BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **23**, No 1
Sofia • 2023 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2023-0004

Competent Time Synchronization Mac Protocols to Attain High Performance of Wireless Sensor Networks for Secure Communication

Ismail Hababeh[1], *Issa Khalil*[2], *Rizik Al-Sayyed*[3], *Mahmoud Moshref*[4], *Samer Nofal*[1], *Ali Rodan*[3]

[1]*German Jordanian University, School of Computer Engineering and Information Technology, Amman, Jordan*
[2]*Qatar Computing Research Institute, Hamad Bin Khalifa University, Doha, Qatar*
[3]*The University of Jordan, Department of Information Technology, King Abdullah II School for IT, Amman, Jordan; Corresponding Author*
[4]*Palestine Technical University, Kadoorie, Computer Systems Engineering Department, Tulkarm, Palestine*
E-mails: Ismail.Hababeh@gju.edu.jo ikhalil@qf.org.qa r.alsyyed@ju.edu.jo mahmoud.moshref@ptuk.edu.ps samer.nofal@gju.edu.jo a.rodan@ju.edu.jo

***Abstract*:** *Clock synchronization in the Mac layer plays a vital role in wireless sensor network communication that maintains time-based channel sharing and offers a uniform timeframe among different network nodes. Most wireless sensor networks are distributed where no common clock exists among them. Therefore, joint actions are realized by exchanging messages, with time stamps using local sensor clocks. These clocks can easily drift seconds and cause functional problems to the applications that depend on time synchronization. Time synchronization is a major and challenging factor in wireless sensor networks that needs to be studied and explored. In this paper, we propose integrated time synchronization protocols that serve wireless sensor network applications under normal, secured, and unreliable environments. The proposed protocols are discussed and evaluated based on their accuracy, cost, hierarchy, reliability, and security. Simulation results show that the proposed time synchronization protocols outperform the state-of-the-art techniques in achieving a minimum synchronization time.*

Keywords: *Wireless sensor network, Time synchronization, Reliable synchronization, Secured synchronization, On-demand synchronization.*

1. Introduction

Local time clocks of Wireless Sensor Networks (WSN) are controlled by time synchronization. To guarantee the accuracy of time synchronization, it is essential to correct the time deviation caused by unsteadiness, where this deviation should be preserved from the drift and offset reimbursements [1-5]. Depending on the request, this can be done by two types of time synchronizations, specifically relative and absolute clock synchronization.

In addition, some factors related to resource limitations in wireless sensor networks, such as bandwidth and energy, should also be taken into consideration. Moreover, securing the WSN local clock synchronization is a vital structure block for both secure and non-secure real-time systems, where it has been used for restructuring secure and capable WSN nodes and preventing violations against data privacy and security [1, 6, 7].

Current WSN studies implement a signal processing method that helps in simplifying the evaluation of clock synchronization protocols. To attain high dynamics in the performance of WSN systems, time synchronization needs to be considered to get reliable, energy-efficient, and remote checking in distributed systems [8]. Such dynamic systems generate information from WSNs that can be used to inspect data security attacks and serve as a reference to obtain the right and instant actions. Moreover, in External Gradient Time Synchronization Protocol [9], each node is synchronized based on the information gathered from its base and the neighboring nodes.

In many WSN systems, where the global clock is not shared, a local clock exists in each node that serves and is used to guess the time in other nodes by exchanging time-stamped messages [10]. However, message delivery is not guaranteed. Accordingly, the message delay factors including transmission, propagation, and reception as well as sending and receiving times should be considered when designing WSN synchronization protocols.

Some current time synchronization approaches, such as E-FTSP [11] enhance previous protocols by speeding up time synchronization processes, while others establish new time synchronization algorithms, such as SANSync [12], which are based on clustering. A new pairwise time synchronization approach is the proposed K-Sync [13] where high accuracy can be realized only by utilizing the time-stamps information in the process of message exchange.

WSN asynchronous dynamic response time is detected and corrected based on frequency domain decomposition with frequency-squeezing processing [14]. Most existing time synchronization approaches are tended to achieve high synchronization performance and decrease synchronization time error.

The WSN node clock skew and offset are evaluated by the pairwise synchronization algorithm K-Sync [15] which is based on the Kalman filter to estimate the normalized clock skew and offset of the node for underwater WSN applications. This method achieves high-precision clock synchronization, and it also enhances the robustness of a variety of underwater motion scenes. The integration of the Ultra-Wide Band (UWB) transceivers in the nodes of the wireless networks is used to improve the time synchronization between the WSN nodes [16].

A new Multi-hop Average Consensus Time Synchronization Algorithm (MACTS Algorithm) considers the multi-hop over short distances between nodes [17]. This algorithm is based on a one-way broadcast model that achieves hundreds of time convergences when compared to the Average TimeSync (ATS). A study on a clock synchronization scheme based on the attack detection mechanism, attack compensation, and maximum consensus protocol is presented in [18]. This method is proven theoretically to achieve attack detection correctly, and its effectiveness is validated using several simulations.

A Hybrid Time Synchronization Protocol (HTSP) utilizes the advantages of reference-based and consensus-based approaches [19]. This protocol exploits a temporary reference node to significantly reduce the convergence time, and, on the other hand, to employ the average-based consensus during normal operation to handle node failure.

Authors in [20] have presented a Distributed Time Synchronization Algorithm based on Sequential Belief Propagation (SBP-DTS). This technique establishes a Factor Graph (FG) model for the time synchronization and then uses the sequential Belief Propagation (BP) algorithm to estimate node clock parameters. A new method based on the sequential least square algorithm [21] is introduced to estimate the skew under delays and improve the accuracy while reducing the memory requirement. This method theoretically and empirically proves that it outperforms existing consensus-based synchronization schemes under random delays in different environments.

The rest of the paper is organized as follows: a detailed literature review of traditional and state-of-the-art synchronization protocols is presented in Section 2. The proposed time synchronization protocols are introduced in Section 3. Experimental results analysis and the performance evaluation of the proposed approach are presented in Section 4. Finally, conclusions and future research directions are discussed in Section 5.

2. Related work

A local clock synchronization time in WSN nodes can be accomplished by using the pairwise broadcast synchronization protocol [22] in terms of exchanging packets among its neighbors while preventing sending packets from the node itself, where this procedure can minimize the required energy for time synchronization.

In a dynamic-time synchronization technique [23], a Kalman filter is used to achieve clock synchrony in a centralized wireless network synchronization system. This filter minimizes the time error threshold to avoid synchronization repetition in systematic processes. However, this technique has not been tested on real-time wireless network systems when large numbers of nodes are used to represent different applications at different hierarchy levels.

A distributed synchronization algorithm for wireless sensor networks is presented in [24]. Each node is synchronized by a proportional integral feedback controller based on reference time, clock offset, and frequency deviations. However, this method does not test the multi-hop with a large number of nodes where the nodes and their parents need to be synchronized.

Authors in [25] build the technique of Timing Synchronization Protocol for Sensor Networks based on network time protocol [26] that offers a universal time synchronization over the Internet. In such protocols, a node starts a byway synchronization message exchange with its base node, where its neighbors are synchronized in a pairwise mode. The time deviation between the sender and the receiver is evaluated based on message exchange time stamps. However, this protocol requires a significant amount of energy for message exchange between nodes. Moreover, its method complexity and the less significant scale make it unsuitable for use in WSN.

