Maximizing Wireless Sensor Network Lifetime by Communication/Computation Energy Optimization of Non-repudiation Security Service: Node Level versus Network Level Strategies

Huseyin Ugur Yildiz[a,*], Kemal Bicakci[a], Bulent Tavli[a], Hakan Gultekin[a], Davut Incebacak[b]

[a] *TOBB University of Economics and Technology, Ankara, Turkey*
[b] *Middle East Technical University, Ankara, Turkey*

Abstract

In a typical Wireless Sensor Network (WSN) application, the basic communication service is the transportation of the data collected from sensors to the base station. For prolonging the network lifetime, energy efficiency should be one of the primary attributes of such a service. The amount of data transmitted by a node usually depends on how much local processing is performed. As an example, in visual sensor networks the amount of image processing on the nodes affects the amount of data transmitted to the base station (*i.e.*, the higher the computation, the lower the communication and vice versa). Hence in order to improve energy efficiency and prolong the network lifetime this communication/computation energy trade-off must be analyzed. This analysis may be performed at the network-level (*i.e.*, all nodes in the network use the same strategy) or at a node level (*i.e.*, sensor nodes do not necessarily have identical strategies). The latter is more fine-grained allowing different nodes to implement different solutions. To guide designers in effectively using these trade-offs to prolong network lifetime, we develop a novel Mixed Integer Programming (MIP) framework. We show that the optimal node level strategy can extend

*Corresponding author. Tel.: +90 312 555 63 04
Email addresses: huyildiz@etu.edu.tr (Huseyin Ugur Yildiz), bicakci@etu.edu.tr (Kemal Bicakci), btavli@etu.edu.tr (Bulent Tavli), hgultekin@etu.edu.tr (Hakan Gultekin), idavut@metu.edu.tr (Davut Incebacak)

network lifetime more than 20 % as compared to a network-level optimal strategy. We also develop a computationally efficient heuristic to overcome the very high computational requirements of the proposed MIP model.

Keywords: wireless sensor networks, digital signature, network lifetime, energy efficiency, mixed integer programming, heuristic.

1. Introduction

Wireless Sensor Networks (WSNs) consist of a plurality of sensor nodes deployed over a geographical area which are, typically, equipped with low computational capacity processors, short range wireless transceivers, and limited energy resources (*i.e.*, batteries) for monitoring physical phenomena like temperature, humidity, acoustic vibrations, and light intensity [1]. Note that there are also WSNs consisting of mobile nodes [2], however, in this study we only consider WSNs with stationary nodes. Communication and computation are the two main energy consumption categories for a typical WSN, communication energy is, usually, assumed to dominate the energy expenditure of the sensor nodes [3]. The severe energy constraints imposed on WSNs lead designers to endeavor for energy efficient operation strategies and algorithms. Therefore, maximizing network lifetime by optimizing the energy dissipation of sensor nodes is considered to be the most important design goal for WSNs [4]. In this study, similar to [3, 5], and [6], the network lifetime is defined as the duration between the time network starts operating and the time when the first sensor node in the network exhausts all its energy and dies. The acronyms used throughout the text are explained in Table 1

We consider the following scenario to motivate our work. A WSN is deployed for surveillance purposes. Each node has a built-in camera periodically capturing still images. Transportation of data from the sensor nodes to the base station may involve the other nodes acting as relays (multi-hop communication). The base station processes the collected data and interacts with a user. We note that since image processing require the use of complex algorithms, vi-

Table 1: Acronyms used in this paper

Acronym	**Description**
DS	Digital Signature
WSN	Wireless Sensor Network
OTS	One time Signature
RSA	Rivest-Shamir-Adleman
$ECDSA$	Elliptic Curve Digital Signature Algorithm
LP	Linear Programming
MIP	Mixed Integer Programming
GSS	Golden Section Search
SA	Simulated Annealing

sual sensor nodes dissipate significantly higher power than scalar nodes. Besides, bandwidth requirements for carrying visual data are also much higher [7]. As a result, exploiting the communication/computation energy optimization tradeoff is more prominent in visual sensor networks.

In such an application, if the design goal is to prolong the network lifetime, then the following question becomes highly relevant. How much processing should be performed locally on the nodes? On one extreme, there may be no local processing at all and nodes simply transmit raw images. On the other extreme, nodes may run the most sophisticated image processing and machine vision algorithms so that only minimal amount of data is transmitted to the base station (*e.g.*, only semantic data pertaining to a suspicious situation). Other options may lie in the middle with a moderate amount of local processing (*e.g.*, images may be compressed before transmission).

