Power Consumption Analysis Model in Wireless Sensor Network for Different Topology Protocols and Lightweight Cryptographic Algorithms

Nemanja Radosavljević, Djordje Babić

School of Computing, Union University, Serbia
nradosavljevic@raf.edu.rs, djbabic@raf.edu.rs

Abstract

In this paper we give power consumption estimation model for WSN for given set of topology protocols and lightweight cryptographic algorithms. The proposed power consumption estimate method is analyzed in simulation scenario for several protocols and cryptographic algorithms. We test the mathematical model in the simulation environment where WSN is applied for agriculture field monitoring of a typical size. The simulation results confirm the proposed power estimation formula. The proposed model can be used to select appropriate topology protocol and cryptographic algorithm. Based on the defined model and the simulation results, a sorted list of combinations of topology protocols and cipher algorithm is presented. The best power consumption results are obtained by using A3 topology protocol and KATAN64 cryptographic protocol.

Keywords: Topology protocols, Lightweight cipher algorithms, Field monitoring, Wireless sensor network security, Power consumption

1 Introduction

Wireless Sensor Networks (WSN) are applicable in different areas. Body Sensor Networks are used for health monitoring in ubiquitous computing for health and medicine. WSN are used in military applications, industry, smart city, railways [1], traffic flow [2] and smart house application [3]. In this paper, we are specifically interested in precision agriculture and application of WSN in field monitoring. WSN are used in precision agriculture to collect data about field state and crop conditions, for example sensors can be applied to track soil humidity and temperature. The data collected are used to make decision about watering, fertilizing, in order to produce crops effectively [4].

WSN are composed of certain number of sensor nodes. The main components of a sensor node are a microcontroller, transceiver (RF module), external memory, power source and one or more sensors. Sensors nodes are interconnected for collecting data, and sending information to a remote server.

In many applications, including precision agriculture, there are several challenges related to the implementation of WSN. The most important aspects of WSN are energy consumption and data security. The topology protocol in the WSN is used to optimize the number of messages, reducing energy consumption and thus extending the life span of the observed WSN.

The energy consumption can be further reduced by an appropriate lightweight cryptographic algorithm. The lightweight cryptographic algorithm make tradeoff between security performance and energy consumption which is directly related to number of operations per information bit. In order to optimize security performance and energy consumption in WSN, it is of a great importance to select optimal combination of topology protocol and lightweight cryptographic algorithm for implementation in WSN.

In this paper, we derive a power consumption estimation model for WSN which is based on the given topology protocol and cryptographic algorithm. The proposed mathematical model of power consumption is based on three parameters: total number of messages sent throughout the entire system, energy cost for different encryption algorithms and coverage coefficient of topology protocol. The proposed mathematical model is an effective, simple to use tool which can be implemented by using simple code or spreadsheets. The power consumption estimation is further evaluated in simulation scenario for several cryptographic algorithms by means of Atarraya. We focus on lightweight cryptographic algorithms in order to reduce power consumption. The simulation scenario is agriculture field monitoring of given area. The simulation results confirm the proposed power estimation formula. Based on the presented model and the simulation results, a sorted list of combinations of topology protocols and cipher algorithm is presented. The best power consumption results are obtained by using A3 topology protocol and KATAN64

[*] Corresponding Author: Nemanja Radosavljević; E-mail: nradosavljevic@raf.edu.rs

DOI: 10.3966/160792642021012201007

cryptographic protocol. This concludes the main finding of the paper on final decision which combinations of cryptographic algorithms and topologies are acceptable for given WSN application scenario.

2 Related Work

Jawad et al. [5] studied how WSNs are used to observe various ecological phenomena on a large surface, which is achieved by using different sensor nodes. The way of communication between sensor nodes depends primarily on the assembly of the nodes, as well as, on the topology on which the WSN is based. Valente et al. [6] displayed that the use of WSNs in PA takes place in several stages: data collection, diagnosis, data analysis, precise crop management, and evaluation of the achieved results. Sensor nodes used in data collection can be very complex, as in the case where they monitor location or analyze images, but also they can be relatively simple, when they monitor changes in temperature, pressure, humidity, PH values, water level, and other indicators.

Ojha et al. [7] define the following challenges when applying WSNs in field monitoring: development of optimal node deployment plans, determination of the measurement period, selection of routing protocols, energy efficiency, data transfer method, scalability, degree of fault tolerance, transmission security and data accuracy. In addition to challenges with energy consumption, Oliveira et al. [8] emphasize the problem of data security in WSNs. Security aims of WSNs in PA such as availability, confidentiality, integrity and authentication of data transmitted over the network, are mostly achieved by using cryptography.

The problem of controlling the arrangement of sensor nodes in large areas is solved in [7], which represents a simple, time-efficient and efficient technique for setting up sensor nodes. In [9], Pachnanda et al. consider a protocol of building an efficient topology in terms of coverage and reduction of energy consumption in WSN. Three different protocols are presented in the paper. The results indicate that the Kneigh tree provides higher coverage and consumes less energy compared to other protocols.

Bogdanov and Knudsen [10] developed a new block cipher algorithm named PRESENT consisting of 31 rounds with block size of 64 bits and two keys of size of 80 and 128 bits. The PRESENT block cipher has software and hardware efficiency comparable to other algorithms of similar purpose. In [11], Gong et al. describe a group of lightweight square ciphers called KLEIN, which can be used in low power devices such as remote sensors and RFID tags. The variable length of KLEIN keys offer flexibility for different needs. Suzaki and Minematsu in [12] propose TWINE, lightweight 64-bit square cipher, which shows satisfactory performance in devices with limited hardware capabilities such as sensor nodes. Another lightweight block cipher named LBlock is described in [13]. Similarly to many other lightweight square ciphers, the size of the LBlock block is 64-bit and the key size is 80-bit. LBlock is an adequate measure to cope with some of the known threats such as differential cryptanalysis, direct cryptanalysis, impossible differential cryptanalysis and related-key attacks. Lightweight command block cipher named RECTANGLE is proposed by Kushwaha et al. in [14]. The basic idea behind the RECTANGLE is to enable an easy and fast execution by using bit reduction procedures. RECTANGLE is computationally efficient in both hardware and software implementation, and thus it can be easily adapted to different hardware.