In the Broadcast Synchronization Protocol method [27], the base node does not need to exchange synchronization time packets with other WSN nodes. This protocol gets rid of the send and access times, which is used to remove the uncertainty of received time and form an accurate relative timescale. This method is appropriate for applications where a small amount of synchronization energy is required. Though, a physical transmit channel is required for such a protocol.

Authors of Flooding Time Synchronization Protocol [28] assume that closer nodes to each other have strong time synchronization than the nodes that are far away from each other where large time synchronization error occurs. In this time synchronization protocol, the information proliferation is stable because each node waits for a certain time to broadcast its time information to the base node. Nevertheless, this protocol needs an incline time property or zone time.

The Gradient Time Synchronization Protocol introduced in [29] is a distributed time synchronization protocol with the slope time property. In this protocol, each node transmits occasionally a position point message containing its global time. To reduce the time synchronization error, the protocol uses a small-fixed propagation period. However, this approach will considerably consume the sensor’s energy.

In a security-oriented synchronization field, an Attack Tolerant Time Synchronization Protocol [30] is used as a joint interference and recognition system where sensor nodes face security attacks by adjusting their neighbor nodes' typical time synchronizing actions. This protocol implements an unordinary time synchronization recognition method to realize different types of security attacks. As scattered time synchronizations are used, small interference caused by the attacker will be curved out, but this will decrease time synchronization errors.

The Level Diffusion Based Clock Synchronization protocol [31] presents the synchronization of all sensors’ clocks without depending on any configuration conditions in dynamic system settings. Even though this protocol has a wide transmission range, it suffers from much overhead and less accuracy.

Authors in [32] present the Level Based Time Synchronization technique that is intended to launch universal timescales for all WSNs by supporting a secure clock time synchronization mechanism. In this protocol, even if some up-normal sensors cooperate to interrupt clock synchronization, each normal sensor can still synchronize its local clock based on its base node. However, the security procedures are not clearly stated in this approach.

In the Energy Efficient Synchronization field, Energy Gradient Time Synchronization Protocol [33] is introduced to optimize energy saving for applications by selecting suitable message transmitting periods. It is designed to realize the universal time agreement by using effective flow reimbursement and increasing average evaluation. In addition, this protocol saves much energy by tuning the transmitting period without decreasing the time synchronization precision. Nevertheless, this method does not show high performance for collecting data from all WSN nodes.

In this context, Multi-Hop Time Synchronization Scheme [34] minimizes the sensor energy consumption and extends sensor life [34]. This method decreases the time synchronization errors and the overhead by minimizing the number of subtrees. However, it suffers to achieve a high level of performance since it does not attain consistent data collection performance from all WSN nodes.

The temperature Compensation Algorithm [35] is proposed as an adaptive synchronization algorithm to permit consistent use of time synchronization for sensors that share common tasks in different temperature system environments. This approach uses master node transmissions to synchronize other nodes and shows little overhead. Though, this algorithm depicts a performance shortage in collecting data from all network sensor nodes.

The Time-Based Synchronization Sleep Model [36] argues the need for correct time drift forecasts because the network sensor does not swap time stamps. This method investigates the issues that depreciate the time synchronization drift rate. Each sensor transmits its timestamp irregularly to all neighbors to decrease timestamp transmissions, and the neighbor nodes adjust their time accordingly. However, this technique does not stretch out the performance required for collecting data from all sensors.

Feedback-Based Synchronization Scheme FBSS [37] is planned to reimburse the clock time drift caused by internal and external interruptions. This scheme is superior to Delay Measurement Time Synchronization Protocol in different time synchronization periods and sensor node configurations. In addition, FBSS outperforms other synchronization protocols, especially when the time-sliced is greater than 60 s. Yet, FBSS does not elevate the performance to the level required for consistent data collection from all sensors.

In a cluster mode, the Cluster Based on Demand Time Synchronization Protocol [38] is based on many sensors that are set to sleep mode most of the time for the configuration of clustering. When the incident is discovered, the time synchronization process is started immediately. This protocol enhances energy efficiency and performance as compared to other protocols such as TPSN and RBS because the interval time of the incident makes a very small overhead effect. As the time interval increases and the incident arise often, more energy is required. Though, the Cluster Based on Demand Time Synchronization Protocol does not support the high data collection performance required from all sensors.

To handle the fault tolerance in time synchronization, the Fault Tolerate Distributed Time Synchronization Protocol [39] uses the sliding window to estimate sensors' reference time and the convergence function to compute sensors' synchronization times. Nevertheless, this protocol does not show the simulation results or the comparisons with other protocols.

In a heterogeneous synchronizing environment, the Heterogeneous Time Synchronization Model [40] is used to considerably reduce the sensors' time synchronization error and boost the flexibility of the sensors' failures. However, this synchronization model does not raise the performance reliability that is required for data collection from all sensors.

Wireless Deterministic Clock Synchronization (WiDeCS) technique [41] presents the delay factors in WSN and clarifies the effect of WiDeCS on the facts of various delays. In addition, WiDeCS offers a solution to achieve an acceptable level of performance needed for collecting data from all sensors. WiDeCS decides the clock offset between the network sensors and their parents. Nevertheless, this scheme is proposed for the master/slave's structural design in cluster networks.

The Distributed Heuristic Synchronization Algorithm [42] presents how to select sensor couples to minimize the number of transmitting synchronization message interactions while tolerating sensors' synchronization. It permits the sensor to synchronize itself based on the information collected from its neighbors. However, for multi-hop networks, there is a lack of protocols to conclude which sensors should be responsible for performing message exchange.

In the Flooding with Clock Speed Agreement protocol FCSA [43], the main goal is to reduce the undesired effect of slow flooding on the synchronization accuracy without changing the propagation speed of the flood. The results show that the synchronization accuracy and scalability of slow flooding can drastically improve by employing a clock speed agreement algorithm among the sensor nodes.

In an extended version of the Flooding Time Synchronization Protocol E-FTSP [11], the authors propose an enhancement of FTSP and evaluate its advantages. They have explained the problem with FTSP caused by the accumulation of jitter and described their improvements. The simulation results prove that the E-FTSP has been enhanced upon the performance of FTSP significantly, especially in large-scale multi-hop networks.

The study of Sensor and Actuator Networks Synchronization Protocol SANSync [12] introduces an accurate time synchronization protocol that allows the clocks of all sensors and actuators in the WSN to be synchronized with a reference clock. SANSync is based on clustering and takes advantage of the large transmission capacity of actuators to increase the accuracy of time synchronization.

The authors of [44] have done a simulation study that compares the suggested strategies to an existing allocation mechanism and demonstrates how the centralized fair allocation model performs better in terms of spectrum usage.

3. Integrated time synchronization protocols

The proposed integrated time synchronization methodology consists of three different protocols namely, Reliable Single Hop Protocol, Secure Single Hop Protocol, and On-Demand Synchronization Mac Protocol. The architecture of the proposed approach is described in Fig. 1 and summarized as follows:

- The nodes of the WSN are connected in a tree topology with the Base station (B) being the root of the tree for better flexibility and scalability. The simplest topology of the WSN is a tree structure where any two nodes can be reached by a single route.
- Each node in the tree has one parent and a set of children.