Unlike computational costs, energy consumption of communication cannot be expressed with a simple formula because it depends not only on the distance between the node and the base station but also on the exact route taken which is not necessarily the shortest hop route or the minimum energy path. A Linear Programming (LP) model with the objective of maximizing the lifetime has

been proposed in an earlier work [4] which can be used to analyze the trade-off exemplified above. However, there is one issue which has not been investigated yet. Instead of adopting a single processing option applied to all nodes in the network, we may implement a hybrid solution, in which different nodes may use different processing options.

To explain this problem in more concrete terms, in our work we analyze non-repudiation security service[1] which has the highest energy overhead [8, 9] among all security-related techniques in WSNs [4]. Digital signature *(DS)* algorithms that could be used to implement a non-repudiation service include widely adopted Rivest Shamir Adleman *(RSA)* algorithm [10], Elliptic-Curve Digital Signature Algorithm *(ECDSA)* [11] - a robust and efficient alternative to RSA, and One-Time Signature *(OTS)* algorithm [12, 4] having a unique trade-off between computation and communication with its long signature sizes and practically zero computational cost.

Since ECDSA has better performance figures than RSA algorithm in terms of both computation and communication (see Table 3), we can easily say which one is more energy-efficient. On the other hand, as mentioned, it is not straightforward to tell whether ECDSA or OTS algorithm should be preferred in a given application. The problem becomes even more complicated if we may have the option to perform the processing (for generating ECDSA signatures) only in a subset of nodes in the network but not in the remaining ones (where OTS is used)[2].

In this paper our objective is to seek answers to the following research problems:

1. How much improvement in terms of network lifetime can we attain with a

[1] An example regarding security is chosen due to availability of energy overhead information from previous work [4] but our framework could be easily tailored to analyze other computation/communication trade-offs once the required numerical data is available.

[2] We acknowledge that this node level strategy brings additional complexity and may not always be applicable.

strategy in which different nodes may implement different options instead of all nodes use the same option? In more concrete terms, what is the impact of assigning a single digital signature algorithm to all nodes in the network globally instead of choosing the optimal algorithm for each node, separately?

2. Can we build a mathematical programming framework to uncover the research challenge posed in Question (1) without any optimality gap?
3. Is it possible to develop a computationally efficient heuristic algorithm which closely approximate the exact solution for the research problem in Question (2)?

To answer Question (1) and Question (2), we build a novel Mixed Integer Programming *(MIP)* framework. The computational complexity of the MIP model [13, 14] lead us to develop heuristic methods for finding an answer to Question (3). Heuristic methods are commonly used in applied optimization [15, 16]. Their main purpose is to ensure that optimization problems are solved in reasonable time (*i.e.*, reduce computational complexity) by providing feasible solutions close to the optimal solution, but without any guarantee of finding the exact optimal.

The rest of this paper is organized as follows: Section 2 summarizes the related work and our original contributions. Section 3 presents the MIP framework. Section 4 presents the results of numerical analysis of the MIP model. In Section 5, two heuristic algorithms are presented and their performances are investigated. In Section 6, we present two greedy algorithms and evaluate their performances through simulations. The conclusions of this study are presented in Section 7.

2. Related Work and Original Contributions

In this section, we first present a review of the studies on non-repudiation service and mathematical programming based analysis of energy efficiency and

network lifetime maximization in WSNs. Then, we provide our original contributions in a nutshell.

2.1. Related Work

The most similar work to our study is authored by Seys and Preneel [17], where they compare the energy consumption of different DS algorithms in low power devices. They recommended the use of ECDSA in most of the WSN designs because of its easy management and significantly lower power requirement than OTS. However, they did not analyze computational and communication costs of DS algorithms under a unified framework. Unlike this study which basically provides simple rule-of-thumb guidelines, we perform a comprehensive analysis of the DS algorithms within a unified MIP framework. Without the proposed framework, it would not be possible to quantify the impact on network lifetime and to make a systematic comparison of various strategies. For instance, only by using this framework we could show that ECDSA is not always a better choice than OTS.

Another study related to ours is authored by Saied *et al.* [18], where it is advocated that the use of DS algorithms between two IoT (Internet of Things) entities may be infeasible due to the highly demanding cryptographic primitives required to bootstrap them. As a solution they proposed to delegate the heavy cryptographic load to less constrained nodes in neighborhood exploiting the spatial heterogeneity of IoT environment. Obviously, such an approach necessitates the availability of highly capable devices within the close proximity of resource constrained devices. In our study we do not make such an assumption.

