Estr - Energy Saving Token Ring Protocol for Wireless Sensor Networks

Faruk Bagci, Theo Ungerer and Nader Bagherzadeh

Abstract — Most wireless sensor networks rely on battery energy. In applications with a great number of sensor nodes and a wide area of monitored environment it is not always feasible to replace the battery. In some cases, even the location of the sensor node is unknown. In order to achieve a long lifetime of the sensor network and the application it is important to save energy. A sensor node consumes the most of its energy in active mode. In sleep mode the energy consumption can be a couple of hundred times lesser depending on the hardware. This paper proposes an energy-efficient medium access control (MAC) protocol for wireless sensor networks, called Energy Saving Token Ring Protocol (ESTR). ESTR bases on the well-known token ring protocol. The sensor nodes are connected in rings where only the node that holds the token is active and able to communicate. ESTR establishes sleep times for sensor nodes that don’t need to send or receive messages. Thus, nodes save energy spending less time in active mode. Also ESTR minimizes energy wasted due to idle-listening and overhearing. The sensor network can build multiple rings which are interconnected. ESTR allows to adjust the maximal permitted ring size offering a flexible mechanism to control throughput and energy consumption. Furthermore the token includes energy information of preceding nodes that allows self-optimization of the ring by dynamically adapting the sleep time of each node. ESTR is tested in the network simulator NS-2 and achieves the best energy results compared to other MAC protocols.

Index Terms — Wireless sensor network, MAC protocol, energy-saving, energy-efficient, token ring, self-optimization

I. INTRODUCTION

Wireless sensor networks consist of multiple sensor nodes exchanging data per wireless connection. Each sensor node can collect, process and transfer local environmental information. This opens a wide range of applications, not only in the military field, but also in environment and habitat monitoring, healthcare systems, home automation, traffic control, and early disaster detections. Many sensor nodes depend on battery energy that should preferably last a long time. Even if a sensor node has an alternative energy source, like in EnOcean [1], the energy is so small that energy saving remains essential. In most applications battery replacements are not feasible because of the great number of sensor nodes or the unknown or unaccessible location. Imagine a scenario where sensor nodes are spread out from an airplane.

A sensor node consumes most of the energy in its active mode. The energy cost rises enormously if the node uses radio communication. The energy consumption of the microprocessor Texas Instruments MSP430F149, which is used in several sensor boards, is 1.6 $\mu$A in sleep mode and rises to 280 $\mu$A in active mode at 1 MHz. In [2] the energy rate between active:receive:send is determined as 1:1.05:1.4, i.e. the energy consumption of sending a message outweighs other tasks. It is plausible that the energy consumption can be highly decreased by establishment of sleep intervals and avoidance of unnecessary packet sending. Using sleep intervals can lead to a contrary effect, i.e. the data exchange is reduced to a shorter time, which can cause higher packet collisions and increase energy consumption.

The Energy Saving Token Ring Protocol maximizes the lifetime of the overall wireless sensor network by using sleep intervals. The nodes in ESTR are connected to a ring where a token is passed along. The node that holds the token is allowed to communicate to other nodes and can control the communication if there are several nodes contending with each other. Other nodes that do not need to communicate sleep in the meantime. ESTR avoids idle-listening and overhearing, i.e. the receiving of foreign messages where the node is not the recipient. Unfortunately sleep phases decrease the throughput and increase the communication latency. In order to remain flexible ESTR offers the possibility to set the maximum size of a ring and allows to build multiple interconnected rings. By changing this parameter you can balance between energy-saving on the one side and throughput and latency on the other side. The token in ESTR has a special field for energy information. Each node fills this field with its own energy level before it sends the token to its next neighbor. The successive node can adapt its sleep period based on the energy level of its predecessor. This mechanism offers a dynamic self-optimization where the ring balances the overall energy and increases its lifetime.

The next section describes related communication protocols for wireless sensor networks. Section III introduces the ESTR protocol and its energy-saving features. ESTR is evaluated in the network simulator NS-2 in order to compare the performance to related protocols. Section IV describes the evaluation results. The paper ends with the conclusion.

