INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2017, VOL. 63, NO. 2, PP. 209-216
Manuscript received December 8, 2016; revised April, 2017. DOI: 10.1515/eletel-2017-0028

The channel for hidden data transmission in WSN

Radosław O. Schoeneich, Marcin Golański, Michał Kucharski, Marek Franciszkiewicz, and Dawid Zgid

***Abstract*—This paper describes an idea and realisation of hidden data transmision using Tiny Aggregation Covert Channel (TAGCC)in Wireless Sensor Networks. Our solution uses data aggregation mechanism called Tiny Aggregation (TAG). The protocol is based on idea of hidden messages sending without generate additional data packets and encryption. The paper describes details of proposed algorithm and simulation results obtained during testing of the sensor networks with hidden channel TAGCC.**

***Keywords*—Aggregation, message hidding, MANET, WSN**

I. Introduction

DURING the last decade a huge effort was done in many aspects of Wireless Sensor Networks. Wireless Sensor Networks are composed of nodes equipped with measurement devices which can serve different measurement data. These networks are designed primarily to detect and monitor specific phenomena such as the wildlife applications, industrial automation, military, and domestic applications.

Sensor networks are relatively uniform, very often nodes are of the same type and perform the same or a very similar function, they are homogeneous in its structure. An important feature of sensor networks is usually a dense deployment of devices, which necessitates the creation of highly efficient protocols for communication in these networks.

The network topology is not strictly specified, and the location of the nodes may vary with time. This causes a special algorithms need for sensors that will be capable to self organization of the network with the ability to adapt to changing conditions. Due to the limited range of wireless communication, to extend the range of the transmission the necessary is the use of intermediary nodes which are retransmitting information.

The ability to transfer data over long distances thus mainly depends on the number of nodes in the network. Wireless Sensors Network communication is done by exchanging data packets on the common radio channel. This fact means that the network is flooded by packets, which adversely affects the performance due to its large number of transmitted data, which are often very similar. For this reason, the data aggregation is very commonly used.

The data aggregation involves the division of the sensor nodes performing the roles of sensors that generate data and aggregate nodes which processing data received from the sensors. Aggregators perform mathematical operations on data received from the sensors to form aggregated message and passing it toward the sink of the data. The aggregated data is usually only one type, such as measurement of temperature, pressure, etc., so that they are effectively processed during propagation in the network. Putting them additional other types of information is unusual and unexpected. The purpose of this paper is to present the idea of transferring additional information hidden in the standard information transmitted between sources sensors nodes and node outlet via intermediate nodes and aggregate information.

II. State of the Art

A. WSN Routing Protocols

The essential feature of the information flow in a typical sensor networks is the unstructured and explosive character. Often, WSN nodes try to send data packets via radio, regardless of whether the transmission medium is currently available or energy resources of the node are sufficient. On the one hand WSN routing protocols trying to organize the flow of information and, secondly, trying to ensure that is done in the most efficient way. Thus, despite the wide range of routing protocols designed for a MANET only few of them can be adopted directly in WSN networks.

The most common and also the simplest routing protocol of WSN is flooding. It consists in flooding the whole network with packets, substantially without specific direction. To protect the resources against such action it is necessary to define network architecture. This entails the recognition of basic types of nodes, such as: (a) sink node; (b) leaf node; (c) forwarding node. The first two determine the direction of data flow from the leaf to the sink. If on the data path other nodes exist they are used as forwarding nodes. This creates a layered network architecture. When the leaf node is an immediate neighbor of the sink node they form a one layer. Otherwise a forwarding nodes are used and thus additional layers are created . Their number depends on how far away is the sink node. Typically, in each layer at least one forwarding node exist. An example of a protocol for the construction of the layered architecture is [16].

The second type of architecture is the cluster network. Generally, the nodes are organized into groups managed by the node called head. This node collects the data and then sends it to the sink node. An example of a protocol used in this case is [17].

To optimize the data packets flow, many WSN network designers distinguishes between two basic stages of the communication process. The first is the distribution of control messages, usually in form of queries, generated by the sink node called dissemination while the second is called the gathering. Such communication allows to send back to the sink node only this data which he is interested in. The data gathering phase, characterized by high data traffic flow to the

Radosław O. Schoeneich, Marcin Golański, Michał Kucharski, Marek Franciszkiewicz, Dawid Zgid are with the Institute of Telecommunications, Warsaw University of Technology, Warsaw, Poland (e-mail: rschoeneich,mgolanski@tele.pw.edu.pl).

sink node may be used to transmit additional information. In particular case these information can be hidden.

A typical routing protocol used for gathering phase is PEGASIS [18]. Assuming that the network topology is known by each node a chain of transmission can be created with the nodes called leaders. They aggregate the data and then send them directly to the sink node. In each round of gathering, probability of becoming a leader node is the same. This guarantees equal load of nodes and minimizes the number of sent data packets.