There are a number of independent works addressing the issues related with the energy costs of public key algorithms in resource limited devices [8, 9, 19, 20]. However, in these studies the impact of non-repudiation support on system lifetime was not investigated.

Before an overview of the literature on the investigation of various aspects of WSNs through mathematical programming, we present a brief introduction to mathematical programming, as follows.

As examples of mathematical programming models, both LP and MIP models are used to find the best solution considering a given set of constraints, which characterize the set of legitimate decisions [21]. Alternative decisions are compared based on their objective function values and the one with the best value (could be the smallest or the largest depending on the nature of the function) is selected as the optimal. Although they are used for the same reason, LP and MIP models cannot be used in place of each other in many settings. Hence, they should not be considered as alternatives. Basically, types of decisions to be made determine which type of a mathematical model should be used to model the problem under consideration. Namely, if the decision variables are required to take integral values, we need to use integer values.

LP models whose variables take continuous values are relatively easier to solve. This is due to the special geometry of the set of feasible solutions (called the feasible set) of LPs. The vertices of the feasible set are defined by the constraints of the model and it is known that, given a nonempty feasible set, there is always a vertex solution which is optimal. Hence the well-known Simplex Algorithm, which searches the optimal solution among the vertices greedily, is a quite effective solution method for LPs, on the average. Unfortunately, MIP models do not have such a property in general and hence call for more advanced solution algorithms such as branch-and-bound, branch-and-cut, etc. These methods guaranteeing an optimal solution are called exact solution methods. At each step of such algorithms, first the problem without the integrality restrictions on variables (*i.e.*, the LP relaxation) is solved. Then, if an integer variable (*e.g.*, the one which is actually required to be an integer in the original problem) has a fractional value in the current solution, then the problem is divided into two subproblems by setting that variable's value to the nearest integer values. Then, the new problems are solved recursively in the same manner until the optimal solution is found. This basic method can be improved and fastened by incorporating problem specific information in the subproblem creation step.

The literature on mathematical programming based modeling and analysis

of WSNs is extensive and has grown rapidly in recent years. We refer interested readers to the recent review papers on this topic [22, 23]. However, we briefly present studies on WSNs utilizing mathematical programming for optimizing the network lifetime which are most related to our study.

Ergen and Varaiya [5] investigated the lifetime achieved by two multi-hop routing schemes through an LP framework. The objective of one of the schemes is the maximization of lifetime while the other one's is the minimization of communication energy consumption. It is concluded that increasing transmission range has a profound impact on energy conservation in a WSN depending on the ratio of transmission energy to circuitry energy.

Alfieri *et al.* [24] investigated the effects of exploitation of sensor spatial redundancy in WSNs on network lifetime through an MIP framework. To allow sensors to save energy when inactive, they proposed a scheme to let only a subset of selected sensors be active in certain time periods. Both centralized (based on column generation) and distributed approaches (heuristic algorithm) were presented to maximize WSN lifetime.

Cheng *et al.* [3] explored a wide range of network deployment and data collection alternatives (*i.e.*, non-uniform node density, mobile sinks, and multiple sinks) to mitigate WSN hot spot problem through an LP framework. The objective function in their framework is to maximize the WSN lifetime. They considered a rich set of extra costs other than energy costs involved in more complex deployment and data collection strategies.

Bicakci *et al.* [25] investigated the impact of one-time energy costs on the overall system lifetime in WSNs through an MIP framework. Public-Key Cryptography is used as the representative one-time initialization operation. It is shown that the effect of public-key cryptography on optimal route selection depends on the ratio of electronics energy to amplifier energy.

Tavli *et al.* [26] developed an LP framework to model WSN lifetime when compression for data reduction is utilized. They proposed three data compression and forwarding techniques to jointly optimize data compression and flow balancing. The optimal strategy is shown to outperform pure strategies (*i.e.*,

always compress and never compress strategies) in terms of network lifetime. They also built a more sophisticated LP framework to investigate the effects of multi-level dynamic compression in conjunction to flow balancing [27].

Santos *et al.* [28] proposed a mathematical programming framework to model the problem of clustering a WSN topology as a variation of the independent dominating set problem. Data generated at a sensor node belonging to a certain cluster is transmitted to the cluster's clusterhead. Clusterheads sent the data they received from the cluster members to other clusterheads which are closer to the base station through gateway nodes (*i.e.*, nodes that can reach multiple clusterheads). Eventually all data terminate at the base station. In this communication structure, which is a typical data flow arrangement in clustered WSNs, optimal role assignment (*e.g.*, determination of clusterheads, membership to a particular cluster, gateway node selection) in cluster formation is vital for network lifetime maximization. The objective is the maximization network lifetime which is defined as the time when first sensor node runs out of energy. They also designed two heuristic methods to generate optimized WSN clustering.