- The parent is responsible for synchronizing the children.
- The propagation time in the synchronization protocols is ignored.

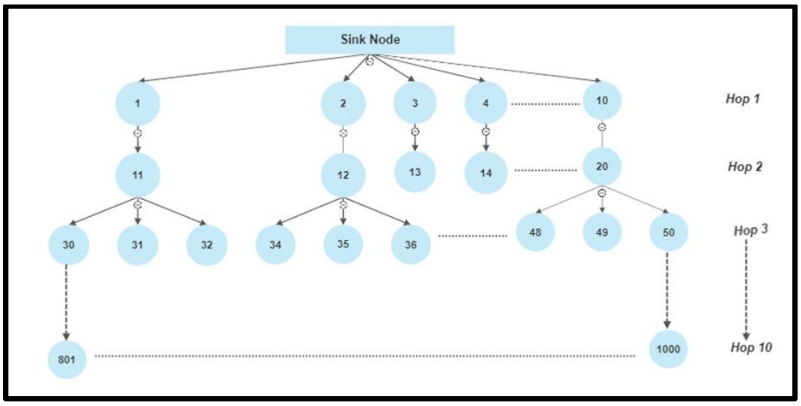

Fig.1. Time synchronization architecture

A novel and reliable Hybrid time synchronization protocol for large-scale WSNs denoted as HTSP that contain a combination of the average-based consensus and reference-based synchronization approaches is proposed in [45]. Each node in HTSP can seamlessly switch between the reference and consensus modes during its operation, and the OPNET simulator is used to model the performance of HTSP. The evaluation results show that HTSP operates efficiently in various types of network topologies. So, achieving reliability in large-scale WSNs is very important to enhance time synchronization between nodes.

3.1. Reliable single hop synchronization mac protocol

In wireless sensor networks, the MAC layer is liable for establishing reliable communication by reducing the radio frequency interference between WSN nodes [46]. It is designed to control the data transfer to a shared channel, providing reliable data transfer services to the upper layers. The MAC layer (sub-layer of the data link layer presented in the OSI model) provides shared link addresses known as Mac Address within the network devices [47]. Hence, the MAC layer resolves the addressing of source and destination stations by encapsulating data frames to be suitable for transmission via the network channels.

In addition, the MAC layer mechanism improves communication reliability by decreasing radio frequency interference in wireless sensor network systems [48]. This mechanism is based on a system called Carrier Sense Multiple Access with Collision Detection [49]. Therefore, a Reliable Single Hop Synchronization Mac Protocol called RSHSMP is proposed in which a reliable single-hop transmission mechanism is implemented.

The base station B synchronizes the nodes one hop away from it. In addition, the nodes at hop h_i synchronize nodes at hop h_{i+1} as illustrated in Fig. 1.

The RSHSMP consists of two phases:

- **Phase 1.** Registering the trigger time of the RF module at both B and the nodes one hop away from it.

- **Phase 2.** Sending the time synchronization data packets from B to the nodes one hop away from it.

The MAC layer [50] is assumed to be reserved for the entire length of the two phases.

We define TS as the time it takes for the nodes that are one hop away from the current nodes to be synchronized. Based on TS, the total time, T, is defined as the time needed to synchronize the whole WSN nodes, and computed as in the equation:

(1) $$T = \sum_{h=1}^{n} \mathrm{TS},$$

where, h is the current hop number and n is the total number of hops.

The RSHSMP processes are described as follows:

- Phase 1 starts when the base node B_i, sends a Registration Synchronization Packet (SyncRPkt) to h_i (the children of B_i). The base node B_i registers the trigger time of its RF module ($t_{B_{i,i}}$) and h_i also registers the trigger time of its RF module ($t_{h_{i,i}}$).
- In Phase 2, node B_i sends a time synchronization data packet (SyncDPkt) to h_i carrying $t_{B_{i,i}}$. Let t_{h_i} be the current time at the node in h_i, then h_i is synchronized as in the equation

(2) $$t_{h_i} = t_{h_i} + (t_{B_{i,i}} - t_{h_{i,i}}),$$

where t_{h_i} is the current time at the node of hop i, $t_{B_{i,i}}$ is the current time of the base node of hop i, and $t_{h_{i,i}}$ is the triggered time of the node at hop i.

```
Input: $B_i$: Base station node of hop $h_i$, $N_i$: List of nodes at hop $h_i$, $H_i$: List of hops in the network
Initialization:
Step 1. Set 1 to $i$
Step 2. Do steps (3-23) until $i$ > number of $H_i$
Step 3. set the registration time at $B_i$ and each of $N_i$ nodes $T_u$ to 0
Step 4. initialize the synchronization time at each of $N_i$ nodes
Step 5. initialize the minimum Synchronization Time $TS_{min}$
Step 6. initialize the maximum Synchronization Time $TS_{max}$
Processing Phase 1 (Triggering Time Registration)
Step 7. Set 1 to $j$
Step 8. $B_i$ triggers its time registration $t_{B,i}$
Step 9. Do steps (10-13) until $j$>number of $N_i$
Step 10. $B_i$ sends (registration synchronization packet SyncRPkt) to $N_j$
Step 11. $N_j$ triggers its time registration $t_{h_{j,i}}$
Step 12. Add 1 to $j$
Step 13. Loop
Processing Phase 2 (Synchronizing the nodes one hop away):
Step 14. Set 1 to $j$
Step 15. Do Steps (16-21) until $j$> number of $N_i$
Step 16. $B_i$ sends (synchronization data packet SyncDPkt carrying $t_{B,i}$) to $N_j$
Step 17. $t_{h_j} = t_{h_j} + (t_{B,i} - t_{h_{j,i}})$
Step 18. $T_u = T_u + t_{h_j}$
Step 19. Update $TS_{min}$, $TS_{max}$
Step 20. Add 1 to $j$
Step 21. Loop
Step 22. Add 1 to $i$
Step 23. Loop
Output: Minimum Synchronization Time ($TS_{min}$), Maximum Synchronization Time ($TS_{max}$)
End.
```

Fig. 2. Reliable single hop synchronization mac protocol

The RSHSMP processes continue with h_1 and its children until the whole network gets synchronized. Each of the registration SyncRPkt and the data SyncDPkt is sent after a back-off time taken randomly from a uniform distribution of minimum and maximum synchronization times [TSmin, TSmax]. This prevents multiple nodes located within the interference range from inhibiting when they synchronize their children. The RSHSMP processes are described in Fig. 2.