II. RELATED WORK

Regarding MAC protocols for wireless communication there are two different approaches: TDMA (Time Division Multiple Access) and contention-based protocols. In TDMA each node has its own time slot within a frame where only itself can send or receive messages depending on the protocol. The message exchange is handled on the
basis of schedules which repeat periodically. In contention-based protocols there are no time restrictions for message transfer. Each node can access concurrently the medium at any time. The protocol tries to avoid collisions and when collisions appear it resolves them.

The most common contention-based protocol is defined in the IEEE 802.11 standard. Although not designed for wireless sensor networks it is the most common protocol in wireless communication. The only energy-saving feature of IEEE 802.11 is the avoidance of collisions. Some wireless sensor protocols are direct descendants of IEEE 802.11.

The Sensor-MAC (S-MAC) protocol [2] is one of them. S-MAC is a slot-based protocol where each sensor node has alternating sleep and awake phases. The network is divided into clusters and all members of a cluster are awake or asleep at the same time. During sleep phase, the node turns off its radio, and sets a timer to awake itself later. All cluster nodes exchange schedules in an initial phase. A node becomes a follower if it receives a schedule from one of its neighbors. A synchronizer is a node that didn’t get any foreign schedules and decides to build its own schedule. This ensures that within a cluster only one schedule is used, i.e. the schedule of the first node, the synchronizer. If a node receives more than one different schedules it follows all of them. Such nodes are members of multiple clusters and have a higher energy consumption, because they have to be awake for each schedule. Within an awake phase all cluster nodes contend with each other for medium access. The contention mechanism of S-MAC is the same as that in IEEE 802.11, i.e., using RTS (Request To Send) and CTS (Clear To Send) packets. The medium is assigned to the node who first sends out the RTS, and the receiver replies with a CTS packet. S-MAC needs a strict timer synchronization in order to achieve correct functionality. Periodic synchronization among neighboring nodes is performed to correct their clock drift. An extension of S-MAC by adaptive listening is described in [3]. If a node A notices an ongoing communication of node B whom it wants to send a message, it sleeps the time until B is ready.

A modification of S-MAC called Timeout-MAC or T-MAC is introduced in [4]. In S-MAC all nodes need to be awake in the contention phase even if they have nothing to send or receive. T-MAC uses a specific timer $T_A$ to shorten the awake phase if the node does not need to communicate. Obviously the $T_A$ is smaller than the contention phase, thus the energy consumption is reduced. But if the timer $T_A$ is chosen too small the node sleeps early missing possible message requests of other nodes. This early sleeping problem could even lead to unfairness. In further extensions of T-MAC this problem is solved by future request to send (FRTS) messages, but this increases again the energy consumption. Nevertheless T-MAC gains better energy results compared to S-MAC.

The Etiquette protocol described in [5] is a TDMA protocol and doesn’t use synchronized awake cycles like in S-MAC and T-MAC. The idea is that each node announces its own office hours where it is ready to communicate. These time slots are established in the initial phase in the way, that neighboring nodes do not have overlapping slots. Thus, the collision risk is minimized, but at the same time the number of slots is increased. This has the negative effect of increasing latency. On the other hand Etiquette has a very good energy balance.

Token ring [6] can be classified as a combination of TDMA and contention-based. Each node has its own time slot (token holding time) where it has to manage the communication with multiple contending nodes. The Wireless Token Ring Protocol (WTRP) [7] comes close to our approach. WTRP was developed for mobile ad-hoc networks. All nodes build a single ring in the initial phase. The aim of WTRP is to maximize the throughput and minimize the latency without restraining the mobility. Energy efficiency is not considered in WTRP because the nodes are mobile devices with strong energy resources like Laptops or PDAs. A mapping of WTRP on wireless sensor networks is $E^2$WTRP that is described in [8]. $E^2$WTRP aims to enhance the energy balance by dynamic adaptation of the token holding time. An active node can send more messages if the token holding time is increased. The frequency of token hand-over is decreased at the same time that reflect in lower energy consumption.

### III. The ESTR Protocol

The **Energy Saving Token Ring Protocol (ESTR)** is based on the idea of the wireless token ring protocol, but chooses a different communication mechanism and introduces energy-efficient techniques. ESTR assumes that sensor nodes can not change their location if attached to the environment. New nodes can join the sensor network or some sensors can fail due to energy loss, but this is not assumed to be in an ad hoc manner like in mobile ad hoc communication networks. Using sleep phases is still the best approach to save energy. ESTR introduces sleep periods for sensor nodes which does not need to send or receive messages. Thus nodes spend less time in active mode and even receive less foreign messages reducing the overhearing problem.