Hoang *et al.* [29] investigated the problem of conserving energy by allowing sensors to exploit the inherent broadcast nature of the wireless channel (*i.e.*, joint data compression in a cluster-based wireless sensor network). When a particular sensor transmits its data to the clusterhead, other sensors can receive and use that data in order to compress their own data. They formulated the optimization problem for network lifetime maximization by considering the energy dissipation on transmission, reception, and compression. They also proposed a heuristic scheme which has lower complexity and near optimal performance.

Fateh and Manimaran [30] formulated an MIP for the joint scheduling of computation tasks and communication messages in data collection tree based networks. They also proposed a heuristic algorithm. The proposed heuristic produces results in close proximity of the optimal solutions.

Gu *et al.* [31] considered energy efficient routing and sleep scheduling, jointly, to maximize the lifetime of delay sensitive WSNs. However, the exact formu-

lation of the problem is a non-linear MIP which is extremely challenging to be solved even with a few nodes. Therefore, an LP based heuristic is devised to obtain a relaxed upper bound of the original problem.

El-Sherif *et al.* [32] investigated the problem of minimizing energy consumption in Video Sensor Networks by jointly optimizing the encoding power, the transmission power, and the source rate at each sensor node. They transformed the problem into a mathematical programming formulation and developed an efficient solution algorithm using Lagrange duality.

Note that mathematical programming based analysis of WSNs does not need to adopt lifetime maximization or energy minimization as objectives. Many other objective functions can be utilized. For example, Cheng *et al.* [33] presented two cross-layer design schemes and established optimization models to achieve minimum end-to-end delay in a multi hop wireless networks under interference constraints. Establishment of a novel sufficient condition for conflict-free transmission and formulation of an LP model for minimum interference routing are the main contributions of the study.

As exemplified in the preceding paragraphs on mathematical programming based analysis of WSNs, a suitable mathematical programming framework is the key to successfully analyze the problem under investigation. Therefore, for each unique problem a novel mathematical program should be created with proper objective function, constraint equations, variables, and parameters. Note that, it is also very desirable to build efficient heuristics for the solution of the problem under consideration. In this study, we investigate the effects of node-level strategies exemplified by the DS algorithm assignment problem in conjunction to data flow optimization for WSN lifetime maximization which is an important original problem and has never been investigated in the literature. Hence, the creation of a novel mathematical programming framework for the analysis of this problem is a major contribution of this study to the literature. Furthermore, we design an efficient polynomial time solution heuristic to solve the problem efficiently which is another novel contribution of this study.

In our problem, the objective function is to maximize the network lifetime.

To maximize the network lifetime, the amount of data packets flowing over each link should be optimized and the DS algorithm utilized at each sensor should be decided optimally so that over-utilization of any sensor node's energy resource is prevented (*i.e.*, network-wide energy dissipation is balanced). Indeed, energy efficient protocol design in WSNs is generally achieved by establishing a set of rules in the form of an algorithm. However, most of these algorithms are, essentially, heuristic algorithms that perform suboptimally. On the other hand, when the lifetime maximization problem is cast as an MIP model, then the results obtained by solving the MIP model are guaranteed to be the optimal solutions. Hence, in this study, we do not propose a new network protocol for enabling network lifetime maximization in WSNs supporting non-repudiation service. Instead, we analyze the concept of WSN non-repudiation support from energy efficiency perspective within a general framework and without going into the details of specific protocols or algorithms. The results we report in this study represent optimistic performance bounds because we do not include any protocol specific control message exchange in our analysis. Yet, our results reveal the benchmarks that can be used to evaluate and compare different protocols supporting non-repudiation in WSNs. In fact, a high level analysis is necessary to grasp the main trends in energy dissipation characteristics of various DS algorithms utilized in WSNs. Furthermore, the advantages of node-level decisions on DS algorithm selection over network-level DS-algorithm selection are quantitatively characterized. Such an analysis necessitates the use of certain abstractions and assumptions to eliminate the shadowing effects of implementation details not specifically related to the concept under investigation, *per se*.

2.2. Original Contributions

In this subsection, we clearly outline the original and novel contributions of this study in an itemized fashion as follows:

1. Energy efficiency in WSNs is of utmost importance and should be made a priority in all system design decisions in WSNs. The optimization of

energy expenditure usually requires a careful trade-off analysis. Our work is the first in the literature to distinguish between node level and network level optimization strategies and introduce a novel mathematical programming framework to compare quantitatively these two strategies.