An example of another algorithm is Binary Scheme [19]. This method similar to PEGASIS forms communication chain also. However she divides whole process into rounds in which nodes are grouped into levels. Each group aggregates the data. With each round, number of nodes involved in the procedure falls twice. This means that every time the communication chain is divided into equal groups of nodes. The data transmission takes place using CDMA technique. If this is not possible it is necessary to create a tree of connections instead of the single chain and sequentially transfer the data using TDMA technique. Data aggregation methods used in the phase of data gathering seem to be promising as a undiscovered area of WSN network. Especially in terms of creating hidden channels of information exchange. Similar techniques commonly used in communication in wireless networks usually cannot be directly adapted. This is due to the limitations of WSN nodes, low computing power, small memory resources and insufficient power sources [20]. Therefore, there can also think about steganography in WSN in accordance with its usual definition. Over the years, attempts have been made to use different solutions using eg. 802.15.4 physical layer protocol to transmit hidden content [21][22]. These works demonstrated the potential issues associated with knowledge areas of creating a special hidden information channels in WSN networks. However, according to the best knowledge of authors of this paper [23][24][25][26], there are no well documented algorithms and methods used for this purpose.

To the group of data aggregation protocols belongs the TAG protocol [1] as well. Due to its unique features it can be used as a base to build a new hidden data transfer method in WSN networks, which has been widely discussed later in this article.

III. TAG The Aggregation Protocol

One of the most popular solutions for data aggregation in Wireless Sensor Networks is a TAG protocol. The protocol is designed for use in monitoring application such as [2]. A main feature of these networks is the socalled cyclical sending of significant measurement results by source nodes. In the TAG protocol it is done by using a simple queries and answers which are spread in the WSN. The important part of the protocol is idea of retransmitted data processing called aggregation. The data aggregation is done in intermediary retransmitting nodes, and is to reduce the amount of data outgoing towards destination. The usage of aggregation results in a reduction of data traffic sent in the network.

TAG is characterized by three main features:

(1) Offers a simple and transparent way for describing the type of data collected, the scope and the method of their aggregation. Through simplified SQL query sink node programs available sensors in the network to work in a certain way.

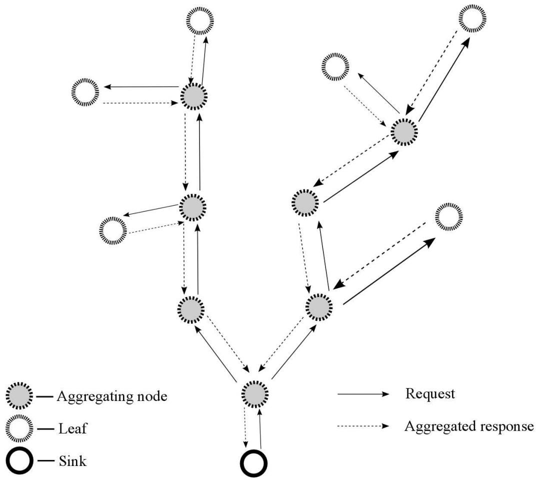

Fig. 1. The tree structure of the aggregation network

(2) a unique way to control the network topology, which allows for the queries distribution in the sensor network.

(3) a nodes synchronization system is done at the application level. It solves the problem of waiting time for data aggregation nodes caused by delay from nodes at the higher level at the network. This issue is particularly important and is discussed in detail in this paper.

The aim of the TAG protocol is to use a source node with higher performance in relation to the rest of the sensors. The source node is responsible for organization of the network topology of aggregation nodes. The typical aggregation network is composed of: source nodes, aggregation nodes, and a sink node. The source node is producing data. The aggregation node is an intermediary in the transmission. Aggregators can also be used as the source node. Sink nodes distribute queries to sensor in the networks and collect data.

The TAG use a tree network structure. The simple example of the tree structure was presented in Figure 1. The sensor data delivery consists of two phases: in the first phase the query is sent by the sink node to the entire network of sensors. The query contains information about which sensor readings to be transmitted e.g. temperature or ambient volume and how it should be aggregated e.g. average temperature, total temperature, the highest temperature etc. In the second phase data are transferred to the sink node.

The example of the TAG protocol usage may be counting the number of nodes in the network. In the case of TAG protocol failure, each node would send a message to the sink node by using the basic type of flooding routing method [3]. In this case, the sink node have to count separately each node for analyze its movement. Using TAG protocol the sink node receives a score, which was calculated by aggregating nodes in the network. Such solution disperses the task of the sink node to the entire network, and reduces network traffic. The example of the aggregation network is shown in the Figure 1.