Usually in token rings only the node which holds the token is allowed to send messages. In ESTR we have reversed this communication scheme by allowing the token holding node to receive messages, because the reception consumes less energy than the sending.

Using sleep periods leads to a problem because nodes don’t necessarily know if the neighbor is awake or not. This problem rises if the two nodes are in different neighboring rings. Since the token is sent in a message the node has to be sure that its neighbor receives the message. To ensure full token ring functionality, ESTR offers solutions for token loss, multiple tokens, and node failures that are described below.

#### A. Network Structure and Configuration

The sensor network in ESTR can consist of one large ring or multiple smaller rings that are interconnected. Each structure has its advantages and disadvantages. One ring built by all nodes comes close to the original Token Ring,
but this is not always possible and depends on the location of nodes. Imagine a sensor node with only one neighbor where the role of predecessor and successor is given to one single node. As long as all nodes can reach each other, we can first build a tree structure to determine the Eulerian cycle as a ring structure. Since we use radio transmission, the edges in the tree are undirected. That means that the father node and son node in the tree can reach each other, as seen in Figure 1. The Eulerian cycle is obtained by visiting each node of the tree in accordance to a depth-first search and putting the edges in visited order to a path together. The path starts and ends at the root and reaches all nodes of the tree. This leads to some nodes that are visited more often than others, if they have several children, but it creates a circular structure. Another possibility is to modify the algorithm of WTRP so that the predecessor and successor nodes for a new node can be the same. This allows the integration of nodes in the ring, which can only reach one node. In both algorithms one node needs to be declared as root which initializes the ring structure. The modified WTRP solution costs less energy, since no additional messages need to be sent to organize the ring structure. The Eulerian ring algorithm on the other hand brings more structure in the ring, which can be used for further improvement.

![Fig. 1. Eulerian cycle](image1)

The process of building multiple interconnected rings except for some minor changes is similar to the process of building one large ring. First, each node in the network has to be a potential root of a ring, i.e. after a random time the node decides itself to create a ring if it didn’t receive an invitation from an existent ring in its vicinity (see Figure 2). Interconnected rings require advanced collision handling, since foreign messages can cross ring borders. The number of collisions also increases in the initialization phase, as several rings are built in parallel causing more message traffic. On the other hand, building smaller rings reduces initialization time.

In ESTR we can control the number of rings in the network by setting a maximum size of ring members for each node. Within the initialization phase the root node sends invitation messages until the maximum size of members is reached. In order to finish this initial process there is an initialization timer. After it expires, each ring remains with the obtained number of members, even if the maximum is not reached. Controlling the size has a direct effect on the latency and throughput of the network. It is applicable if the network is not reached. Controlling the size has a direct effect

![Fig. 2. Multiple interconnected rings](image2)

cation dependent which results in a ring size.

### B. Node Communication

As mentioned above in ESTR only the node that holds the token receives messages from neighboring nodes. After expiration of the token holding time ($THT$) the current node $A$ has to hand over the token to its next ring neighbor $B$. In order to be sure that $B$ is awake and able to receive the token, $A$ waits for a $M_{idle}$ message from $B$. If $A$ receives this messages in the time of $T_{waitMax}$ it forwards the token to $B$. The time $T_{waitMax}$ is two times longer than the maximum sleep period of the nodes $T_{sleepMax}$ to ensure enough time for token hand-over.

If node $B$ doesn’t receive the token after sending $M_{idle}$, it repeats the $M_{idle}$ message. If the token is still missing after $T_{waitMax}$, $B$ stops sending $M_{idle}$ assuming that its predecessor failed. On the other side node $A$ will only go asleep when $B$ stops to send the awake message $M_{idle}$ ensuring that the token is received correctly. Through this secure transmission a token can only be lost if the token holder fails or if the token holder cannot reach its neighbor in $T_{waitMax}$. In both cases the process of token recovery is the same. If the token could not be transferred in the designated time, either the token holder or the token expecting node will induce the dissolution and then the reorganization of the ring by sending respective messages. If the network consists of multiple smaller rings the reorganization is executed with only few messages.