3.2. Secured Single Hop Synchronization Mac Protocol

In Secured Single Hop Synchronization Mac Protocol SSHSMP, we assume a single-hop transmission mechanism with acknowledgment messages to secure the processes of time synchronization between WSN nodes. An additional phase has been added to the previous phases of the RSHSMP to ensure sending and receiving acknowledgment security packets between parents and child nodes. In SSHSMP, *hi* nodes send an acknowledgment message to its parent upon receiving the synchronization data packets, where the parent node sets a timer to receive the acknowledgment security packets from h_i. The child that has successfully synchronized with its parent proceeds to synchronize with its subsequent children until the whole WSN node gets synchronized. Finally, the synchronization data packets are sent after a back-off time taken randomly from a uniform distribution between the minimum and maximum synchronization times. The SSHSMP processes are illustrated in Fig. 3.

```
    Input: B: Base station node B, N: List of nodes at hop $h_i$, H: Number of hops in the network
    Initialization:
    Step 1. Do steps (3-23) in the Reliable Single Hop Synchronization Mac Protocol
    Step 2. Set 1 to $i$
    Step 3. Set 1 to $k$
    Processing:
    {Synchronizing upon acknowledgment security packet from the nodes one hop away from B}
    Output: Minimum Synchronization Time ($TS_{min}$), Maximum Synchronization Time ($TS_{max}$)
    End.
    Steps for Synchronizing upon acknowledgment security packet from the nodes one hop away from B:
    Step 4. Do until $i>N$
    Step 5. Node of $h_{i,k}$ sends an acknowledgment security packet (SyncSAck) back to its parent node
    Step 6. Parent node sets a timer (TsyncSAck) to receive the SyncSAck packet(s) from its children nodes.
    Step 7. If parent node does not receive the SyncSAck packet(s) from one or more children within TsyncSAck, it starts the synchronization process again for a maximum of three times.
    Step 8. A child that is successfully synchronized with its parent ignores the repeated synchronization security packets
    Step 9. Add 1 to $i$
    Step 10. Set 1 to $k$
    Step 11. Update $TS_{min}$, $TS_{max}$
    Step 12. Loop
    Step 13. Send the SyncRPkt, SyncDPkt, and SyncSAck packets after a back off time taken randomly from a uniform distribution [$TS_{min}$, $TS_{max}$]
```

Fig. 3. Secured single hop synchronization protocol

3.3. On-demand synchronization mac protocol

The On-Demand Synchronization Mac Protocol (ODSMP) handles situations where a node is out of synchronization. This may happen due to the looseness of the wireless network. If the Synchronization Duration (SD) is, for example, 24 hours, then each node sets a local Timer (TSD) immediately after being synchronized, which initializes after SD time. This process will request the node's parent to be synchronized. In case the parent itself is out of synchrony, it will repeat this process recursively until the synchronization is reached.

If a child node is limited to synchronizing only with its parent, the node that is out of synchrony is considered a root of a sub-tree with all its descendant nodes being out of synchrony. Therefore, it is necessary to avoid the race condition among nodes asking for on-demand synchronization which can be achieved by randomizing the Synchronization Request Packet (SyncRPkt).

We define the duty time cycle as the sum of the wake and sleep times. Let the T_C represent the duty time cycle, so the T_W represents the duration of wake time, and the T_S represents the duration of sleep time. T_C is computed in the equation

(3) $$T_\text{C} = T_\text{W} + T_\text{S}.$$

Assume the maximum clock drift in SD is TD. Then, the node that discovers that it is out of synchrony extends its wake-up period to guarantee to be awake when other nodes in the network are awake. This can be accomplished by:

a) Increasing the wake-up duration by T_D, i.e., when starting the wake-up phase, it continues to be awake for a longer time, then T_W is computed in the equation:

(4) $$T_\text{W} = T_\text{W} + K.T_\text{D},$$

where K is the number of out-of-synchrony durations (the number of SD's).

b) Shorten the sleep duration by T_D, i.e., when starting the sleep phase, continues to sleep for a shorter time, then T_S is computed as in the equation

(5) $$T_\text{S} = T_\text{S} - K.T_\text{D}.$$

The out-of-synchrony node, say X, starts the on-demand synchronization phase, immediately when discovering it is out of synchrony according to the on-demand synchronization mac protocol processes described in Fig. 4.

Input: *P*: Parent node, *N*: List of nodes at hop h_i, *X*: Out of synchronization node
Processing:
On-Demand Synchronization
Output: Synchronizing Out of synchronization nodes
End.
Steps of On-Demand Synchronization:
Step 1. *X* sends a request for synchronization (SyncRPkt)
Step 2. *X* gets synchronization packets from neighbor nodes
Step 3. *X* picks a synchronization packet sent by node *Y*
Step 4. *X* starts on-demand synchronization selection (SyncS) to continue the synchronization with its sender node *Y*
Step 5. *Y* sends Synchronization Duration (SyncD) to *X*
Step 6. *X* uses SyncS and SyncD to compute its local time

Fig. 4. On-demand synchronization mac protocol

To enhance the chances that all nodes in the WSN are synchronized, a synchrony node may synchronize with any neighbor (not necessarily its parent). However, a node only sends the SyncAck packet to its parent. If node *N* with parent *P* happens to get the synchronization packet from another neighbor *M*, then *N* proceeds and synchronizes with *M* without sending a SyncAck packet back to *M*. When parent *P* sends the synchronization packet, *N* sends a SyncAck back to *P* without entering the first two phases since it has already done with *M*.

4. Experimental results and performance evaluation

Analyzing the time synchronization of WSN helps in evaluating the network system performance. Accordingly, we have developed a java editor framework using Eclipse IDE for Windows 10, on a desktop machine with a Core i5 CPU, 2.40 GHz, and 8 GB RAM, based on the normal MAC mechanism of IEEE 802.15.4, where one backoff unit/slot in the IEEE 802.15.4 standard is of 20 symbols. This is done to validate the feasibility of the proposed time synchronization protocols. In addition, the proposed protocols are validated in comparison with four state-of-art time synchronization protocols, namely FTSP, E-FTSP, FCSA, and SANSys. The experiments have been conducted using the WSN time synchronization configuration parameters described in Table 1. The experimental results and the performance evaluation of the proposed protocols and the protocols under comparison are discussed in the following subsections.

Table 1. Time synchronization configuration parameters

Parameter	Value(s)
Number of nodes for small range	5, 10, 15, 20, 25, 30, 35, 40, 45, and 50
Number of nodes for large range	50, 100, 150, 300, 600, 800, and 1000
Average distance between nodes	100 m
Wireless communication radius	300 cm
Network coverage distance	600 m×600 m, 1000 m×1000 m, and 1200 m×1200 m
Clock frequency	1 MHz
Initial clock drift	±50 ppm - ±100 ppm
Synchronization time	20 s
Simulation time	7200 s

4.1. Experimental nodes hierarchy

The initial WSN simulation of the proposed approach of nodes hierarchy consists of one base node and up to 1000 nodes distributed in a random tree structure over a maximum of 10 hops. This distribution is illustrated clearly in Fig. 1 above.

4.2. Validation of the proposed approach

The three proposed protocols RSHSMP, SSHSMP, and ODSMP are applied on multiple WSN nodes in a random tree structure hierarchy described in Fig. 1. The average time synchronization results compared with state-of-the-art protocols FTSP [28], E-FTSP [11], FCSA [43], and SANSys [12] are presented in Table 2.