3 Security problems and objectives in WSN

Sensor nodes in WSN have limited memory, energy, calculation ability, bandwidth, and range of communications.

- Ad hoc deployment of sensor nodes in the WSN makes it easy for intruders to launch various types of intrusions ranging from active interference to passive eavesdropping.
- The WSN topology is dynamic and it doesn't have fixed infrastructure that is why continuous monitoring of the network is difficult.
- The wireless network is can be easily tracked by radio receivers at the same frequency. In this way, intruders can have more chances to break into WSN.
- Strong security protocols can degrade the performance of WSN as they consume more resources at sensor nodes. Therefore, a compromise between performance and safety must be established. However, intruders can easily break down weak security protocols.

We conclude that it is very important to determine and implement appropriate light cryptographic algorithm in terms of computational complexity. The main target of this study is to analyze use of lightweight cryptographic algorithms in different WSN topology and implementation scenarios.

In order to protect data transfer in WSNs, there are two fundamental goals security and survival requirements of WSNs. The most important security objectives in WSNs with respect to their applications are [15]:

- confidentiality, where the greatest risk is the existence of compromised nodes, which can be exploited by the intruder to steal important data such as cryptographic keys;
- authentication, all base stations and sensor nodes must be able to determine whether the packet is sent by an attacking or legitimate node,
- data integrity, the use of incorrect or inaccurate data can lead to serious consequences, thus WSN relies

heavily on the integrity of information transmitted through the network;

- security management, where base station control is of particular important.

4 Topology protocols in WSNs

The main task of topology protocol in the WSN is to optimize the number of messages, as well as, to reduce energy consumption and thus extend the life span of the observed WSN. The topology protocols A3, A3 coverage, Energy efficient connected dominating set and CDS rule K are analyzed in this scope.

4.1 A3 Protocol

The A3 protocol builds a non-optimal connected dominating set (CDS) on a connected graph based on information about the residual energy in the nodes and the signal strength between them. The A3 operates in three phases: Neighborhood Discovery, Children Selection and Second Opportunity [16].

To build A3 topology, the protocol relies on four types of messages. The predefined sync node initiates the establishment of (CDS) by sending a HELLO message. The other nodes respond to this message by sending information about the idle time energy and signal strength. The topology is based on selection metric which favors nodes of greater energy and distance compared to the cluster head [17]. Based on the received messages, the cluster head forwards a sorted list of idle time energy information to all nodes in its cluster. In the same manner, all nodes independently examine the nodes located in their communication radius. Nodes that are not selected for the topology at this point send a Sleep Message [9].

The cluster head is selected among candidate parent nodes based on the metric given in Equation (1) which favors nodes with greater amount of energy and farther from the parent node. The metric tends to build a tree with smaller number nodes and wider coverage [16]:

$$M_{x,y} = W_E * \frac{E_x}{E_{\max}} + W_D * (\frac{RSSL_y}{RSSI_*}) \qquad \textbf{(1)}$$

where M is the metric cost, x is a regular node, y is a candidate node for cluster head, W_E is the residual energy of the node, E_x is the residual energy of the node x, E_{max} is the maximum initial energy, W_D is the weighted distance from the cluster head, $RSSI_y$ is the signal strength received by the cluster head, and $RSSI_*$ is the minimum $RSSI$ required to connect the two nodes [16].

4.2 A3 Coverage

A3 coverage is a modification of the original A3 algorithm, which is based on the sensing radius. The A3 coverage algorithm requires additional phase in the A3 algorithm. The algorithm relies rather on the sensing radius and not on node's communication radius. The sensing radius generally much smaller than the communication radius. If a regular node is not within the sensor's sensing radius, it cannot be active in the network. The A3 coverage node setting algorithm requires higher density of sensor nodes in the network and thus it directly affects the topology complexity, as well as, the increase in energy consumption [16].

4.3 Energy Efficient Connected Dominating Set (EECDS)

The EECDS algorithm is based on the fact that a node in the WSN that initiates communication changes its status to black and thus declares itself as part of newly established maximal independent set (MIS). The status of the other nodes remains white, i.e., they still have no role in the network. After changing its status, the black node sends a message informing the nodes in its communication radius that they belong to the same MIS. All nodes that receive the message from the node that initiated the communication change status to gray, and send messages about status change to all neighboring nodes. After a gray message is received, a node that has not changed initial status, i.e., the white node, , determines whether it changes status to black and creates a separate MIS or joins the existing MIS. In order to determine if there is any black node in communication radius, the white nodes send an inquiry message about the status of the neighboring nodes. If they do not receive a response from a node that has already switched status to black, then they become a black node and this is repeated in multiple iterations.

Following the initial communication of the nodes, a CDS is created based on the connector nodes that remain outside of MIS [18]. These nodes, called connectors, are selected by MIS nodes using three types of messages: blue message, update message, and invite message [9].

The connector node sends a blue message to its neighbors and, using the invite message, invites the nodes to become gateway nodes. The nodes that received the blue message calculate the weight and return an update message. Gateway nodes join CDSs, which are the highest-weight nodes [16].

4.4 CDS Rule K

The CDS rule K algorithm consists of two phases. The initial phase is to create a maximally connected tree. During the second phase, the tree is to reduce to the necessary minimum CDS [16].

The first phase is defined by the following statement [16]:

$$S = (\forall v \in S : x, y \in N(v), \neg E(x, y) \in V) \qquad \textbf{(2)}$$

In the statement *N(v)* is a set of neighboring nodes *v*. According to the statement above, a node *v* that has two or more neighbors, which are not interconnected, is added to the initial tree. Nodes transmit HELLO messages that contain a list of neighboring nodes. Upon receiving a HELLO message, the node compares the data from the message with a list of its neighbors. A node is added to the tree if the number of neighbors in its list is greater than the number from the received list [16].

Based on the K rule, the maximized starting tree is truncated. The sync node is the node of the lowest level in the starting tree, and the neighboring nodes are marked by one layer higher. Lower-level layers are nodes with higher priority.