Nevertheless delay or failure in communication can lead to the case that more than one token can exist in the ring. In order to solve this problem each token has two numbers: the sequence number ($SN$) and the generation sequence number ($GSN$). $SN$ is used to count the number of passed nodes. $GSN$ can only be increased by the root and counts the number of rounds. A node deletes all tokens with a $SN$ or $GSN$ smaller than the last sent token. The root which can only change $GSN$, deletes all multiple tokens.

Assuming that there is no node failure and the token is transferred correctly from $A$ to $B$, the communication can go on. In this case, node $A$ sends its data to $B$, since $B$ as the new token holder is now allowed to receive data. $B$ acknowledges each received data packet by sending an
ACK message. Figure 3 describes the message exchange.

![Diagram of message exchange](image)

Fig. 3. Token hand-over and message exchange

Other nodes from neighbor rings that desire to send data to B, have to wait until A has no more packets to transfer. This initiates the contention-based phase of ESTR. After the last packet of A the other nodes set a random timer before they send a request to B. Node B decides if it has enough time before THT expires and sends a permission to the first node. This process reflects the RTS/CTS-approach of IEEE 802.11 that is also used in S-MAC and T-MAC.

The nodes cyclically change their state during the lifetime in order to react correctly to incoming messages. All possible states and transitions are described in Figure 4. After the initial phase all nodes are in the sleep-state except for the root node which generates the token at first. The root is in the state sendToken. As soon as its successor awakes, it changes to the waitToken. After the state sendData a node can either go to sleep or wait for another node in a neighboring ring if it wants to send data to it. In some cases, a node could miss its sleep phase waiting for other nodes and changes directly to the waitToken state. Figure 5 shows all sections of the awake phase.

![State chart of the ESTR-MAC-level](image)

Fig. 4. State chart of the ESTR-MAC-level

![Sections of the awake phase](image)

Fig. 5. Sections of the awake phase

C. Self-optimizing by Dynamic Sleeping

The length of the sleep phase depends on the number of nodes in a ring. The more nodes are in the ring the longer a single node can sleep. But this affects also the message transfer time because the path through the ring becomes longer. A node that has to send many messages to neighboring rings cannot use its sleep periods so often and will fail faster due to energy loss. In order to balance the energy consumption in the ring, ESTR offers a dynamic self-optimizing approach where nodes can change their sleep duration. ESTR uses the token to transfer energy information from node to node.

The token holder A uses the energy field of the token to inform the next neighbor B about its energy level. B increases its sleep duration by a constant factor c if the remaining energy of node A is factor c higher than the energy level of B. The effect of this adaptation is seen as A forwards the token again to B, i.e. in the next cycle. This time A has to wait longer for B who sleeps longer. If the energy level between A and B is balanced, B sets back its sleep duration. Since node A has now a higher energy consumption, it can also adapt dynamically its sleep duration. Over the time the energy consumption of the entire ring becomes dynamically balanced, increasing the overall lifetime of the sensor network.

IV. Evaluation

In order to evaluate the performance of ESTR and to compare it with related approaches, we used the network simulator NS-2 from University of Berkeley, California [9]. The protocol stacks of IEEE 802.11 and S-MAC were already available in NS-2. We implemented the protocols T-MAC, Etiquette, and WTRP according to information given by the authors.

A. Average Energy Consumption

The average energy consumption is an essential factor for the lifetime of the wireless sensor network. A high energy consumption leads to faster node losses. We measured the energy consumption of ESTR in comparison to the other protocols, increasing the number of nodes in the sensor network. Figure 6 visualizes the energy consumption of a wireless sensor network with 20 nodes. The chart shows clearly that IEEE 802.11 does not have energy saving possibilities except for CSMA/CD. It loses the most energy as compared to other protocols in this scenario with 20 nodes.

![Energy efficiency with 20 nodes](image)

Fig. 6. Energy efficiency with 20 nodes

It is important to note the high energy loss of Etiquette and ESTR at the beginning. After some time both proto-
cols decrease linearly as S-MAC and T-MAC, but gaining
the best energy results at the end. ESTR as well as WTRP
build only one ring since the maximum number of nodes
per ring is set to 30 nodes. The initialization costs for
smaller rings are still acceptable, as clearly seen at WTRP
that has similar results as ESTR and Etiquette. S-MAC
is factor of two better than the 802.11 protocol due to the
sleep phases. T-MAC with its timeout mechanism is even
better than S-MAC.