2. Digital signature algorithms (*i.e.*, *RSA*, *ECDSA*, and *OTS*) are used in WSNs for non-repudiation security service. As being an important real life research problem, the investigation of energy efficiency of DS algorithms for WSNs becomes a focal point in this study. We investigate the scenario where different sensor nodes are allowed to select one of the available DS algorithms as opposed to the case where only one DS algorithm is utilized by all sensor nodes. In fact, network lifetime maximization by employing node-level decisions on the DS algorithm selection is a research topic of practical significance which has never been investigated in the literature before.
3. We build a novel MIP framework which is capable of capturing both communication and computation energy dissipation mechanisms of WSNs employing DS algorithms to characterize the benefits of utilizing the node-level DS selection approach instead of the network-level DS selection approach. Using an MIP based analysis framework has the distinct advantage of comparing different approaches under a unified framework by using the same set of assumptions. Furthermore, the MIP approach eliminates the possible shadowing effects brought by suboptimal approaches which enables us to focus on the main research problem (*i.e.*, our objective is not to investigate secondary effects like routing protocol behavior).
4. One disadvantage of solving MIP problems (especially the ones with a large number of variables) is the computational complexity. As a remedy, we create a novel heuristic algorithm to solve the proposed MIP model efficiently even with large problem instances without significant deviations from the optimal solutions.
5. In addition to the MIP based and heuristics based analysis of the problem, we develop two novel greedy algorithms to assess the performance of

node-level DS algorithm assignment approach. Simulations of the greedy algorithms also reveal the superior performance of node-level approach over network-level approach.

In summary, we contribute to the literature by presenting a comprehensive analysis of the impact of node-level and network-level strategies with respect to energy efficiency by provisioning non-repudiation service which captures the essence of both security and networking perspectives. We note that this paper is an extended version of a short contribution [34] and includes a more detailed experimental analysis, heuristic algorithms which approximate the optimal solutions, and simulations using greedy algorithms.

3. The MIP Model

In this study, we use a well known and heavily utilized energy model in WSN literature (Heinzelman-Chandrakasan-Balakrishnan –HCB– energy model) [35]. The amount of energy to transmit a bit from node i to node j is represented as $E_{tx,ij} = E_{Elec} + \varepsilon_{amp} \times d_{ij}^{\alpha}$, and the amount of energy to receive a bit is represented as $E_{rx} = E_{Elec}$ where E_{Elec} refers to the energy dissipation on electronic circuitry, ε_{amp} denotes the transmitter efficiency, α represents the path loss, and d_{ij} is the distance between node i and node j.

The network topology is represented as a directed graph $G(V, A)$ where we define V as the set of all nodes including the base station (node 1). We also define the set W which includes all nodes except the base station $W = V \setminus \{1\}$. The set of arcs is defined as $A = \{(i, j) : i \in W,\ j \in V - i\}$. Note that in this definition no node sends traffic to itself and the base station does not send out any traffic to the sensor nodes. The amount of traffic that flows from node i to node j is represented as f_{ij}. Time is organized into constant duration rounds. Each sensor node i creates the same amount of data traffic (s_i) at each round to be conveyed to the base station (*i.e.*, the amount of information bits generated at each node is the same). Traffic generated at each node terminates at the base station either by direct transfer or through other sensors acting as relay nodes.

The investigation of a lifetime optimization problem typically starts with the decision on the definition of the network lifetime. According to a common definition [3, 5, 6], the network lifetime is the duration between the time network starts operating and the time when the first sensor node in the network exhausts all its energy and dies. If a single node dies much sooner while the remaining nodes are left with plenty of energy then this definition of lifetime cannot capture the energy efficiency of a particular strategy. However, since the objective of our optimization problem implies that the lifetime of the node that dissipates the highest amount of energy is to be maximized, all nodes collaborate to avoid the premature death of any individual node by network wide sharing of the communication and computation burden in a balanced fashion. Therefore, this definition is a good metric that characterizes the energy efficiency of the investigated strategies.

The notation used for the decision variables and the parameters used throughout this paper is presented in Table 2.