Table 2. The average time synchronization (ms) ± 0.005 ms of RSHSMP, SSHSMP, and ODSMP compared with FTSP, E-FTSP, FCSA, and SANSys

No of nodes	Protocol						
	FTSP [28]	E-FTSP [11]	FCSA [43]	SANSys [12]	RSHSMP	SSHSMP	ODSMP
10	0.38	0.26	0.018	0.013	0.007	2.45	2.12
20	0.47	0.39	0.072	0.024	0.018	3.37	3.12
50	18.5	11.5	0.58	0.083	0.037	5.27	4.93
100	36.3	26.6	1.25	0.156	0.043	8.20	7.9
200	72.4	51.4	2.22	0.317	0.079	14.5	13.9
400	109.3	60.4	3.55	0.462	0.111	23.6	22.9
500	181.7	73.5	5.64	0.846	0.125	32.7	31.9
600	218.3	82.5	6.78	0.963	0.136	43.7	43.1
800	291.1	93.6	9.34	1.34	0.154	47.6	46.2
1000	363.9	100.4	11.40	1.70	0.239	66.2	65.3

The experimental results of the Reliable Single Hop Synchronization Mac Protocol (RSHSMP) show that the synchronization time of each hop can be achieved in an optimal time (less than 1 ms) which outperforms state-of-the-art protocols and precisely enhances the WSN synchronization time, especially with a large number of hops. Figs 5 and 6 depict the time synchronization performance of the proposed RSHSMP compared with FTSP and E-FTSP, and FCSA and SANAys protocols, respectively.

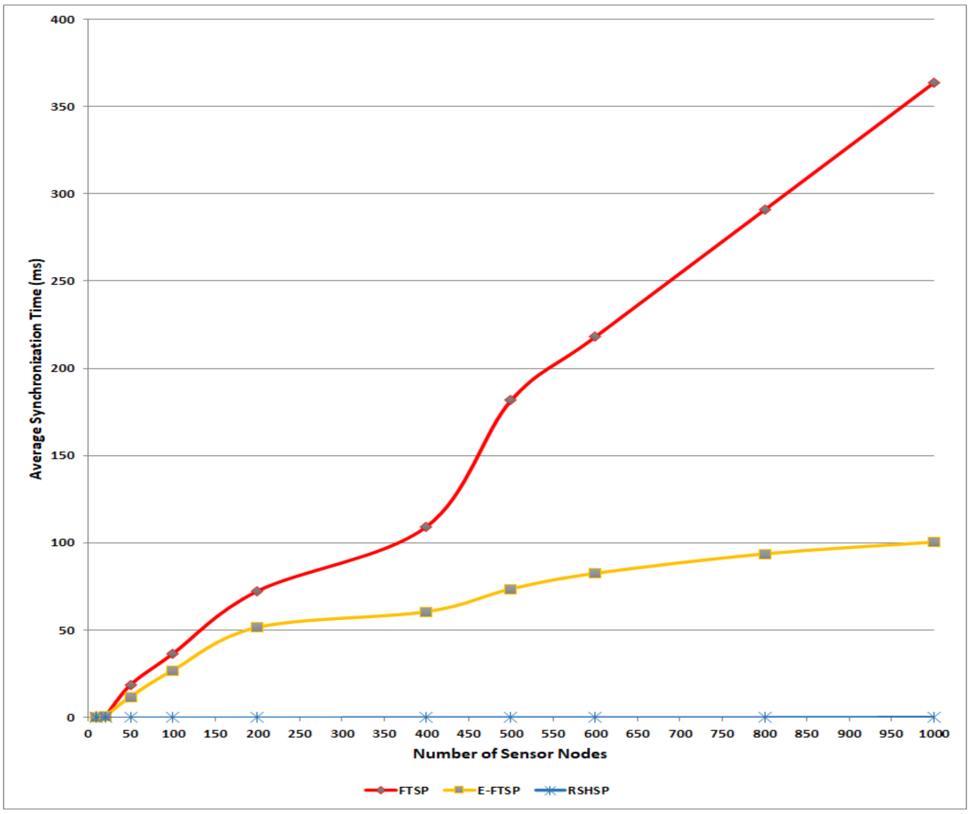

Fig. 5. RSHSMP, FTSP and E-FTSP time synchronization performance

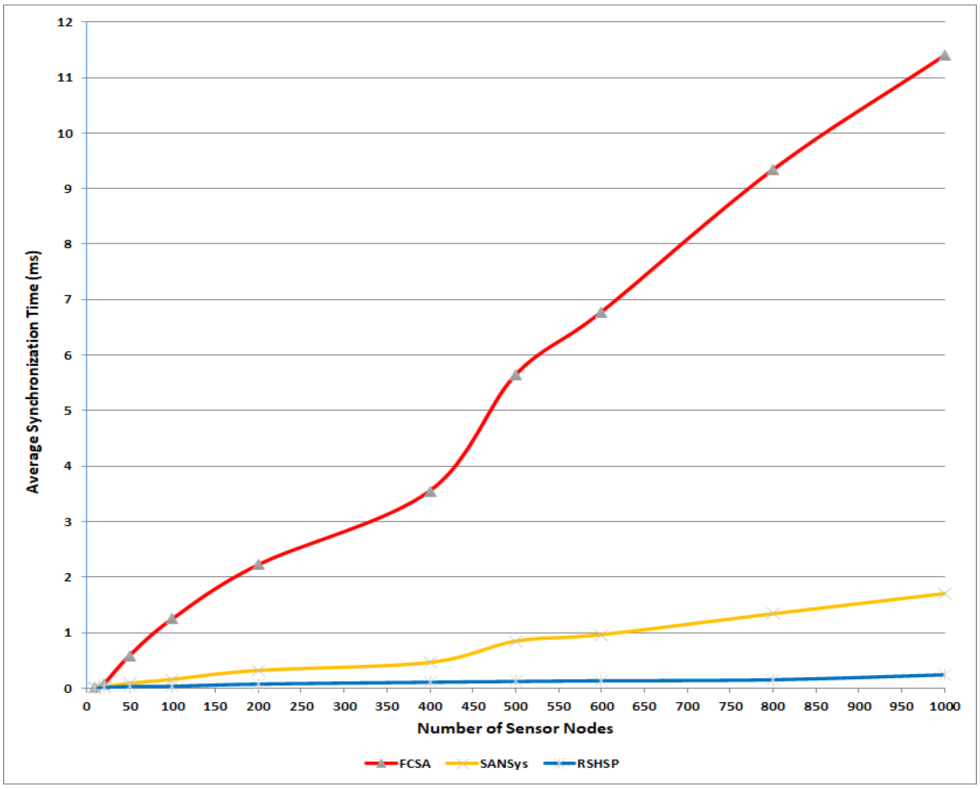

Fig. 6. RSHSMP, FCSA and SANAys time synchronization performance

The Secure Single Hop Synchronization Protocol (SSHSMP) has an additional phase based on the node's acknowledgment to its parent in the hop. This acknowledgment phase is essential in WSNs that support security systems where confidential data are transferred between WSN nodes. In order to support WSN security control applications, the WSN nodes need to receive and send time synchronization acknowledgment packets in order to guarantee sending and receiving secure data packets afterward at the correct time. Therefore, the synchronization time is expected to be increased due to the receiving and sending of acknowledgment messages between WSN nodes and their parents in each hop.

Despite its increasing synchronization time, the SSHSMP proves better performance than state-of-the-art protocols FTSP [28] and E-FTSP [11] as long as the number of nodes in the WSN hops is increased. On the other hand, both protocols FCSA [43] and SANSys [12] do not support WSN security controls, hence they do not receive and send time synchronization acknowledgment packets between WSN nodes; therefore, the time synchronization achieved is less than the proposed SSHSMP. Figs 7 and 8 represent the time synchronization performance of the SSHSMP compared with FTSP and E-FTSP, and FCSA and SANAys protocols, respectively.