What we can also see here is that even in smaller net-
works TDMA- and token-based approaches are more effi-
cient than contention-based protocols. The difference rises
if the number of nodes becomes higher. Figure 7 shows the
energy consumption in a networks with 100 nodes. This
time WTRP needs more time for building the token ring.

S-MAC as well as T-MAC have a higher energy con-
sumption compared to 20 nodes because the nodes have to
send and receive more messages, e.g. in the synchroniza-
tion phase. ESTR and Etiquette offer nearly the same en-
ergy efficiency. In ESTR, the network consists of multiple
rings, hence, the initialization phase does not need more
time. Regarding the overall results, the average energy
consumption increased compared to the smaller network.
As a last step we increased the number of nodes to 500.
Figure 8 shows that the task of building the rings takes
disproportionately more time.

WTRP cannot manage to build the ring within the de-
finite time of 300 seconds. After expiration of this time
the nodes start to reorder. In a second step we set up the
timeout for building the ring to 400 seconds. Even this
was not enough to manage the initial process, and WTRP
loses a large amount of energy after the timeout. The rea-
son for this is that the remaining nodes try to build their
own ring after 300 seconds. The invitation to join the ring
is sent from node to node in WTRP. This becomes a prob-
lem if there are too many nodes. At 100 nodes WTRP
needed 150 seconds to reach each node, at 500 nodes it
can not reached most of them in 300 seconds. Another
problem is the calculation of the node sleep time. In order
to compute the sleep phase which depends on the number
of actual nodes, the token has to be passed through the
ring at the end of the initialization. Assuming an awake
time of 0.25 seconds and 500 nodes, the token should take
62.5 seconds for one circulation to conclude the building of
the ring.

This is also the reason why ESTR as well as Etiquette
are more energy efficient. ESTR also suffers from hav-
ing too many nodes. The packet collisions increase re-
transmissions, which consume more energy. Also it could
be that some roots cannot get new members since they
are surrounded by existing rings. But the great advan-
tage of ESTR is that these nodes can still communicate as
rings with only one member having their own sleep periods.
Thus, the initialization phase does not need to be extended
in ESTR using the same time like in smaller networks.

In the beginning phase, ESTR seems to consume more
energy as WTRP. The reason is that in ESTR multiple
roots try to build rings in parallel. In WTRP there is only
one root that sends invitations to other nodes. But ESTR
quickly compensates this negative energy balance after the
initialization.

In the previous evaluations we did not used the self-
optimizing mechanism of ESTR using dynamic sleeping. In
order to examine the impact of this feature for energy con-
sumption, especially the balance of energy between nodes,
we ran some measurements using a network with only 30
nodes.

If a node loses faster energy than its neighbors, then the
loss of this node will impact the network greatly. The re-
mainding nodes will have to take over the tasks of the lost
node. Thus, we consider the node with maximum energy
as well as the node with minimal energy to draw conclu-
sions about the lifetime of the overall network. Etiquette,
ESTR and T-MAC can offer the best results (Figure 9).
The proportion between awake and sleep phases have not
changed, but ESTR has now the possibility to balance the
energy in the sensor network by dynamically adapting the
sleep phase.

Since the lines of S-MAC and IEEE 802.11 overlap in
the chart, the values of S-MAC can not be recognized di-
rectly. The timeout functionality of T-MAC ensures that
the energy of the node with the least energy does not de-
crease rapidly compared to S-MAC. In ESTR one can see
improvements because of dynamic sleep adaptation.

The next chart (Figure 10) shows the energy quotient. $E_{\text{Min}}$ is the energy minimum at a point in time and $E_{\text{Max}}$ is the energy maximum at the same time. Thus, the energy quotient is $E_{\text{Quot}} = E_{\text{Min}}/E_{\text{Max}}$. With this value we can see the relative distance between minimum and maximum of energy. In IEEE 802.11 the minimum and maximum are always equal. The nodes have no sleep periods and the number of sent and received packets is for each node the same. S-MAC and WTRP have the worst results. It is assumed that in S-MAC the synchronizer has the lowest energy since it has to manage the initialization. Its the same with the root node in WTRP. The Etiquette protocol does not have a designated node that leads to a better energy quotient. ESTR can compensate the difference at the initial phase through dynamic sleeping. In some phases the energy is balanced, and the minimum and maximum are equal. In T-MAC the energy quotient is decreasing constantly which means that the minimum and maximum lose the same amount of energy.