The optimization problem in its general form is formulated with the objective of maximizing t (the minimum lifetime of sensor nodes) as follows:

$$\text{Maximize } t \tag{1}$$

subject to the following constraints:

$$\sum_{\substack{j\in V\\ j\neq i}} f_{ij} - \sum_{\substack{j\in W\\ j\neq i}} f_{ji} = s_i t + S_{c,i}, \quad \forall i \in W \tag{2}$$

$$E_{rx} \sum_{\substack{j\in W\\ j\neq i}} f_{ji} + \sum_{\substack{j\in V\\ j\neq i}} f_{ij}\ E_{tx,ij} + E_{c,i} \leq e_i, \quad \forall i \in W \tag{3}$$

$$S_{c,i} = s_i \times r \times t \times \{\sum_{k\in\mathcal{D}} o_1^k \times a_k^i\}, \quad \forall i \in W \tag{4}$$

$$E_{c,i} = s_i \times r \times t \times \{\sum_{k\in\mathcal{D}} o_2^k \times a_k^i\}, \quad \forall i \in W \tag{5}$$

$$\sum_{k\in\mathcal{D}} a_k^i = 1, \quad \forall i \in W \tag{6}$$

$$f_{ij} \geq 0, \quad \forall (i,j) \in A \tag{7}$$

Table 2: Terminology for MIP Formulations

Notation	**Description**
t	Network lifetime *(s)*.
f_{ij}	Flow from node i to node j *(bits)*.
s_i	Amount of data generated at node i *(bits)*.
E_{rx}	Energy consumption for receiving one bit of data *(J)*.
$E_{tx,ij}$	Energy consumption for transmitting one bit of data from node i to node j *(J)*.
d_{ij}	Distance between node i and node j *(m)*.
d_{int}	Inter-node distance *(m)*.
E_{Elec}	Energy dissipated in electronic circuitry *(J)*.
$S_{c,i}$	The amount of signature overhead at node i *(bits)*.
$E_{c,i}$	The amount of energy overhead at node i *(J)*.
ε_{amp}	Transmitters efficiency *(J)*.
α	Path loss exponent.
e_i	Battery energy of each sensor node *(J)* .
r	Signing rate.
$\mathcal{D}$	Set of DS algorithms. $\mathcal{D} = \{OTS, RSA, ECDSA\}$
o_1^k	Signature size of the DS algorithm-k *(bits)*.
o_2^k	Signature energy cost of the DS algorithm-k *(J)*.
a_k^i	Binary variable to determine which DS algorithm is used by node i. a_k^i takes the value of 1 if node i uses DS algorithm $k \in \mathcal{D}$, and 0 otherwise.
w_k^i	Continuous variable used to eliminate the product of lifetime and a_k^i for linearization.
M	Upper bound on lifetime t (lifetime obtained when DS is not employed).
$G = (V, A)$	Directed graph that represents network topology.
V	Set of nodes, including the base station.
W	Set of nodes, excluding the base station .

$$a_k^i \in \{0,1\}, \quad \forall i \in W, \forall k \in \mathcal{D} \tag{8}$$

Equation (2) is the flow balancing constraint which states that for all nodes except the base station, the difference between the amount of data flowing out of node i and the amount of data flowing into node i is equal to the amount of total data generated by node i which is the sum of information bits generated by node i throughout the lifetime ($s_i t$) and the signature overhead bits ($S_{c,i}$). Equation (3) is the energy constraint. Total energy dissipation in a sensor node is comprised of reception energy, transmission energy, and also the energy overhead ($E_{c,i}$) due to processing performed on the node. This equation states that no sensor node can spend more than its initial battery energy (e_i).

Equation (4) and Equation (5) give the total amount of signature overhead bits and computation energy overhead arising due to DS operations at node i, respectively. Only these equations needed to be modified to model other aspects of communication/computation trade-offs. The total number of signing operations during the entire network lifetime (t) is given by the term $s_i \times t \times r$, where r is the signing rate (*i.e.*, reciprocal of number of bits signed per signature). With a compressing factor of 8, a typical compressed image size in WSNs is 25344 bits [36] hence in our analysis we use the signing rate r as $1/25344$ unless stated otherwise. Signature sizes and signing energy costs of different DS algorithms are denoted by o_1^k and o_2^k, respectively. The results of the summations in both Equation (4) $-\{\sum_{k\in\mathcal{D}} o_1^k \times a_k^i\}-$ and Equation (5) $-\{\sum_{k\in\mathcal{D}} o_2^k \times a_k^i\}-$ are a single value of o_1^k and o_2^k for each node, respectively, due to the constraint over the binary variable a_k^i given in Equation (6). For example, if $a_1^3 = 1$, then according to this equation $a_2^3 = a_3^3 = 0$ and the only nonzero terms of the summations in Equation (4) and Equation (5) are o_1^1 and o_2^1, respectively (*i.e.*, node 3 utilizes OTS). Hence, for each node the optimal selection of the signature algorithm that maximizes the network lifetime is selected (*i.e.*, different nodes can implement different types of DS algorithms – node-level strategy). Equation (7) states that all flows within the network are non-negative. Equation (8)

enforces a_k^i to take binary values.