The K rule ensures that the communication tree is minimized by using the rule that a node goes to the idle state in the case when all neighboring nodes are in the higher priority node's communication radius, or in the case when all neighboring nodes are covered by the communication radius of the other two connected nodes [16].

5 Lightweight Cipher Algorithms

The development and implementation of Lightweight Block Cipher (LWC) algorithms is of great importance in WSNs. Hatzasasilis et al. [19], made a comprehensive review that addressed, among other things, cipher algorithms which we use in simulation scenarios in this paper. In this paper, we evaluate the following cipher algorithms: AES, NOEKEON, PRESENT, LED, Piccolo, TWINE, KATAN, KATANAN, PRINCE, SIMON. In a sequel, we shortly describe these cipher algorithms, we also point out the basic parameters relevant for our research.

AES is a block cipher with 128-bit blocks and three different key sizes of 128-, 192- and 256-bit which is implemented in 10, 12 and 14 rounds, respectively. In order to reduce computational complexity the AES S-box is modified in the way that it requires 2400 gate equivalents (GE) and 226 cycles per block [20]. The S-box used in the lightweight versions is further minimized by the use of scan flip-flops. The spatial requirements of the control logic are optimized using the linear-feedback shift registers.

NOEKEON [21] uses 128-bit keys and block through 16 identical rounds. The first hardware implementation [22] occupies 2880 GE and is suitable for lightweight devices. The software implementation requires 364 bytes of code for a flow rate of 21.7 Kbps and is suitable for 32-bit processors. The algorithm uses the same data for encryption and decryption, allowing the same round to be reused for the opposite operation.

PRESENT [10] uses 80- and 128-bit keys with 64-bit block through 31 rounds. Encryption implementation requires nearly 1000 GE, making it suitable to meet ultra-lightweight requirements [23]. PRESENT is a turning point in the evolution of lightweight block ciphers and is used in conjunction with AES as a benchmark for newer proposals. The main feature of this algorithm is that it changes the ordinary eight S-boxes with one carefully selected S-box. When implemented on an 8-bit microcontroller, the algorithm is executed in 11342 cycles for encryption and 13599 for decryption.

LED uses 64-, 80-, 96- and 128-bit keys with 64-bit block through 32 and 48 rounds (lightweight encryption device) [24].The algorithm is designed to retain reasonable software performance with small hardware modifications for its implementation. The implementation code is AES-like based. Furthermore, the authors apply some recent trends in the field of block-cipher-based lightweight hash functions. The LED uses a key non-allocation process, which is one of the main advantages of this algorithm. Differential error analyzes based on Super-S-box techniques shows significant improvements for error assaults [25].

Piccolo uses 64-bit block with key size equal to 80- and 128-bit, in 25 and 31 rounds. Analyzing the 80-bit variant, we see that it requires 432 GE to encrypt and 60 GE to decrypt. The specificity of this algorithm is the use of AND-NOR and OR-NAND, which are 12.5% more optimal than XOR and XNOR [26]. In software implementation it requires 2434 bytes of code and consumes 79 bytes of RAM with a 7.8 Kbps bandwidth [27].

TWINE is a 64-bit block cipher with 80- and 128-bit keys through 36 rounds. In implementation, it requires 1866 GE. It is a GFN and implements a unique encryption and decryption function. TWINE, unlike LBlock, uses one S-box and half-byte permutations at a bit level.

KATAN [28] supports 80-bit key size and three different 32-, 48- and 64-bit block sizes through 254 rounds. 802 GE is required for optimal KATAN execution. The cipher uses a very simple key allocation mechanism. KATAN is only suitable for devices where the key is initialized once and does not change, making it acceptable for WSN deployment. The software implementation, although taking up a rather small 338 bytes space, has low bandwidth and 72963 cycles for encryption, and 88525 cycles for decryption [29].

PRINCE uses 128-bit key size with 64-bit block through 12 rounds and has small hardware delay. Its implementation requires 2953 GE for 533.3 Kbps bandwidth which means low power consumption [30]. Efficient software implementations has also been presented in [31].

SIMON supports different sizes of key and block. For us, the most important is 64 bit block size and 96 bits key size in 17 rounds. It has good hardware and software implementation [32].

Based on the description of the above algorithms, we give an overview of their characteristics in Table 1. Further, we define the input parameters of the simulation model, where energy per byte column is of particular importance for our analysis

Table 1. Overview of LWC characteristics

Cipher	Block	Round	Unrolled	Num. of	Delay per	Energy per byte
	/Key size	type	Rounds	Cycles	round (ns)	(pJ)
AES	128/128	SPN	1	11	3.32	21.92
Noekeon	128/128	SPN	1	18	3.41	21.20
LED 128	64/128	SPN	1	50	5.25	82.08
Present	64/80	SPN	2	17	2.09	19.44
Prince	64/128	SPN	1	13	4.06	18.64
Picolo	64/80	Feistel	1	26	3.28	22.24
TWINE	64/80	Feistel	2	19	3.10	26.80
Simon64/96	64/96	Feistel	2	22	2.18	26.56
KATAN64	64/80	Shift register	16	17	2.04	17.52

6 Mathematical Model of Power Consumption and Design Support

Here, we derive mathematical model for power consumption which is based on overall number of messages and cryptographic algorithm consumption. We also present algorithm which can be used for calculation of the power consumption for given WSN based on the presented mathematical model.

The power consumption can be represented as random variable W according to the following equation:

$$W=\frac{X*Y}{Z} \quad \textbf{(3)}$$

here *X* represents the total number of messages sent throughout the entire system, *Y* is defined as energy per byte for given cryptographic algorithm whose values are given in the Table 1 for different encryption algorithms, and *Z* is the coverage coefficient of the observed area with different value per topology protocol. All variables *X, Y, Z* follow normal distribution.