B. Communication Latency

Even if the energy consumption is the leading point of ESTR, the latency is noteworthy. In applications with time-critical events, latency becomes a main focus. In order to measure the delay we chose a ping-pong communication where a message is sent from node $A$ to $B$ and immediately back again to $A$.

Especially in ring structures where messages between two physical neighbors often have to pass all nodes of the ring this approach is more significant than the latency of single packets. In most of the network structures the latency of a ping-pong message is determined by halving the transmission time, since the node can reach each other directly.

In our latency evaluation we used a single ring with 30 nodes. Thus, the results of ESTR also represent WTRP regarding the latency. We compared our results with S-MAC and IEEE 802.11. First we chose two nodes that were physical as far away as possible. The 802.11 protocol doesn’t have sleep phases, thus it has the least latency as seen in figure 11. The first chart has a large scale, where the value of 802.11 is not visible. The second chart uses a smaller scale to estimate the time of 802.11. In ESTR the message is sent in one or two cycles through the ring. Each cycle consists of awake and sleep phases as mentioned above. In S-MAC the slots for sending are short. The message has to pass multiple intermediate nodes. At each station it has to wait another sleep phase to be forwarded. This results in large delays, if the nodes are far away from each other.

C. Real-life Tests

Beside the NS2 simulations we implemented ESTR on sensor boards to evaluate its usability in real-life scenarios. The used hardware was the Embedded Sensor Board (ESB 430) from the FU Berlin [10]. This sensor board has the microcontroller TI MSP430, a radio transceiver, and several sensors on board. The MSP430 is a low-power mi-
crocontroller offering four different sleep modes. Thus the sensor board ESB430 is appropriate to test energy-efficient protocols.

We implemented two scenarios: one ring with three sensor nodes and two rings with three nodes per ring. One node was always connected over the serial port to a PC where we could get visual outputs in order to observe the communication.

To establish the desired ring structure, we started the nodes successively. The first node that was attached to the PC became the root. In the first scenario, the other nodes sent a message to each other every 96 seconds, after joining the ring. Since every message passed the root, we could follow the ongoing communication. In the two ring scenario we set the maximum member size to three. Successively starting of the nodes should result in two different rings interconnected through one or more nodes. To test the ring-to-ring communication we sent messages from the root node to all other nodes which answered with an ACK packet.

The firmware of the ESB430 has an interface where you can get the remaining energy of the battery. This led us to the idea also to measure the energy consumption of ESTR in real-life. We implemented the IEEE 802.11 protocol to compare the energy usage with ESTR. Since the firmware does not provide accurate energy values, battery energy is surprisingly lower at the beginning of the measurement and fluctuates during the time (Figure 13). Nevertheless it is clearly seen that ESTR achieves a better energy balance as IEEE 802.11 although it is a small network with only three nodes. The main advantage of ESTR becomes apparent in larger networks.

V. Conclusion

This paper presented the Energy-saving Token Ring Protocol ESTR for wireless sensor networks. ESTR uses sleep phases to decrease the energy consumption. Only the node that holds the token can receive messages from other nodes. This reduces idle-listening and overhearing. ESTR provides to adjust the maximal permitted ring size, offering a flexible method to control throughput and energy consumption. Furthermore, the token includes energy information of preceding nodes that allows self-optimization of the ring by dynamically adapting the sleep time of each node. In detailed simulations, ESTR was compared to related MAC protocols. ESTR achieves together with Etiquette the best energy results. But the probability of collisions in ESTR is less than in Etiquette and ESTR offers a dynamic energy optimization in the overall network, increasing network life-time. In further evaluations we measured communication the latency. Although the advantage of ESTR lies in energy-efficiency, it achieves reasonable latency results. We also proved the usability of ESTR in scenarios with real sensor boards.

REFERENCES