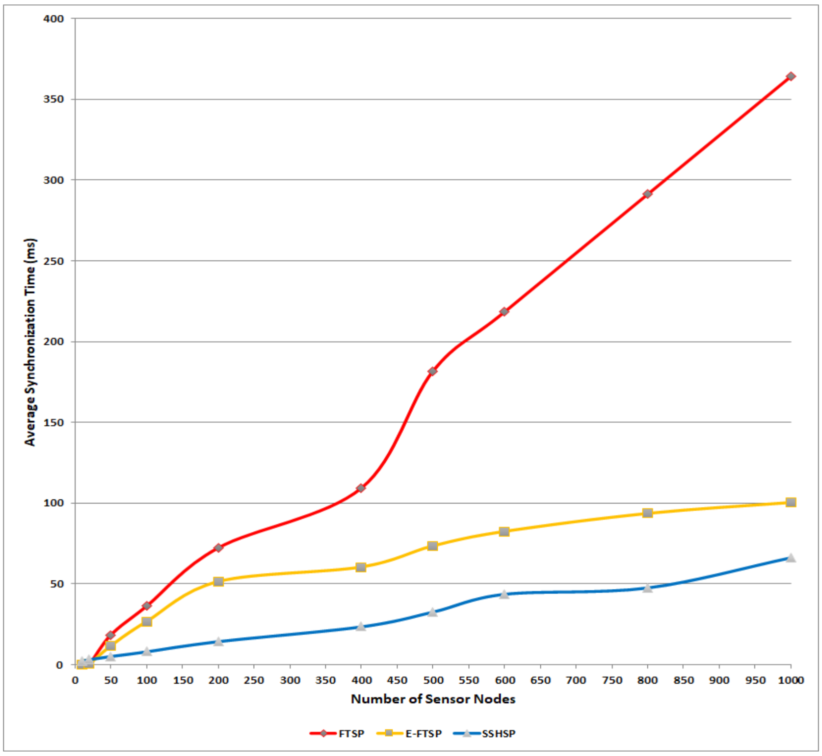

Fig. 7. SSHSMP, FTSP and E-FTSP time synchronization performance

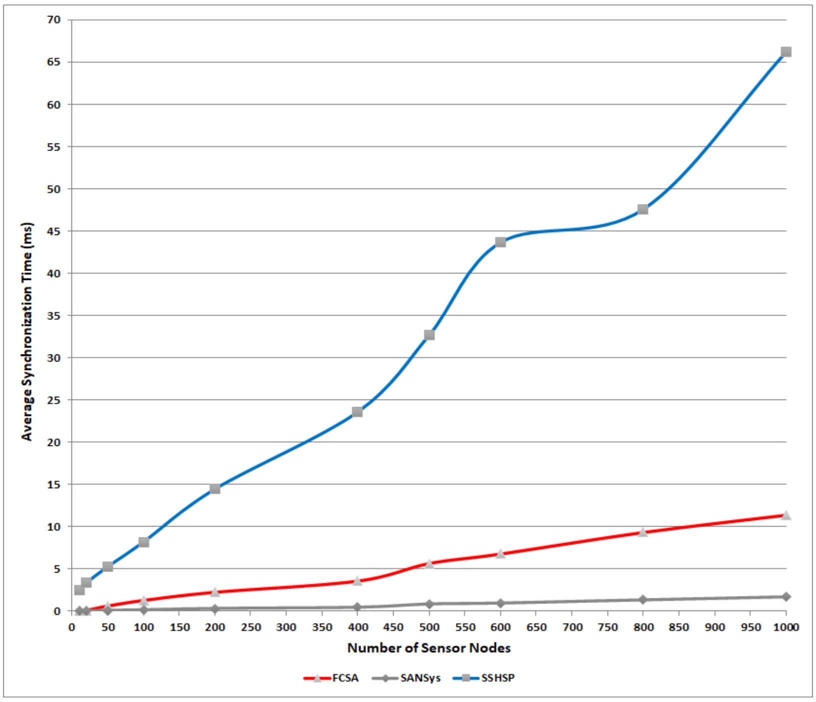

Fig. 8. SSHSMP, FCSA and SANAys time synchronization performance

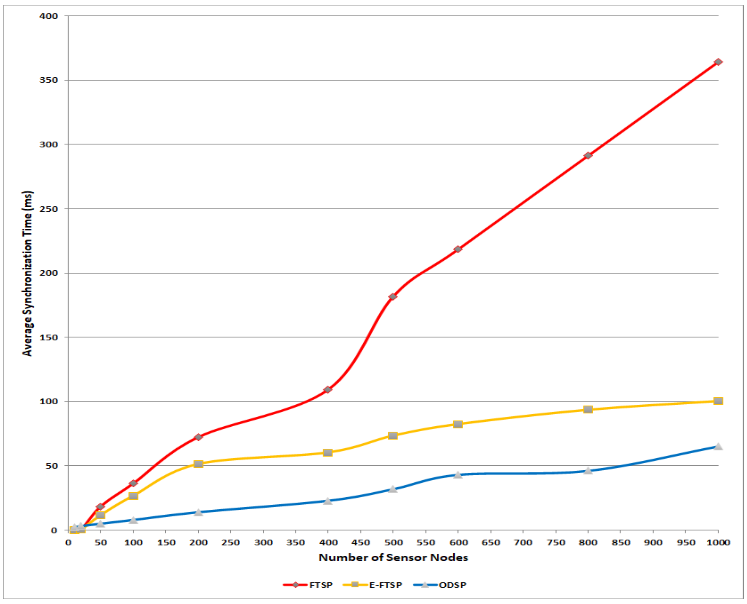

Fig. 9. ODSMP, FTSP and E-FTSP time synchronization performance

The On-Demand Synchronization Mac Protocol (ODSMP) is based on node acknowledgment packets, not necessarily with its parents but with its neighbor nodes in the hop. In addition, the acknowledgment packet is one way from the node to its parent. Therefore, the synchronization time is less than that of the SSHSMP where two-way acknowledgment packets are needed between the nodes and their parents in each hop. The experimental results show that the ODSMP outperforms state-of-the-art protocols FTSP [28] and E-FTSP [11]. However, FCSA and SANAys protocols are faster as they do not support out-of-synchrony cases. Figs 9 and 10 present the time synchronization performance of the ODSMP compared with FTSP and E-FTSP, and FCSA and SANAys protocols, respectively.