Values of variable *W* represent the comparatively obtained results for different topology protocols and encryption algorithms in the form of a matrix. Based on the obtained values of *W* (W_1, W_2, W_3, ..., W_n), we calculate the mean value, which we denote with μ ˆ, where the following formula applies:

$$\hat{\mu}=\frac{1}{n}*(w_1+w_2+\cdots+w_n) \quad \textbf{(4)}$$

The confidence interval for the unknown mean value is represented using the Student's t-distribution. Student's t-distribution can be approximated by the normal distribution if number of samples is greater than 30. In our derivation we have more than 30 samples, and thus we use normal distribution to approximate mean value:

$$\frac{\hat{\mu}-\mu}{\frac{s}{\sqrt{n}}}\sim\mu(0,1) \quad \textbf{(5)}$$

where *S* is the dispersion estimate.

Next, we determine standard deviation *δ*:

$$\delta=\sqrt{\frac{1}{n-1}*\sum_{i=1}^{n}(w_i-\hat{\mu})^2} \quad \textbf{(6)}$$

Based on the three-sigma rule [33], we take an interval of 68% and obtain the following results:

$$P=\{|\frac{\hat{\mu}-\mu}{\frac{s}{\sqrt{n}}}|\}<\varepsilon=68\% \quad \textbf{(7)}$$

Based on the previous formula, we can calculate the value of *ε* as:

$$(\phi)\varepsilon=\frac{0.68}{2}+0.5=0.84\Rightarrow\varepsilon=0.995 \quad \textbf{(8)}$$

$$[\hat{\mu}-\delta,\hat{\mu}+\delta] \quad \textbf{(9)}$$

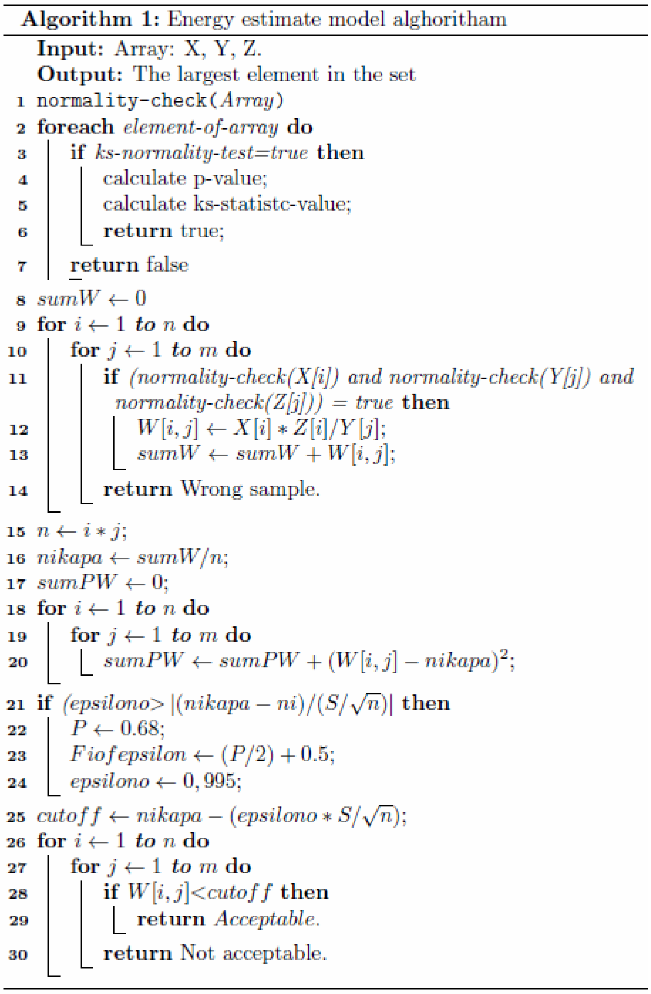
for the interval:

$$\mu\in[\hat{\mu}-\varepsilon*\frac{s}{\sqrt{n}},\hat{\mu}+\varepsilon*\frac{s}{\sqrt{n}}] \quad \textbf{(10)}$$

In Equation 11, we present an interval based on the set of equations above. The displayed interval is the input parameter for decision which values are acceptable to us as the output from the simulation. We reject a portion of the confidence interval along with the right side, and we find acceptable results based on the interval:

$$[\min,\hat{\mu}-\varepsilon*\frac{s}{\sqrt{n}}] \quad \textbf{(11)}$$

The proposed model for energy estimate can be implemented using the following algorithm. The algorithm contains the *function normality-check(array)* which is used to check if samples of the input array follow normal distribution. Further, if all three vectors *X*, *Y* and *Z*, follow normal distribution, the energy estimate sample W_n is calculated using procedure explained above. Next, mean value and deviation are calculated. Finally, Equation 11 is used to determine whether the energy estimation is within confidence interval or it is rejected.



```
Algorithm 1: Energy estimate model alghoritham
  Input: Array: X, Y, Z.
  Output: The largest element in the set
normality-check($Array$)
foreach $element$-$of$-$array$ do
    if $ks$-$normality$-$test$=$true$ then
        calculate p-value;
        calculate ks-statistc-value;
        return true;
    return false
$sumW \leftarrow 0$
for $i \leftarrow 1$ $to$ $n$ do
    for $j \leftarrow 1$ $to$ $m$ do
        if ($normality$-$check(X[i])$ $and$ $normality$-$check(Y[j])$ $and$
         $normality$-$check(Z[j])) = true$ then
            $W[i,j] \leftarrow X[i] * Z[i]/Y[j]$;
            $sumW \leftarrow sumW + W[i,j]$;
        return Wrong sample.
$n \leftarrow i * j$;
$nikapa \leftarrow sumW/n$;
$sumPW \leftarrow 0$;
for $i \leftarrow 1$ $to$ $n$ do
    for $j \leftarrow 1$ $to$ $m$ do
        $sumPW \leftarrow sumPW + (W[i,j] - nikapa)^2$;
if ($epsilono > |(nikapa - ni)/(S/\sqrt{n})|$ then
    $P \leftarrow 0.68$;
    $Fiofepsilon \leftarrow (P/2) + 0.5$;
    $epsilono \leftarrow 0,995$;
$cutoff \leftarrow nikapa - (epsilono * S/\sqrt{n})$;
for $i \leftarrow 1$ $to$ $n$ do
    for $j \leftarrow 1$ $to$ $m$ do
        if $W[i,j] < cutoff$ then
            return $Acceptable$.
        return Not acceptable.
```

7 Experimental and Simulation Results

In this section we test the proposed energy estimation model in simulation environment. The output of the simulation is sorted list of power consumption estimates for given topology protocols and lightweight cryptographic algorithms.