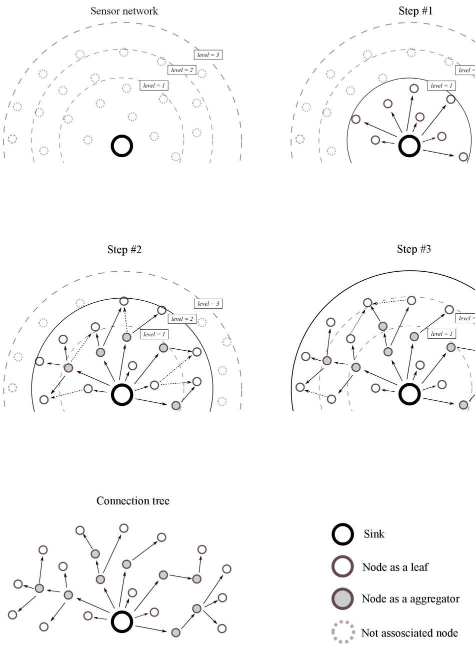

Fig. 2. The tree structure creation process

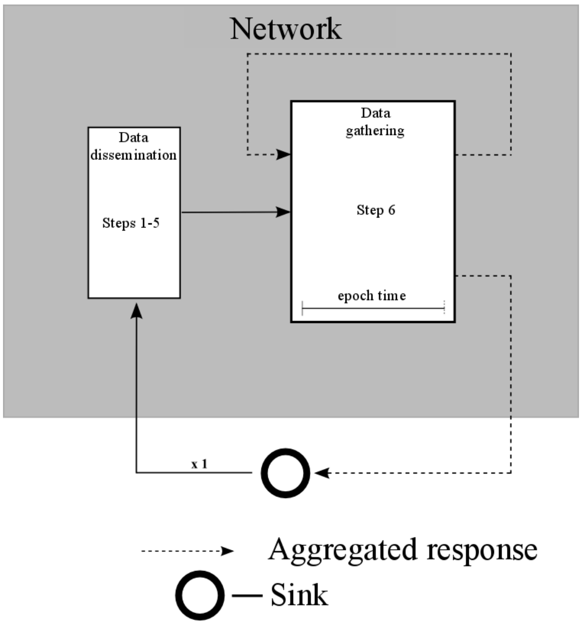

Fig. 3. The communication scheme nodes in TAG

The TAG aggregation protocol is based on a tree based structure on application level. In principle, for communication in lower layers can be used any routing protocol that meets the following requirements: (a) the ability to deliver messages to each node in the sensor network, (b) the usage at least one physical path to data delivery to the sink node, and (c) the guarantee of usage the only one copy of the message without any duplication.

The tree based topology is based on sink node. In order to start the process of aggregation in WSN the necessary is to create a tree links between the sink node and all others sensor nodes. For this purpose, the base station sends a broadcast message through the entire network. The message contains, among others, parameters which specify the unique node identification number ID, and the distance between sink node and sensor source nodes, a socalled level of the node. The sink node has always level 0, first nodes that receive the message given by sink nodes set the level 1 etc. The process of increasing the level and assign the next ID will be repeated until each node within the network receives the message at least once. The topology creation process is cyclic, which allows nodes to update the network topology. The example of the network tree creation process is presented in Figure 2.

After determining the sensor sources, it starts the data collection phase. At this stage, nodes send the value of their measurements, which are then subjected to the aggregation process. The main goal of TAG protocol is to send the smallest possible number of messages, therefore the data collection phase was divided into time slots, which depends on the level number. The greater level parameter causes the longer duration of the distribution phase. The duration of the distribution phase is called the epoch. The time slot during the epoch should be appropriate to ensure the aggregator node enough time to receive a message from the higher level of nodes, process the message and then give the result to the lower degree nodes.

The Figure 3 presents the basic scheme of communication in the TAG protocol. It looks as follows:

(1) The sink node sends a broadcast message to all nodes that are within its range. This message contains the content of the SQL request, called r, and the level parameter.

(2) At the moment at which node p receives request r the synchronization procedure is activated. The procedure synchronize the clock of the node with clock of the sender of the message [4].

(3) Node p selects a sender of the message r as its parent which acts as an aggregation node. Message r contains the maximal time for which node p must upload its own and received data.

(4) Node p retransmits received message. The level parameter is increased and the maximum time to wait for a reply parameter is reduced.

(5) The process of retransmitting messages r and its modification is repeated by all nodes in the network.

(6) After receive the message in all nodes in the network, the data collection phase begins. Each node waits for a new message from its parent node. The data with are received before the maximum waiting time are subjected to a process of aggregation and then are sent to its parent. The process of sending messages by nodes is repeated until the message reaches the sink node.

IV. Methods of Data Encryption in WSN

Data encryption in WSN can be done based on cryptographic and hiding techniques. The cryptographic techniques are the most commonly used in practice. Techniques are based on the public key. The typical example of application of this technique in WSN is work [5], where there are used the pair of different interrelated cryptographic keys. Usually, one of keys is public available and the other one is private and it is only known by the owner. Using cryptographic methods source sensor node can encrypt the message using already sent to public key of the recipient. Only the recipient who knows its private key can decrypt transmitted message.

Based on cryptographic techniques can be analyzed the compatibility of the received data, especially in the process of aggregating data. Hash functions usage prevents spoofing by unwanted nodes in existing aggregation network. The example of hash function methods is described in work [6]. The data encryption methods make impossible to read the secret message, but cannot hide the fact of sending a message. Therefore an object of the second method is to hide the message. The main task of the second method is not the encrypt the message, but to hide the fact of transmission. The main idea is to not arouse the observer suspicion of the network. The hidden message is often implicated in the data stream.