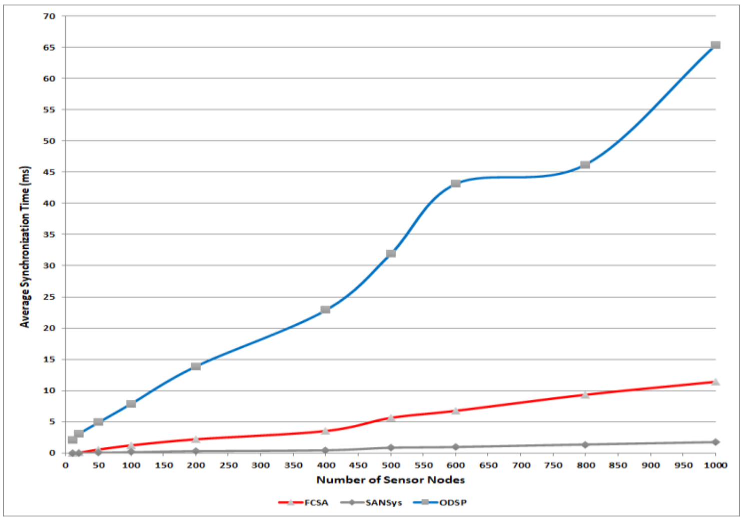

Fig. 10. ODSMP, FCSA and SANAys time synchronization performance

Finally, the experimental results show that the proposed time synchronization approach generates the minimum synchronization time at each node and subsequent nodes in the WSN, supports the security needs of real-time systems synchronization, operates in high-loss and unreliable environments, and helps in reducing the communications cost and improving WSN performance.

5. Conclusion

Providing a common time sharing is necessary for many wireless sensor applications since the related data should be meaningful to generate consistent inferences about the environment being sensed. Therefore, WSNs need to accurately operate in high-loss and untrustworthy environments. Consequently, three integrated time synchronization protocols have been proposed to serve multiple WSN applications under normal, secured, and unreliable situations. The protocols presented in this research help much in building competitive time synchronization schemes for wireless sensor network applications. In addition, the design considerations in this approach assist in integrating various WSN configurations and comparing the results with the state-of-the-art protocols in the literature. In future work, more attention shall be paid to combining the synchronization of several cloud clusters and improving the accuracy of clock drift computations.

References

1. Khalil, I., A. Khreishah, F. Ahmed, K. Shuaib. Dependable Wireless Sensor Networks for Reliable and Secure Humanitarian Relief Applications. – Ad Hoc Networks, Vol. **13**, 2014, pp. 94-106.
2. Bagchi, S., N. Shroff, I. Khalil, R. Panta, M. Krasniewski, J. Krogmeier. Protocol for Secure and Energy-Efficient Reprogramming of Wireless Multi-Hop Sensor Networks. US Patent No 8107397, 2012.
3. Wu, Yik-Chung, Q. Chaudhari, E. Serpedin. Clock Synchronization of Wireless Sensor Networks. – Signal Processing Magazine, IEEE, Vol. **28**, January 2011, Issue 1, pp. 124-138.
4. Fadela, E., V. C. Gungorb, L. Nassefa, N. Akkaria, M. G. Abbas Maika, S. Almasria, I. F. Akyildiza. A Survey on Wireless Sensor Networks for Smart Grid. – Computer Communications, Vol. **71**, 1 November 2015, pp. 22-33.
5. Khediri, S. E., N. Nasri, M. Samet, A. Wei, A. Kachouri. Analysis Study of Time Synchronization Protocols in Wireless Sensor Networks. – arXiv preprint arXiv:1206.1419, 2012.
6. Wu, Y.-C., Q. Chaudhari, E. Serpedin. Clock Synchronization of Wireless Sensor Networks. – IEEE Signal Processing Magazine, Vol. **28**, 2011, No 1, pp. 124-138.
7. Khalil, I., M. Awad, S. Bouktif, F. Awwad. MSN: Mutual Secure Neighbor Verification in Multi-Hop Wireless Networks. – Security and Communication Networks (A Wiley Journal), Vol. **5**, February 2012, Issue 2, pp. 186-196.
8. Mei, L., Y.-C. Wu. Distributed Clock Synchronization for Wireless Sensor Networks Using Belief Propagation. – Reading, Vol. **101**, 2011, t1.
9. Yildirim, K. S., A. Kantarci. External Gradient Time Synchronization in Wireless Sensor Networks. – IEEE Transactions on Parallel and Distributed Systems, Vol. **25**, March 2014, Issue 3, pp. 633-641.
10. Akl, R., Y. Saravanos. Hybrid Energy-Aware Synchronization Algorithm in Wireless Sensor Networks. – In: Proc. of 18th International IEEE Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC’07), September 2007, pp. 1-5.

11. Phan, L., T. Kim, T. Kim, J. Lee, J. Ham. Performance Analysis of Time Synchronization Protocols in Wireless Sensor Networks. – Sensors, Vol. **19**, 2019, 3020. DOI:10.3390/s19133020.

12. Boukhechem, N., N. Badache. SANSync: An Accurate Time Synchronization Protocol for Wireless Sensor and Actuator Networks. Wireless Personal Communications. Springer Science + Business Media, LLC, Part of Springer Nature, 2019.

13. Ni, X., T. Lu, S. Ye, Y. Zheng, P. Chen, L. Chen. Pair Nodes Clock Synchronization Algorithm Based on Kalman Filter for Underwater Wireless Sensor Networks. – Sensors, Vol. **21**, 2021, No 13, 4426.

14. Chen, Y., X. Zheng, Y. Luo, Y. Shen, Y. Xue, W. Fu. An Approach for Time Synchronization of Wireless Accelerometer Sensors Using Frequency-Squeezing-Based Operational Modal Analysis. – Sensors, Vol. **22**, 2022, 4784.

15. Ni, X., T. Lu, S. Ye, Y. Zheng, P. Chen, L. Chen. Pair Nodes Clock Synchronization Algorithm Based on Kalman Filter for Underwater Wireless Sensor Networks. – Sensors, Vol. **21**, 2021, 4426. DOI: 10.3390/s21134426.

16. Perez-Solano, J. J., S. Felici-Castell, A. Soriano-Asensi, J. Segura-Garcia. Time Synchronization Enhancements in Wireless Networks with Ultra-Wide Band Communications. – Computer Communications, Vol. **186**, 2022, pp. 80-89. ISSN: 0140-3664.

17. Shi, F., X. Tuo, L. Ran, Z. Ren, S. X. Yang. Fast Convergence Time Synchronization in Wireless Sensor Networks Based on Average Consensus. – IEEE Transactions on Industrial Informatics, Vol. **16**, February 2020, No 2, pp. 1120-1129. DOI: 10.1109/TII.2019.2936518.

18. Zhang, X., Y. Liu, Y. Zhang. A Secure Clock Synchronization Scheme for Wireless Sensor Networks Against Malicious Attacks. – J. Syst. Sci. Complex., Vol. **34**, 2021, pp. 2125-2138.

19. Phan, L.-A., T. Kim. Hybrid Time Synchronization Protocol for Large-Scale Wireless Sensor Networks. – Journal of King Saud University – Computer and Information Sciences, 2022.

20. Hu, B., Z. Sun, J. Liu. Distributed Time Synchronization Algorithm Based on Sequential Belief Propagation in Wireless Sensor Networks. – Computer Communications, Vol. **176**, 2021.

21. Wang, H., P. Gong, M. Li. Consensus-Based Time Synchronization via Sequential Least Squares for Strongly Rooted Wireless Sensor Networks with Random Delays. – Automatica, Vol. **136**, 2022.

22. Noh, K. L., E. Serpedin, K. Qaraqe. A New Approach for Time Synchronization in Wireless Sensor Networks: Pairwise Broadcast Synchronization. – IEEE Transactions on Wireless Communications, Vol. **7**, September 2008, No 9, pp. 3318-3322.