7.1 Simulation Environment

We use the Atarraya network simulator (http://www.csee.usf.edu/~mlabrador/Atarraya/), as a convenient tool. The simulator has proven itself to be the best starting point for generating the topology of the observed area, number of messages and size of messages. All these parameters depend on the number of surrounding nodes that the observed node uses as intermediaries for relaying messages.

Atarraya is an open source simulator written in Java, making it suitable for modification and personalization. In its base form, Atarraya does not deal with encryption algorithms or energy consumption for encryption. Table 1 provides the power consumption of different encryption algorithms which we incorporate into the simulation environment. Based on the obtained data, we make a comparative analysis of different protocols for generating topology and cluster head selection from the perspective of the energy efficiency of the implementation of encryption algorithms.

Another important issue considered in this paper is the communication coverage of the observed area. It depends significantly on the used topology protocol. The mutual dependence of communication coverage and power consumption has not been taken into consideration in the available literature.

7.2 Simulation Scenario

We simulate an observation area of 16 ha of square shape. The simulation area is comparable to common fields where WSN can be applied for field monitoring. The area is large enough for unobstructed node distribution and calculation of the coverage for selected value of sensor node communication radius. In addition, the selected area size can be linearly scaled, along with node density and other simulation parameters explained below, to larger and smaller areas without significant difference in performance of the WSN. The results and main conclusions will stand for areas of different shape, too. The scalability makes results presented in this paper more general and applicable to similar scenarios of different area size.

Table 2 introduces basic simulation parameters, which are the same for all topology protocols and encryption algorithms.

Table 2. Basic simulation parameters

Area of application	400 x 400 m
Communications range	100m
Reading range	20m
Small packet size	10 to 20 bytes
Large packet size	20 to 40 bytes
Metrics	W1=0.5; W2=0.5
Node distribution	normal distribution

The topology protocols under consideration are: A3, A3 coverage, EECDS, CDS rule K. The method of operation in simulation environment is according to what we described earlier in this paper. The number of sent and received messages is used as the main metric for the efficiency of the topology protocol.

The Atarraya simulator provides information about the number of sent and received messages. However, in order to analyze topology protocols from the point of view of energy efficiency we need to modify the simulator to determine the traffic of each node in bytes. Furthermore, the simulator must take into calculation the coverage ratio of the observed area based on the communication radius of the cluster head nodes.

We execute simulation with previously described encryption algorithms: AES, Noekeon, LED 128, Present, Prince, Picolo, TWINE, Simon64/96, KATAN64, and we make comparative analysis of the power consumption based on Energy per byte column in pJ provided in Table 1.

In previous Section, we derived power consumption model based on the assumption that the values indicated must meet the criteria of normal distribution. Based on K-S statistical test [34] 0.42396 and p-value of 0.05545, these values do not deviate significantly from the normal distribution.

The energy consumption in WSN simulation scenario for the given encryption algorithm is calculated using the size of relayed messages and energy per byte consumption. The results are implemented into the simulator in order to determine the average power consumption for the given encryption algorithm and different topology protocols and coverage ratios. The following equation shows the average energy consumption for given encryption algorithm E_{avg}:

$$E_{avg} = (M_s * rand\_num\_btw(x_s, y_s)) + M_b * (rand\_num\_btw(x_b, y_b)) * E_c \quad \textbf{(12)}$$

Where M_s is number of small messages (between 10 and 20 bytes), M_b is number of large messages (between 20 and 40 bytes), E_c is energy consumption of encryption algorithms per byte (see Table 1). The function *rand_num_btw(x,y)* is used to generate random message size from the given range, where x_s represents the smallest small message, y_s represents the largest small message, x_b represents the smallest large message, and y_b represents the biggest large message

7.3 Simulation Results

In a sequel, we gradually introduce and present simulation results, from intermediate to final results. Finally, we draw main conclusion which are based on simulation results and mathematical model of power consumption presented in Section 6. In addition, we also present a graphical representation of all topologies with the arrangement of nodes in the observed area, with defined role of cluster head or regular nodes that are used only for data collection. In Figure. 1 to Figure 4, graphical representation of the node distribution and coverage for different topology protocols is shown. The red dots represent the cluster heads while blue dots represent regular nodes.