The message hiding is divided into three groups [7]. The first one is based on a modification of the message. In this idea the content of the packet is modified. The modification can be done in the header of the message, in the data field, or in the header and data field at the same time in one message. The message modification group is characterized by a relatively large bandwidth for the only headers modification. The drawback is the ability to easily detect the modifications. The modification of the data field is much more effective. The modification detection is difficult, but the cost is the relatively low bandwidth. The method is used in photo watermarking. Third message modification, simultaneous the header and the data field modification is harder in the implementation but is very effective and difficult in detection.

The other group of the data hiding is stream modification. The stream modification is a method in which in order to create a hidden channel is managed the sequence of packets sent is managed. The solution applies expedient loosing packets and intentional controlled delays between packets. The stream modification solution requires synchronization between source and sink of data, has smaller data bandwidth then in data content modification, but is easy in implementation.

The last type of message hiding is a combination of previous groups. The hybrid modification is a group of the method in which modification techniques are used simultaneously. The group is characterized by a good bandwidth effectiveness, very difficult hidden message detection.

V. TAGCC The Hidden Channel Aggregation

A. Assumptions

The TAGCC protocol is a modification of the original TAG which focuses on the hiding messages based on the cyclical nature of measurements and reports. The proposed modification uses a hidden channel for detecting unwanted objects, which have the ability to eavesdrop on messages sent by the nodes that are within radio range.

The basic assumptions for TAGCC hidden channel are: (a) the solution is a hybrid modification of the message hiding technique, (b) the solution is based on the data aggregation protocol commonly known as TAG, (c) moreover we assume no message encryption, (d) and no aggregation process for hidden message frames. Additionally we assume (e) no increased need for computing power of each sensor node in the network and (f) active use of network delays to hide data as the natural behavior of WSN aggregation network. A characteristic feature is (g) the use of existing sensor network traffic and lack of additional packets. The comparison of the basic idea TAGCC message hiding and classical encryption method is presented in the Figure 4.

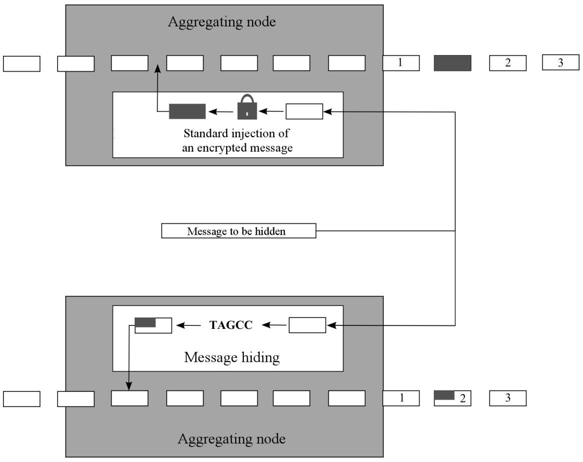

Fig. 4. The idea of TAGCC message hiding and classic encryption method

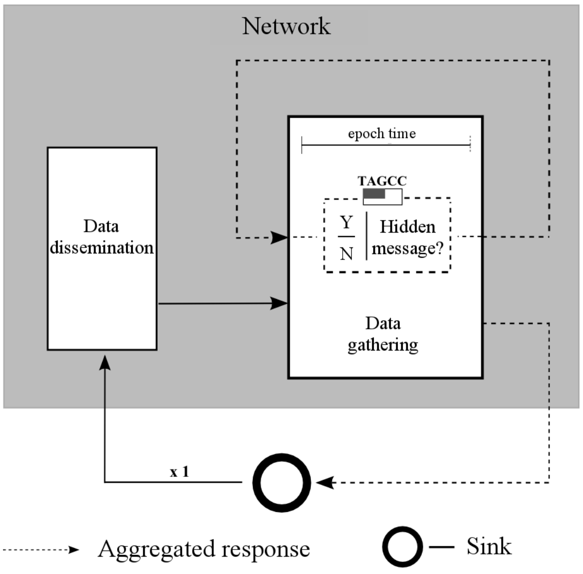

Fig. 5. TAGGCC and operating phases TAG aggregation protocol

B. The TAGCC Protocol

The TAGCC protocol is used in the WSN during the data gathering phase. As long as there is no need to send hidden data, WSN sensors work like a normal aggregation network according to the principles TAG protocol rules. When sending a secret data is actuated TAGCC protocol according to the scheme shown in the Figure 5.