Multiplication of the continuous variable t and the binary variable a_k^i in Equations (4) and (5) makes the proposed model nonlinear. However, it can be transformed into an equivalent linear MIP model. We opt to present the nonlinear optimization problem first since it is easier to comprehend the nonlinear formulation.

In order to linearize this model we need to define new variables and include a set of new constraints. We define w_k^i as the multiplication $t \times a_k^i$ and replace the multiplication terms in the corresponding constraints in order to get the following:

$$S_{c,i} = s_i \times r \times \{\sum_{k \in \mathcal{D}} o_1^k \times \underbrace{w_k^i}_{t \times a_k^i}\}, \quad \forall i \in W \tag{9}$$

$$E_{c,i} = s_i \times r \times \{\sum_{k \in \mathcal{D}} o_2^k \times \underbrace{w_k^i}_{t \times a_k^i}\}, \quad \forall i \in W \tag{10}$$

Then, we include the following constraints to ensure that the continuous variable possess the characteristics of its constituent variables (*i.e.*, $w_k^i = t$ for only one value of k and it is zero for all other values at each node).

$$w_k^i \geq 0, \quad \forall i \in W, \forall k \in \mathcal{D} \tag{11}$$

$$w_k^i \leq t, \quad \forall i \in W, \forall k \in \mathcal{D} \tag{12}$$

$$w_k^i \leq M \times a_k^i, \quad \forall i \in W, \forall k \in \mathcal{D} \tag{13}$$

$$w_k^i \geq t - M \times (1 - a_k^i), \quad \forall i \in W, \forall k \in \mathcal{D} \tag{14}$$

In these constraints M is a big number which represents an upper bound on t. If $a_k^i = 0$, Equations (11) and (13) enforce $w_k^i = 0$. On the other hand, if $a_k^i = 1$, Equations (12) and (14) enforce $w_k^i = t$.

As a result, the linearized MIP model is formulated with the objective of maximizing t subject to the constraints stated in Equations (2), (3), (6)–(14).

Generally speaking, WSNs are assumed to be consisting of stationary sensor nodes and unlike mobile ad hoc networks topology changes are not frequent. Thus, topology discovery and route creation are one-time operations and for

substantial amount of time (rounds/epochs) these functions are not repeated [37]. If the epoch durations (network reorganization cycle time) are long enough then the energy costs of these operations constitute a small fraction (less than 1 %) of the total network energy dissipation [38]. On the other hand, in highly dynamic topologies network organization energy costs can be as high as 60 % of the total energy dissipation [39]. Considering that routing overhead can be neglected in stationary WSNs without leading to significant underestimation of total energy dissipation, our characterization of energy overhead due to route diversity is based on realistic assumptions.

4. Analysis

In this section, we analyze the trade-offs discussed in Section 1 in more concrete terms by investigating the use of DS algorithms in WSNs via node-level and network-level strategies. We select the following digital signature algorithms in our work:

RSA: RSA system is the most widely used public-key cryptosystem today and has often been called a de facto standard. Although it is not well-suited to resource-constraint environments such as WSNs, because of its well-established implementations and high-quality standards it is still an applicable choice in practice [40]. We also note that RSA patents have expired in 2000, while some elliptic curve patents are still alive.

ECDSA: With ECDSA, we can get the same level of security as RSA with smaller keys. Smaller keys are better than larger keys especially when bandwidth is a scarce resource, as in WSNs. This is why many previous work extensively study the use of ECDSA for WSN applications [8, 9, 17].

OTS: OTS offers a unique tradeoff between computation and communication with its long signature sizes and practically zero computational cost [4]. Although it was invented by Lamport long time ago [12], due to computational advantages recent years have witnessed a rapid surge of interest in OTS among the WSN research community. Many OTS variants such as BIBA, HORS, HOR-

SIC and SCU specifically addressing the needs for resource-demanding WSN applications were proposed recently (see [20] for an overview of these schemes).

To take the level of security provided by the DS algorithms into account, we consider two cases where DS algorithms are categorized at two different security levels: 80 bits and 112 bits. The signature parameters required for this work are listed in Table 3 [9, 41].

We will now prove that the RSA algorithm is never used in the optimal solution .