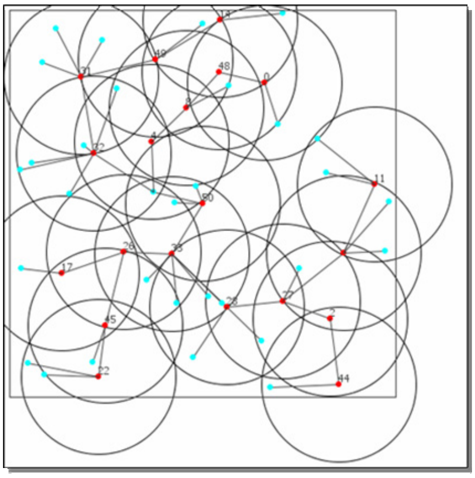


Figure 1. A3 protocol node distribution

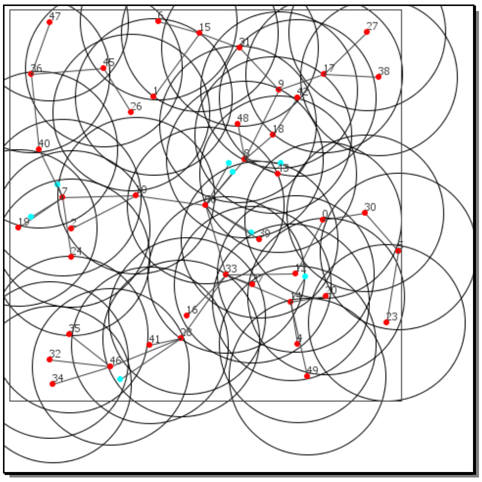


Figure 2. A3 coverage protocol node distribution

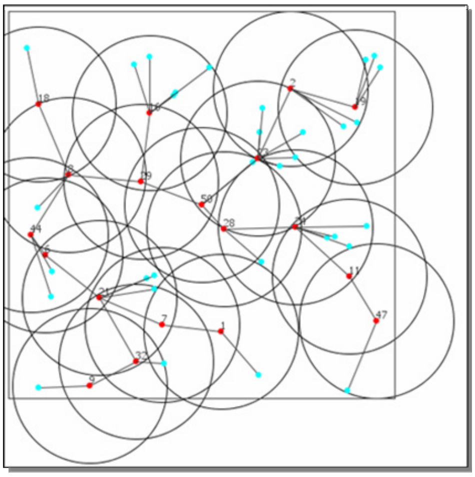


Figure 3. CDS rule k protocol node distribution

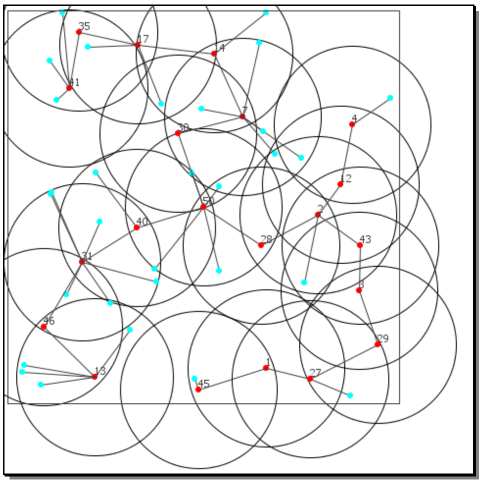


Figure 4. EECDS protocol node distribution

In Table 3, we show the time complexity of the simulation for different topology protocols. The time complexity is the highest in the case when EECDS protocol is used, and it is the lowest in the case when A3 is used. Table 4 presents the coverage coefficient of the observed area from the perspective of the communication radius of the cluster heads, as well as, the coverage beyond the observed area. The coverage of the observed area is an important parameter for selection of topology protocol as stated in this paper. We see that the best coverage is obtained in the case of A3 coverage protocol. The values of the coverage coefficient given in Table 4 do not deviate significantly from the normal distribution, as shown by the following parameters - the K-S test statistic (D) is 0.28748 while the p-value is 0.71277.

Table 3. Simulation time for different topology protocols

Protocol	Time complexity
A3	31.28
A3 coverage	111.35
EECDS	134.86
CDS rule K	94.71

Table 4. Coverage of the observed area by the communication radius

Topology protocol	Coverage of the observed area	Coverage beyond the observed area
A3	0.9261	0.2253
A3 coverage	1	0.5037
EECDS	0.9617	0.298
CDS rule K	0.9184	0.2065

The crucial simulation result is the amount of traffic expressed as number of sent and received bytes for both cluster heads and regular nodes within the sensor network. The traffic depends heavily on the number of messages sent and received by sensor nodes, which is determined by topology protocol in use. From Table 5, we can see that EECDS protocol generates the most traffic, while A3 coverage protocol produces the least traffic.

Table 5. The total number of bytes sent per topology protocol

Protocol	Node type	Amount of sent data (byte)	Amount of received data (byte)	Total amount of sent messages (byte)
A3	Cluster head	2608	10589	4150
	Regular node	1542	10970	
A3 coverage	Cluster head	3906	19341	4309
	Regular node	403	4885	
EECDS	Cluster head	2440	12695	6050
	Regular node	3610	17146	
CDS rule K	Cluster head	1755	9661	4838
	Regular node	3083	17403	

The total number of sent bytes, given in Table 5, is one of the input parameters for the acceptable pairings model of topology and encryption algorithms, as shown in Section 6. The values in Table 5, representing the total number of sent messages in bytes do not deviate significantly from the normal distribution as shown by the following parameters - the K-S test statistic (D) is 0.29844 while the p-value is 0.67012.

The final results which combine simulation results explained in previous Tables, and mathematical model of the power consumption presented in Section 6 are given in Table 6. Table 6 displays power consumption per topology protocol per encryption algorithm. All combinations which are acceptable according to our decision-making system for application and implementation explained in Section 6 are shown in green, while those represented by red are algorithms that do not meet the energy efficiency criteria that we have set. By inspecting obtained results, we can clearly see the combination of Katan64 encryption and A3 protocols yields to the best results.

Table 6. Power consumption per topology protocol per encryption algorithm

	A3	A3 coverage	EECDS	CDS rule K
AES	🟩 61.753,12	🟩 63.524,16	🟥 136.074,04	🟥 113.562,02
Noekeon	🟩 59.724,73	🟩 61.437,60	🟥 131.604,45	🟥 109.831,88
LED 128	🟥 231.236,13	🟥 237.867,84	🟥 509.532,70	🟥 425.235,89
Present	🟩 54.997,08	🟩 56.337,12	🟥 120.678,80	🟩 100.713,76
Prince	🟩 52.733,83	🟩 54.018,72	🟥 115.712,59	🟩 96.569,16
Picolo	🟩 62.918,48	🟩 64.451,52	🟥 138.060,52	🟥 115.219,86
TWINE	🟩 75.819,03	🟩 77.666,40	🟥 166.367,89	🟥 138.844,08
Simon 64/96	🟩 75.140,05	🟩 76.970,88	🟥 164.878,03	🟥 137.600,70
KATAN 64	🟩 49.565,27	🟩 50.772,96	🟥 108.759,90	🟩 90.766,72

Here is another observation which shows the top 10 combinations of protocols and encryption algorithms in terms of energy efficiency and coverage:

1. A3 protocol with KATAN64 cipher
2. A3 coverage protocol with KATAN64 cipher
3. A3 protocol with Prince cipher
4. A3 coverage protocol with Prince cipher
5. A3 protocol with Present cipher
6. A3 coverage protocol with Present cipher
7. A3 protocol with Noekeon cipher
8. A3 coverage protocol with Noekeon cipher
9. A3 protocol with AES cipher
10. A3 protocol with Picolo cipher