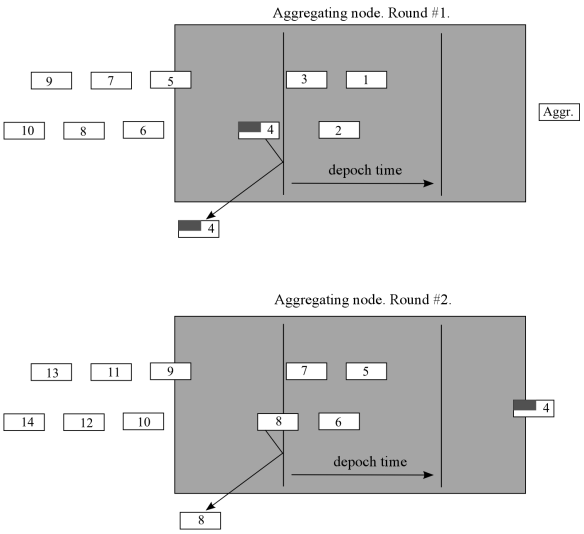

Fig. 6. The TAGCC protocol acceptance or rejection of received packets

One of the basic idea in TAGCC protocol is a way to send hidden messages. The hidden message propagates between two levels of the sensor network within one epoch. As was mentioned earlier, a hidden transmission idea is based on the control of the packets delays. Packet delays are the natural phenomenon caused by busy medium and possible retransmissions need of lost packets in a randomly generated backoff time.

Nodes involved in the data transmission receive messages at a specific time slots. The protocol assumes that packets received by recipient with delay due to retransmission or busy radio channel are rejected. The example of the acceptance or rejection of received packets is presented in the Figure 6. The TAGCC protocol uses the situation where there is a receiving delay. The proposed method is hiding data packets deliberately delayed placing in them a hidden message. The recipient read the delayed message despite the fact that the message will be rejected. Therefore the intentional delay is appropriate channel control method for hidden messages. The content of the hidden message is appended to the message packet in the open form and it arouse no suspicion as it is one of the many delayed packages.

The work of the TAGCC algorithm is divided into the following phases:

(1) The start of the standby TAGCC phase in data collection phase.

(2) The phase of waiting for the event, which is composed of two tasks:

(2.a) The receive a hidden message. The goal of the message receive process is to analyze delayed packets that arrive to the receiver. The packet is considered as delayed when it is received by the node from its child after the allocated time called depoch. The node using the TAGCC algorithm analyzes each delayed packet for the data field tag called Identifying Sequence (IS). The IS tag calculation is done based on the formula 1. If delayed packet contains the IS tag, it means that the message contains confidential information and should be treated as a hidden message. The content of the hidden message is located in payload before the IS tag, and its length should not exceed a length of no hidden message. An example of the hidden message with IS tag is presented in Figure 7. Hidden messages are stored in an internal queue. The reason is, that the node can receive more than one hidden message per epoch. If the node is not a sink node and the queue contains only hidden messages just received, the node performs data aggregation and send as in TAG protocol. After complete this tasks the algorithm starts step 4.

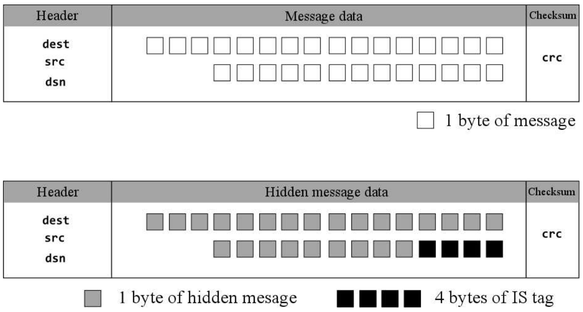

Fig. 7. An example of the hidden message with IS tag

(2.b) The second task is generating the secret content. The step is made when the sensor source node decides to initiate sending hidden messages to the sink node. For this purpose, the packet is generated as is presented in Figure 7 and then algorithm starts the step 3.

(3) After the secret contents generating, the packet with hidden data is transmitted. It is done after the nearest depoch interval. The node immediately blocks their own regular hidden free message. Instead of the correct measurement message it is sent a intentionally delayed hidden message. The delay time is defined by apart of message the ccdelay parameter.

(4) A node internal hidden data queue is handled once per epoch. Therefore the hidden message is transmitted in the next epoch after the round of written to the queue. At the next epoch the node immediately locks the regular hidden free message send. Instead of regular hidden free message, it is sent the intentionally delayed hidden message retrieved from the queue. As in step 3, the delay time is defined by ccdelay parameter.

(5) If the node is a sink, it listens only for the arrival of hidden messages. The sink node also controls the depoch intervals. Any message that is received after the prescribed depoch will be analyzed for IS marker presence. The plain text of the hidden message is in the packet payload omitting the last 4 bytes of data.

Each TAGCC node shares a secret key SK and any hidden message is marked with a value of IS:

$$IS = H(SK|src|dsn) \tag{1}$$

where: SK is a secret key which is known by source nodes and sink node, Src is a source node address, Dsn is a sequence number of transmitting node, H is a hash function,

The node calculates IS tag during the hidden message movement. It is done during receive and transmitting of the message. In the case of transmitting hidden messages to calculate IS tag the node gets own parameters of the src and dsn. The important is, that every new hidden message transmission triggers a new IS tag value recalculation. The new IS value is updated, but the message content is unchanged.

C. An TAGCC Example

For the purpose of the TAGCC example we assume a network composed of 9 sensor nodes and one sink node. The network consists of 4 levels deep which employ 5 aggregators and 4 sensor source nodes. The goal of the system is to count the number of nodes each round. Thus, for the network without any losses the final result should be equal to 10. Due to the node initiating the transmission of the hidden message is located in 4th level, the packet will propagate through the network for 4 rounds.