23. Masood, W., J. F. Schmidt, G. Brandner, C. Bettstetter. DISTY: Dynamic Stochastic Time Synchronization for Wireless Sensor Networks. – In: IEEE Transactions on Industrial Informatics, 2016.

24. Yıldırım, K. S., R. Carli, L. Schenato. Adaptive Control-Based Clock Synchronization in Wireless Sensor Networks. – In: Proc. of European Control Conference (ECC’15), July 2015, pp. 2806-2811.

25. Hu, A. S., S. D. Servetto. Algorithmic Aspects of the Time Synchronization Problem in Large-Scale Sensor Networks. 2003.

26. Mills, D. L. Internet Time Synchronization: The Network Time Protocol. – IEEE Transactions on Communications, Vol. **39**, 1991, pp. 1482-1493.

27. Elson, J., L. Girod, D. Estrin. Fine-Grained Network Time Synchronization Using Reference Broadcasts. – ACM SIGOPS Operating Systems Review, June 2002, pp. 147-163.

28. Maroti, M., B. Kusy, G. Simon, A. Ledeczi. The Flooding Time Synchronization Protocol. – ACM Press, 2004, pp. 39-49.

29. Sommer, P., R. Wattenhofer. Gradient Clock Synchronization in Wireless Sensor Networks. – In: Proc. of International Conference on Information Processing in Sensor Networks (IPSN’09), IEEE Computer Society, Washington, DC, USA, 2009, pp. 37-48,

30. Hu, X., T. Park, K. G. Shin. Attack-Tolerant Time-Synchronization in Wireless Sensor Networks. – In: Proc. of 27th IEEE Conference on Computer Communications (INFOCOM’08), IEEE, 2008, pp. 41-45.

31. Sun, K., P. Ning, C. Wang. Secure and Resilient Clock Synchronization in Wireless Sensor Networks. – IEEE Journal on Selected Areas in Communications, Vol. **24**, 2006, No 2, pp. 395-408.

32. Vijayalakshmi, V., T. G. Palanivelu, N. Agalya. Secure Time Synchronization against Malicious Attacks for Wireless Sensor Networks. – In: Proc. of 1st International Conference on Emerging Trends in Engineering and Technology (ICETET'08), IEEE, 2008, pp. 218-222.

33. Apicharttrisorn, K., S. Choochaisri, C. Intanagonwiwat. Energy-Efficient Gradient Time Synchronization for Wireless Sensor Networks. – In: Proc. of 2nd International Conference on Computational Intelligence, Communication Systems and Networks, IEEE, 2010, pp. 124-129.

34. Ding, Z., N. Yamauchi. An Improvement of Energy Efficient Multi-Hop Time Synchronization Algorithm in Wireless Sensor Network. – In: Proc. of IEEE International Conference on Wireless Communications, Networking and Information Security (WCNIS'10), IEEE, 2010, pp. 116-120.

35. Bader, S., B. Oelmann. Adaptive Synchronization for Duty-Cycling in Environmental Wireless Sensor Networks. – In: Proc. of 5th International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP'09), IEEE, 2010, pp. 49-54.

36. Bian, T., R. Venkatesan, C. Li. Adaptive Time Synchronization for Wireless Sensor Networks with Self-Calibration. – In: Proc. of IEEE International Conference on Communications (ICC'09), IEEE, 2009, pp. 1-5.

37. Chen, J., Q. Yu, Y. Zhang, H. H. Chen, Y. Sun. Feedback-Based Clock Synchronization in Wireless Sensor Networks: A Control Theoretic Approach. – IEEE Transactions on Vehicular Technology, Vol. **59**, 2010, No 6, pp. 2963-2973.

38. Nuo, W., F. Yong, M. Shu, G. Qiang, X. Kong. A Cluster Based on Demand Time Synchronization in Wireless Sensor Networks. – In: Proc. of 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT'10), Vol. **3**, IEEE, 2010, pp. 204-207.

39. Dong, J., L. Gu, C. Zheng. Research and Application on Time Synchronization of Wireless Sensor Network Based on Information Fusion. – In: Proc. of 2nd International Conference on Computer Engineering and Technology (ICCET'10), Vol. **3**, IEEE, 2010, p. V3.

40. Guidoni, D. L., A. Boukerche, H. A. B. F. Oliveira, R. A. F. Mini, A. A. F. Loureiro. A Small World Model to Improve Synchronization Algorithms for Wireless Sensor Networks. – In: Proc. of IEEE Symposium on Computers and Communications (ISCC'10), IEEE, 2010, pp. 229-234.

41. Beluch, T., D. Dragomirescu, F. Perget, R. Plana. Cross-Layered Synchronization Protocol for Wireless Sensor Networks. – In: Proc. of 9th International Conference on Networks (ICN'10), IEEE, 2010, pp. 167-172.

42. Cheng, K. Y., K. S. Lui, Y. C. Wu, V. Tam. A Distributed Multi-Hop Time Synchronization Protocol for Wireless Sensor Networks Using Pairwise Broadcast Synchronization. – IEEE Transactions on Wireless Communications, Vol. **8**, 2009, No 4, pp. 1764-1772.

43. Yildirim, K. S., A. Kantarci. Time Synchronization Based on Slow-Flooding in Wireless Sensor Networks. – IEEE Transactions on Parallel and Distributed Systems, Vol. **25**, January 2014, No 1.

44. Devi, M., N. Sarma, S. K. Deka. A Centralized Model Enabling Channel Reuse for Spectrum Allocation in Cognitive Radio Networks. – Cybernetics and Information Technologies, Vol. **21**, 2021, No 2, pp. 183-200.

45. Phan, L., T. Kim. Hybrid Time Synchronization Protocol for Large-Scale Wireless Sensor Networks. – Journal of King Saud University – Computer and Information Sciences, Vol. **34**, 2022, Issue 10, Part B.

46. Cotrim, R. S., J. M. L. P. Caldeira, V. N. G. J. Soares, Y. Azzoug. Power Saving MAC Protocols in Wireless Sensor Networks: A Survey. – TELKOMNIKA (Telecommunication Computing Electronics and Control), Vol. **19**, 2021, No 6, pp. 1778-1786.

47. Kumari, S., M. Bharti. MAC Layer Protocol for Wireless Security. – Wireless Communication Security, 2023, 23.

48. Mamta, M., R. Singh. A Comprehensive Analysis of Application-Based MAC Protocol for Wireless Sensor Network. – In: Proc. of Advanced Computing and Intelligent Technologies (ICACIT'22), Singapore, Springer Nature Singapore, 2022, pp. 183-198.

49. Yassin, E., A. Boulouz, M. B. Salah, S. El Hachemy. Performances Prediction in Wireless Sensor Networks: A Survey on Deep Learning Based-Approaches. – ITM Web of Conferences EDP Sciences, Vol. **43**, 2022, p. 01010.

50. Afraa, A., et al. EE-MAC: Energy Efficient Sensor Mac Layer Protocol. – In: Proc. of 38th IEEE Conference on Local Computer Networks Workshops (LCN Workshops), IEEE, 2013.

Received: 12.12.2022; Second Version: 09.02.2023; Accepted: 21.02.2023 (*fast track*)