8 Conclusion

In this paper, we have proposed a mathematical model for energy consumption estimation in wireless sensor networks. The estimation is based on the topology protocol and lightweight cryptographic algorithm in use. The energy estimation is the basis to make a decision of the appropriate combinations of the cipher algorithm and the topology protocol. The proposed mathematical model is based on three parameters: total number of messages sent throughout the entire system, energy cost for different encryption algorithms and coverage coefficient of topology protocol. We have also defined algorithm which can be used for energy estimation. We tested the proposed mathematical model in the simulation environment where WSN is applied for agriculture field monitoring of a typical size. Based on the defined model and the simulation results, a sorted list of combinations of topology protocols and cipher algorithm is presented. The proposed estimation method is good starting point to select appropriate topology protocol and cryptographic algorithm for implementation. As a future work, we will consider additional factors that can affect the accuracy of a mathematical model. Further, we will analyze the weighting effect of these additional factors on the final result.

Acknowledgments

This manuscript was supported by the Ministry of Education, Science, and Technological Development of the Republic of Serbia, within the project of technological development TR32023 – "Optimization of performance of energy efficient computer and communication systems".

References

[1] N. Deng, Study on Dynamic Characteristics of Train-bridge Coupling Based on Wireless Sensor Network, *Journal of Internet Technology*, Vol. 20, No. 2, pp. 555-562, March, 2019.

[2] J. Di, Investigation on the Traffic Flow Based on Wireless Sensor Network Technologies Combined with FA-BPNN Models, *Journal of Internet Technology*, Vol. 20, No. 2, pp. 589-597, March, 2019.

[3] E. M. Jovanovska and D. Davcev, No pollution Smart City Sightseeing Based on WSN Monitoring System, *2020 Sixth International Conference on Mobile And Secure Services (MobiSecServ)*, Miami Beach, Florida, USA, 2020, pp. 1-6.

[4] N. Kumar and B. Sharma, Opportunities and Challenges with WSN's in Smart Technologies: A Smart Agriculture Perspective, in: P. Singh, B. Bhargava, M. Paprzycki, N. Kaushal, W. C. Hong (Eds.), *Handbook of Wireless Sensor Networks: Issues and Challenges in Current Scenario's, Advances in Intelligent Systems and Computing*, Vol. 1132, Springer, pp. 441-463, March, 2020.

[5] H. Jawad, R. Nordin, S. Gharghan, A. Jawad and M. Ismail, Energy-Efficient Wireless Sensor Networks for Precision Agriculture: A Review, *Sensors*, Vol. 17, No. 8, Article number: 1781, August, 2017.

[6] J. Valente, D. Sanz, A. Barrientos, J. Cerro, A. Ribeiro and C. Rossi, An Air-Ground Wireless Sensor Network for Crop Monitoring, *Sensors*, Vol. 11, No. 6, pp. 6088-6108, June, 2011.

[7] T. Ojha, S. Misra, N. S. Raghuwanshi, Wireless sensor networks for agriculture: The state-of-the-art in practice and future challenges, *Computers and Electronics in Agriculture*, Vol. 118, pp. 66-84, October, 2015.

[8] F. Oliveira, R. Semente, J. Fernandes, T. Melo, S. Nascimento and A. Salazar, EEWES: An energy-efficient wireless sensor network embedded system to be applied on industrial environments, *Ingeniería e Investigación*, Vol. 35, No. 2, pp. 67-73, August, 2015.

[9] G. Pachnanda, K. Singh and L. Gangwar, Comparative analysis of A3, eecds and kneigh tree protocols in Wireless sensor networks, *International Journal of Electronics and Computer Science Engineering*, Vol. 2, No. 3, pp. 987-991, 2013.

[10] A. Bogdanov, L. Knudsen, G. Leander, C. Paar, A. Poschmann, M. Robshaw, Y Seurin and C. Vikkelsoe, PRESENT: an ultra-lightweight block cipher, in: P. Paillier, I. Verbauwhede (Eds.), *Cryptographic Hardware and Embedded Systems - CHES 2007: 9th International Workshop, Lecture Notes in Computer Science*, Vol. 4727, Springer, Berlin, Heidelberg, 2007, pp. 450-466.

[11] Z. Gong, S. Nikova and Y. W. Law, KLEIN: A new family of lightweight block ciphers, in: A. Juels, C. Paar (Eds.), *RFID. Security and Privacy: 7th International Workshop, RFIDSec 2011, Lecture Notes in Computer Science*, Vol. 7055, Springer, Berlin, Heidelberg, 2012, pp. 1-18.

[12] T. Suzaki, K. Minematsu, S. Morioka and E. Kobayashi, Twine: A lightweight, versatile block cipher, *ECRYPT Workshop on Lightweight Cryptography (LC11)*, Louvain-la-Neuve, Belgium, 2011, pp. 146-169.

[13] W. Wu and L. Zhang, *LBlock: A Lightweight Block Cipher*, Cryptology ePrint Archive, Report 2011/345, June, 2011.

[14] P. K. Kushwaha, M. Singh and P. Kumar, A Survey on Lightweight Block Ciphers, *International Journal of Computer Applications*, Vol. 96, No. 17, pp. 1-7, June, 2014.

[15] B. Bhushan and G. Sahoo, Recent Advances in Attacks, Technical Challenges, Vulnerabilities and Their Countermeasures in Wireless Sensor Networks, *Wireless Personal Communications*, Vol. 98, No. 2, pp. 2037-2077, January, 2018.

[16] Y. Liu, P. Geng, J. Yang and R. Chen, Analysis and improvement of backbone-based topology control for wireless sensor networks, *Journal of Computational Methods in Sciences and Engineering*, Vol. 19, No. 1, pp. 179-195, January, 2019.

[17] H. K. Qureshi, *Graph-theoretic channel modeling and topology control protocols for wireless sensor networks*, Ph. D. Thesis, City University London, London, UK, 2011.

[18] Y. Zeng, X. Jia and Y. He, Energy efficient distributed connected dominating sets construction in wireless sensor networks, *Proceedings of the 2006 ACM International Conference on Wireless Communications and Mobile Computing*, Vancouver, British Columbia, Canada, 2006, pp. 797-802.