An example of the network in first round is illustrated in Figure 8a. In the first round node I hide the message in delayed packet. Node F receives only the correct message from the node J. Node F does not modify their work, and only saves a hidden message in the sending queue for the next round. Node F aggregates only the value of its junction with the value of J, since the value of the node I was not received. The work of rest of the nodes does not change. The sink node receives at the first round the score of 9, and is not receiving a hidden message.

During the second round, which is depicted in Figure 8b, node F receives data to the aggregation from nodes J and I. This data will be omitted because of the hidden message in the node F queue. Node F calculates its own value of IS tag and it sends hidden message in a delayed manner to the node E. In round 2 the data for aggregation from nodes I, J and F are skipped. Node E after successfully receive delayed message saves its hidden content in the queue. The sink node receives a second round score equal 7, but still not receives a hidden message.

The next, 3 round is illustrated in Figure 8c. Nodes not involved in the assignment of the total number of sensors are marked in red. Nodes H, J, I, G and F for the external observer are transmitting data in proper manner without any hidden messages. Node E has hidden message in the queue, and it starts to sending process to his aggregator B. In parallel node E is converting their value of IS tag. Node B after successful received a delayed message from node E, starts to save the hidden content in his own queue. In the second round the sink node receives a total score eqal 4, but it does not received a hidden message. The value of the total score was influenced by those received from nodes B, C and D only. Level 3 and level 4 messages are rejected at level 2 in node E.

The round 4 is the last part of the example (see Figure 8d). Nodes H, I, J, G, F, E and B are those, which aggregated values of the counter was not delivered to the sink node. Correctly received values are derived from nodes C and D only. Node B recalculates its own IS tag value and sends in a delayed manner hidden message to the sink node. Nodes H, J, I, G, F, E, D, and C are still working properly from the external point of view. Deliberately delayed message occurs only once and in one place in the network. Due to the node B is aggregation node at the low level of the network, the failure in aggregation value results in significant reduction TAG performance. This is due to deliver a hidden message to the sink node. The sink node receives score value equal 3 in the final round.

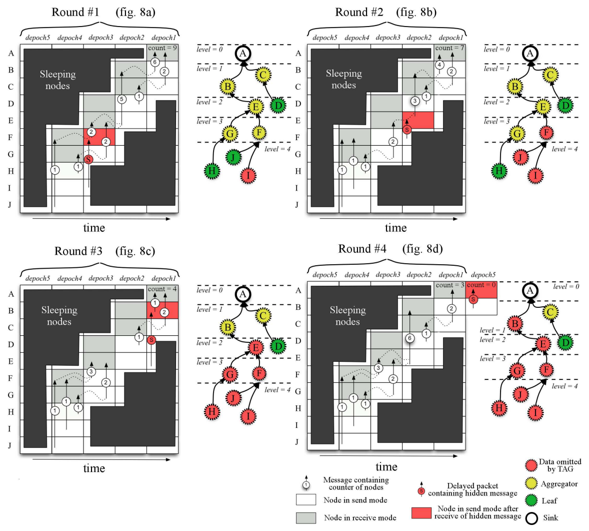

Fig. 8. An example

VI. The Protocol Simulations

In this chapter we present the simulation tests TAG and TAGCC efficiency. At first, we conducted basic research protocol TAG property. This is mandatory to specify the conditions under which messages hide process is successful. Based on this we conduct test of the property of TAGCC protocol.

A. The Core Loss Ratio of the TAG Protocol

For aggregation protocol the main important property is the core loss which is depending on the basic protocol parameters. The core loss factor is measured at the level of aggregation. It is the factor which is based on the final result of the aggregation in the network depending on the main parameters that control the TAG.

During the tests we analyze the impact of the duration of one round of epoch and the duration of the interval depoch for the final aggregated result in the sink node. For this purpose we propose the network with implemented TAB mechanism operating in the mode of simple aggregation. This means that the sink node, after sending out requests for a total sum of nodes, is receiving periodically each epoch reports with results.

The core loss ratio is the level of packet loss that is composed as aggregated number of nodes which is received in the sink node to the total number of nodes participating in the aggregation process.

The simulation results was presented in Figure 9. We made simulations for four different number of nodes in the network: 50, 100, 200 and 300 depending on the depoch and epoch parameters. We observe, that the network core loss less is achieved only within a certain range of epoch and depoch parameters. For very short times of depoch time results in ineffectiveness, the losses are significant because nodes are not waiting to receive all messages. The increase of the duration

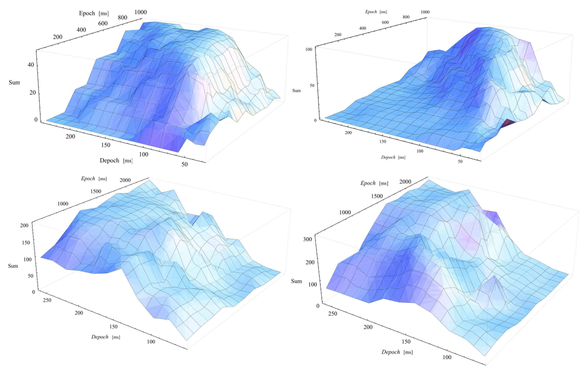

Fig. 9. The core loss ratio of the TAG protocol for 50, 100, 200, 300 nodes in the network

TABLE I
TAGCC SIMULATION PARAMETERS

Name	Values
Networks	300, 60, 4
noise models	meyer heavy noise
sim times	100
epoch times	1800
depoch times	220
rand backoff time	60
ccdelay	210

improves the aggregation network, but too long depoch time causes that all nodes do not participate in the aggregation process.

