we will focus on the multiple channel access scheme for TR-MAC protocol. Furthermore, we will extend the protocol for network-level multi-hop communication and evaluate the protocol performance for scalability, throughput and other QoS parameters. Finally, our plan is to include a model for energy harvesting that will enable the TR-MAC protocol to operate based on available energy on the node.

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-wideband Transmission.

References

1. Bachir, A., Dohler, M., Watteyne, T., Leung, K.K.: MAC essentials for wireless sensor networks. IEEE Communications Surveys and Tutorials 12(2), 222–248 (2010)

2. Buettner, M., Yee, G.V., Anderson, E., Han, R.: X-MAC: A short preamble MAC protocol for duty-cycled wireless sensor networks. In: Proceedings of the 4th International Conference on Embedded Networked Sensor Systems. pp. 307–320. SenSys '06, ACM, New York, NY, USA (2006)
3. Cano, C., Bellalta, B., Sfairopoulou, A., Oliver, M.: Low energy operation in WSNs: A survey of preamble sampling MAC protocols. Comput. Netw. 55(15), 3351–3363 (Oct 2011)
4. van Dam, T., Langendoen, K.: An adaptive energy-efficient MAC protocol for wireless sensor networks. In: SenSys. pp. 171–180. ACM (2003)
5. Dunkels, A.: The ContikiMAC Radio Duty Cycling Protocol. Tech. Rep. T2011:13, Swedish Institute of Computer Science (Dec 2011)
6. El-Hoiydi, A., Decotignie, J.D.: WiseMAC: An ultra low power MAC protocol for multi-hop wireless sensor networks. In: ALGOSENSORS. Lecture Notes in Computer Science, vol. 3121, pp. 18–31. Springer (2004)
7. Hoctor, R., Tomlinson, H.: Delay-hopped transmitted-reference RF communications. In: IEEE Conference on Ultra Wideband Systems and Technologies. pp. 265–269 (2002)
8. van Hoesel, L., Havinga, P.: A lightweight medium access protocol (LMAC) for wireless sensor networks: Reducing preamble transmissions and transceiver state switches. In: 1st International Workshop on Networked Sensing Systems, INSS 2004. pp. 205–208. Society of Instrument and Control Engineers (SICE), Tokio, Japan (2004)
9. Köpke, A., Swigulski, M., Wessel, K., Willkomm, D., Haneveld, P.T.K., Parker, T.E.V., Visser, O.W., Lichte, H.S., Valentin, S.: Simulating wireless and mobile networks in omnet++ the mixim vision. In: Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops. pp. 71:1–71:8. Simutools '08, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), ICST, Brussels, Belgium, Belgium (2008), http://dl.acm.org/citation.cfm?id=1416222.1416302
10. Mahlknecht, S., Bock, M.: CSMA-MPS: a minimum preamble sampling MAC protocol for low power wireless sensor networks. In: Factory Communication Systems, 2004. Proceedings. 2004 IEEE International Workshop on. pp. 73–80 (2004)
11. Meijerink, A., Cotton, S., Bentum, M., Scanlon, W.: Noise-based frequency offset modulation in wideband frequency-selective fading channels. In: 16th Annual Symposium of the IEEE/CVT. IEEE/SCVT, Louvain-la-Neuve, Belgium (2009)
12. Morshed, S., Heijenk, G.J.: TR-MAC: an energy-efficient MAC protocol for wireless sensor networks exploiting noise-based transmitted reference modulation. In: 2nd Joint ERCIM eMobility and MobiSense Workshop, St. Petersburg, Russia. pp. 58–71. Bern, Switzerland (June 2013)
13. Morshed, S., Heijenk, G.: Tr-mac: An energy-efficient mac protocol exploiting transmitted reference modulation for wireless sensor networks. In: Proceedings of the 17th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems. pp. 21–29. MSWiM '14, ACM, New York, NY, USA (2014), http://doi.acm.org/10.1145/2641798.2641804
14. Polastre, J., Hill, J., Culler, D.E.: Versatile low power media access for wireless sensor networks. In: SynSys. pp. 95–107 (2004)
15. Ye, W., Heidemann, J., Estrin, D.: An energy-efficient mac protocol for wireless sensor networks. In: INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE. vol. 3, pp. 1567–1576 vol.3 (2002)