[19] G. Hatzivasilis, K. Fysarakis, I. Papaefstathiou and C.

Manifavas, A Review of Lightweight Block Ciphers, *Journal of Cryptographic Engineering*, Vol. 8, No. 2, pp. 141-184, June, 2018.

[20] A. Moradi, A. Poschmann, S. Ling, C. Paar and H. Wang, Pushing the Limits: A Very Compact and a Threshold Implementation of AES, in: K. G. Paterson (Eds.), *Advances in Cryptology-EUROCRYPT 2011, Lecture Notes in Computer Science*, Vol. 6632. Springer, Berlin, Heidelberg, 2011, pp. 69-88.

[21] J. Daemen, M. Peeters, G. V. Assche and V. Rijmen, Nessie Proposal: NOEKEON, *First Open NESSIE Workshop*, Leuven, Belgium, 2000, pp. 1-30.

[22] T. Plos, C. Dobraunig, M. Hofinger, A. Oprisnik, C. Wiesmeier and J. Wiesmeier, Compact Hardware Implementations of the Block Ciphers mCrypton, NOEKEON, and SEA, in: S. Galbraith, M. Nandi (Eds.), *Progress in Cryptology - INDOCRYPT 2012, Lecture Notes in Computer Science*, Vol 7668, Springer, Berlin, Heidelberg, 2012, pp. 358-377.

[23] C. Rolfes, A. Poschmann, G. Leander and C. Paar, Ultra-lightweight implementations for smart devices- security for 1000 gate equivalents, in: G. Grimaud, FX. Standaert (Eds.), *Smart Card Research and Advanced Applications-CARDIS, Lecture Notes in Computer Science*, Vol. 5189, Springer, Berlin, Heidelberg, 2008, pp. 89-103.

[24] J. Guo, T. Peyrin, A. Poschmann and M. Robshaw, The LED Block Cipher, in: B. Preneel, T. Takagi (Eds.), *Cryptographic Hardware and Embedded Systems-CHES 2011, Lecture Notes in Computer Science*, Vol. 6917, Springer, Berlin, Heidelberg, 2011, pp. 326-341.

[25] G. Zhao, R. Li, L. Cheng, C. Li and B. Sun, Differential fault analysis on LED using Super-Sbox, *IET Information Security*, Vol. 9, No. 4, pp. 209-218, July, 2015.

[26] K. Shibutani, T. Isobe, H. Hiwatari, A. Mitsuda, T. Akishita and T. Shirai, Piccolo: An Ultra-Lightweight Blockcipher, in: B. Preneel, T. Takagi (Eds.), *Cryptographic Hardware and Embedded Systems - CHES 2011, Lecture Notes in Computer Science*, Vol. 6917, Springer, Berlin, Heidelberg, 2011, pp. 342-357.

[27] M. Cazorla, K. Marquet and M. Minier, Survey and Benchmark of Lightweight Block Ciphers for Wireless Sensor Networks, *IDEA*, Vol. 64, No. 128, July, 2013.

[28] C. D. Cannière, O. Dunkelman and M. Knežević, KATAN & KTANTAN - A Family of Small and Efficient Hardware-Oriented Block Ciphers, in: C. Clavier, K. Gaj (Eds.), *Cryptographic Hardware and Embedded Systems - CHES 2009, Lecture Notes in Computer Science*, Vol. 5747, Springer, Berlin, Heidelberg, 2009, pp. 272-288.

[29] T. Eisenbarth, Z. Gong, T. Güneysu, S. Heyse, S. Indesteege, S. Kerckhof, F. Koeune, T. Nad, T. Plos, F. Regazzoni, F. X. Standaert and L. O. tot Oldenzeel, Compact implementation and performance evaluation of block ciphers in ATtiny devices, in: A. Mitrokotsa, S. Vaudenay (Eds.), *Progress in Cryptology - AFRICACRYPT 2012, Lecture Notes in Computer Science*, Vol. 7374, Springer, Berlin, Heidelberg, 2012, pp. 172-187.

[30] L. Batina, A. Das, B. Ege, E. B. Kavun, N. Mentens, C. Paar, I. Verbauwhede and T. Yalçın, Dietary Recommendations for Lightweight Block Ciphers: Power, Energy and Area Analysis of Recently Developed Architectures, in: M. Hutter, J.-M. Schmidt (Eds.), *Radio Frequency Identification: Security and Privacy Issues, Security and Privacy Issues 9th International Workshop RFIDsec 2013, Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg, 2013, pp. 103-112.

[31] M. R. Albrecht, B. Driessen, E. B. Kavun, G. Leander, C. Paar and T. Yalçın, Block Ciphers- Focus On The Linear Layer (feat. PRIDE), in: J. A. Garay, R. Gennaro (Eds.), *Advances in Cryptology - CRYPTO 2014, Lecture Notes in Computer Science*, Vol. 8616, Springer, Berlin, Heidelberg, 2014, pp. 57-76.

[32] R. Beaulieu, S. Treatman-Clark, D. Shors, B. Weeks, J. Smith and L. Wingers, The SIMON and SPECK Lightweight Block Ciphers, *52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, San Francisco, California, USA, 2015, pp. 1062-1067.

[33] G. Upton and I. Cook, *A Dictionary of Statistics (2 rev. ed.)*, Oxford University Press, 2014.

[34] D. S. Dimitrova, V. K. Kaishev and S. Tan, *Computing the Kolmogorov- Smirnov Distribution when the Underlying cdf is Purely Discrete, Mixed or Continuous*, City Research Online - University of London, 2017.

Biographies

Nemanja Radosavljević received his master’s degree in Faculty of Organizational Sciences, Serbia 2009. He is currently a lecturer in The School of Computing, Union University. His main research direction is energetic efficiency in wireless sensor networks.

Djordje Babić obtained his undergraduate degree in 1999 at the School of Electrical Engineering, Belgrade. He defended his PhD at Tampere University of Technology, Finland, in 2004. Since 2008, he has been employed at the School of Computing, Belgrade. He has published over 30 articles in international journals and conferences.