We also observe a significant increase in packet loss with increasing the number of nodes in the network. This is related to increasing collision probability for the transmitted packets in the network and the increase time of the channel occupancy. All retransmissions of packets are added up and causes a larger delay. The increased number of aggregation levels also enhance this problem. The best results for 200 and 300 nodes networks are obtained for depoch time around 200ms and an epoch in the range 1500 2200ms. This parameters are used to investigate the properties of the TAGCC protocol.

B. TAGCC Simulations

After determining the characteristics of the TAG protocol TAG, we started testing the TAGCC method. All simulation parameters are presented in Table 1 below.

We made all simulations with the epoch and depoch parameters which are selected for the highest effectiveness of the TAG protocol. In our opinion, the use of TAGCC at low packet loss and high efficiency aggregation allows for stealth transmitted data.

C. Delays in TAGCC

One of the main parameters for hidden messages protocol examination are delays. The delay show proper operation TAGCC channel. We obtain results which are consistent with the assumptions. The total delay includes not only time between iteration, but also the first depoch interval for message first time sent. The results we presented in Table 2.

TABLE II
THE SUM OF THE DURATIONS OF ALL THE ROUNDS EPOCH DURING WHICH A MESSAGE IS SENT

	Epoch 0	Epoch 1	Epoch 2	Epoch 3
Delay TAGCC [s]	0,0	0,5	2,5	4,5
	Epoch 4	Epoch 5	Epoch 6	Epoch 7
Delay TAGCC [s]	6,5	8,5	10,6	12,6

TABLE III
SUMMARY OF OPERATING DATA AGGREGATION PROTOCOL TAG AND COVERT CHANNEL TAGCC

	Epoch 0	Epoch 1	Epoch 2	Epoch 3
Packets sent	299	299	299	299
Packets received	298	298	294	297
TAGCC packets sent	0	1	1	1
Amount of nodes	300	300	300	300
Sum of TAG	299	297	255	254
Efficiency of TAG [%]	99,7	99	85	84,7
Packet loss [%]	0,3	0,3	1,7	0,7
	Epoch 4	Epoch 5	Epoch 6	Epoch 7
Packets sent	299	299	299	299
Packets received	297	296	296	297
TAGCC packets sent	1	1	1	1
Amount of nodes	300	300	300	300
Sum of TAG	242	219	176	149
Efficiency of TAG [%]	80,7	73	58,7	49,7
Packet loss [%]	0,7	1,0	1,0	0,7

D. The TAGCC protocol influence for TAG

As we stated earlier the TAGCC protocol uses TAG protocol, and for undetectable message transmission should not change their characteristics. Therefore for the proper protocol evaluation important are metrics based on transmission parameters. For this reasons we chose the total loss and packet loss parameters. The first one specifies the loss of the TAG protocol work, the second one the data packet loss in the whole simulated network. Both parameters we express as a percentage ratio is presented in Table 3.

The simulation results that we obtained are in line with our expectations. The TAGCC protocol is not influencing with TAG network and is not generating additional packet loss. The TAG protocol work without TAGCC shows a comparable level of packet loss. The TAGCC hidden channel slightly affects for TAG aggregation performance degradation. In order to keep the secret messages flow through the network, the aggregated data transmission is interrupted for one round epoch, so as to be able to deliberately delay the message with the secret data. This delay gives the impression of a natural operation nodes in a sensor network, characterized by delays in the transmission channel.

For simulated network, we obtained results in which the TAGCC channel bit rate is 15 bits/second, for the assumption that the TAG network node is sending 29 b/s in the data field of the package, therefore TAGCC hidden channel carries 25 bytes of data within 12.6 seconds.

VII. SUMMARY

The aim of this paper was to present a modification of the TAGCC protocol with hidden channel method designed for use in Wireless Sensor Networks. The proposed mechanism allows to send secret messages hidden in the network traffic. It is done by nodes which are participating in the data aggregating process. The essence of the solution is no additional data packets for the secret messages sending, unlike to the encrypted channels by cryptographic techniques. Hidden TAGCC channel provides a secret messages delivery that is invisible for external network observer.

This paper presented a description of the TAG aggregation network protocol with TAGCC data hide method. Hidden TAGCC channel can be used independently of the operations performed on data in the protocol TAG.

References

[1] S. Madden, M.J. Franklin, and J.M. Hellerstein, ”TAG: a Tiny Aggregation Service for Ad Hoc Sensor Networks,” *5th Annual Symposium on Operating Systems Design and Implementation (OSDI),* December 2002.

[2] V. Gupta and R. Pandey, ”Data Fusion and Topology Control in Wireless Sensor Networks,” *Wseas Transactions On Signal Processing,* ISSN: 1790 5052, Issue 4, Volume 4, April 2008.

[3] K. Holger, A. Wilig, ”Protocols and Architectures for Wireless Sensor Networks,” John Wiley Sons, Ltd., 2005.

[4] S.R. Madden, M.J. Franklin, J.M. Hellerstein, W. Hong, ”TinyDB: An Acquisitional Query Processing System for Sensor Networks,” *ACM Transactions on Database Systems (TODS),* Volume 30 Issue 1, March 2005.

[5] D. Martins, H. Guyennet, ”Attacks with Steganography in PHY and MAC Layers of 802.15.4 Protocol,” *Fifth International Conference on Systems and Networks Communications,* 2010.

[6] S I Huang, S. Shieh, J. D. Tygar, ”Secure encrypted data aggregation for wireless sensor networks,” *in Wireless Networks,* pp. 915 927, 2010.

[7] W. Mazurczyk, M. Smolarczyk, K. Szczypiorski, ”RSTEG: Retransmission Steganography and Its Detection,” 2009,

[8] TinyOS, http://tinyos.stanford.edu/tinyos wiki.

[9] W. Mazurczyk, J. Lubacz, ”LACK a VoIP Steganographic Method,” *Telecommunication Systems: Modelling, Analysis, Design and Management,* Vol. 45, Numbers 2 3, 2010.

[10] C. Turner, ”A steganographic computational paradigm for wireless sensor networks,” *IIT09 Proceedings of the 6th international conference on Innovations in information technology,* 2009.

[11] S I Huang, S. Shieh, J. D. Tygar, ”Secure encrypted data aggregation for wireless sensor networks,” *in Wireless Networks,* pp. 915 927, 2010.

[12] F. Amin, A.H. Jahangir, H. Rasifard, ”Analysis of Public Key Cryptography for Wireless Sensor Networks Security,” *World Academy of Science, Engineering and Technology,* 2008.

[13] Y. Yao, J. Gehrke, ”The Cougar Approach to In Network Query Processing in Sensor Networks,” *ACM SIGMOD Record,* Volume 31 Issue 3, pp. 9 18, 2002.

[14] J. Ding, ”Design and Analysis of an Integrated MAC and Routing Protocol Framework for Large Scale Multi Hop Wireless Sensor Networks,” *Technical Report,* Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore, July 2002.

[15] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, ”Energy Efficient Communication Protocol for Wireless Microsensor Networks,” *Proceedings of HICSS 2000,* pp. 4 7, January 2000.

[16] S. Lindsey and C. S. Raghavendra, ”PEGASIS: Power Efficient Gathering in Sensor Information Systems,” *Proceedings of IEEE ICC 2001,* vol. 3, pp. 1125 1130, June 2001.

[17] S. Lindsey, C. S. Raghavendra, and K. M. Sivalingam, ”Data Gathering Algorithms in Sensor Networks Using Energy Metrics,” *IEEE Transactions on Parallel and Distributed Systems,* vol. 13, no. 9, pp. 924 935, September 2002.

[18] A. S. K. Pathan, H. W. Lee, and C. S. Hong, ”Security in wireless sensor networks: Issues and challenges,” *CoRR,* vol. abs/0712.4169, 2007

[19] L. S. Mehta A.M. and P. K., ”Steganography in 802.15.4 wireless communication,” *in Advanced Networks and Telecommunication Systems,* 2008. ANTS 08. 2nd International Symposium on, (Mumbai), pp. 1 3, 2008.

[20] T. Kho, ”Steganography in the 802.15.4 physical layer,” tech. rep., 2007.

[21] Y. Zhou, Y. Fang, and Y. Zhang, ”Securing wireless sensor networks: A survey,” *IEEE Communications Surveys and Tutorials,* vol. 10, no. 1 4, pp. 6 28, 2008.

[22] G. Padmavathi and D. Shanmugapriya, ”A survey of attacks, security mechanisms and challenges in wireless sensor networks.” *CoRR,* abs/0909.0576, 2009.

[23] J. Zhou, ”Efficient and Secure Routing Protocol Based on Encryption and Authentication for Wireless Sensor Networks,” *International Journal of Distributed Sensor Networks,* vol. 2013, Article ID 108968, 17 pages, 2013. doi:10.1155/2013/108968

[24] J. Shao, N. Ye, ”A Spanning Tree Algorithm for Data Aggregation in Wireless Sensor Networks,” *World Congress on Intelligent Control and Automation,* June 25 27, 2008.

[25] W. Yuan, S.V. Krishnamurthy, S.K. Tripathi, ”Synchronization of Multiple Levels of Data Fusion in Wireless Sensor Networks,” *Global Telecommunications Conference,* 2003.

[26] J. Rutkowska, ”NUSHU, Implementation of Passive Covert Channels in the Linux Kernel,” 2004,