Lemma 1. *For any life time maximization problem instance defined by Equations (2), (3), (6)–(14) there exists an optimal solution in which none of the nodes uses the RSA algorithm.*

Proof. Let's assume that there exists an optimal solution in which at least one of the nodes use RSA algorithm. Without loss of generality, let h indicates this node. Let f^*_{ij} denote the optimal flow from node i to j. Since this solution is optimal, then these flows must satisfy Equations (2) and (3). If we rewrite these equations for node h, we have the following:

$$\sum_{\substack{j\in V\\ j\neq h}} f^*_{hj} = \sum_{\substack{j\in W\\ j\neq h}} f^*_{jh} + s_h t + S^{RSA}_{c,h}, \tag{15}$$

$$E_{rx} \sum_{\substack{j\in W\\ j\neq h}} f^*_{jh} + \sum_{\substack{j\in V\\ j\neq h}} f^*_{hj}\ E_{tx,hj} + E^{RSA}_{c,h} \leq e_h, \tag{16}$$

$S^{RSA}_{c,h}$ and $E^{RSA}_{c,h}$ values are calculated using o^2_1 and o^2_2 values given in Table 3 inside Equations (9) and (10), respectively. Now let's change the DS algorithm of node h from RSA to ECDSA. Let's assume that the routes of the flows are not changed, but their magnitudes can be different. Let $\hat{f}_{ij}$ denote these new flow values. For node h Equation (15) becomes $\sum_{j\in V, j\neq h} \hat{f}_{hj} = \sum_{j\in W, j\neq h} \hat{f}_{jh} + s_h t + S^{ECDSA}_{c,h}$. Since the routes of the flows didn't change, all incoming flows into node h remain the same. That is, $\hat{f}_{jh} = f^*_{jh}, \forall j, h$. Additionally, $S^{ECDSA}_{c,h} < S^{RSA}_{c,h}$. Then we can find new values of $\hat{f}_{hj}$ that still satisfy the equation and

$\hat{f}_{hj} \leq f^*_{hj}$, $\forall h, j$. As a consequence, the magnitudes of all outgoing flows from node h is reduced by changing its DS algorithm from RSA to ECDSA.

Secondly, after changing node h to ECDSA since $\hat{f}_{jh} = f^*_{jh}$, $\hat{f}_{hj} \leq f^*_{hj}$, $\forall h, j$, and $E^{ECDSA}_{c,h} < E^{RSA}_{c,h}$, we have

$$\begin{aligned} & E_{rx} \sum_{\substack{j \in W \\ j \neq h}} \hat{f}_{jh} + \sum_{\substack{j \in V \\ j \neq h}} \hat{f}_{hj} \cdot E_{tx,hj} + E^{ECDSA}_{c,h} \\ \leq \quad & E_{rx} \sum_{\substack{j \in W \\ j \neq h}} f^*_{jh} + \sum_{\substack{j \in V \\ j \neq h}} f^*_{hj} \cdot E_{tx,hj} + E^{RSA}_{c,h} \leq e_h. \end{aligned}$$

Therefore, energy constraint is still satisfied for node h after making the change.

Lastly, for all nodes other than h, since their DS algorithms and the routes of the flow are not changed but only the outgoing flows from node h are reduced, their energy usage will not be increased. This means that the flow and energy constraints are satisfied for all nodes on the network. Hence, if the old solution is optimal, then the new solution is also optimal. □

As a consequence of this theorem we know that the RSA algorithm will never be used in the optimal solution. However, in order to give idea about its performance we still use it in our simulation studies.

Communication energy parameters provided in [35] are used in our analysis which are presented in Table 4. Initial battery energy for each sensor node is equal and $e_i = 243\ J$ which is 75% of the energy available from a 30 mAh battery assuming 25% is spent for other tasks like sensing, compressing, route discovery, etc. [4].

We assume that all sensor nodes create raw data at a constant rate ($s_i =$ 3600 bits/hour) to be conveyed to the base station. We use GAMS IDE version 23.9.1 system [42] together with CPLEX version 12.4 solver for numerical evaluation of the mathematical programming models.

Table 3: Signature Parameters

k	DS	Signature Size *(bits)* o_1^k	Cost of DS *(mJ)* o_2^k
1	OTS-80	3120	0
2	RSA-1024	1024	304
3	ECDSA-160	320	22.82
1	OTS-112	6160	0
2	RSA-2048	2048	2302.7
3	ECDSA-224	448	61.54

Table 4: Energy Parameters

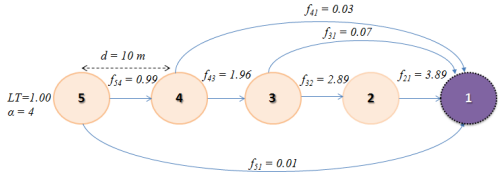
Description	**Symbol**	**Value**
Electronics Energy	E_{Elec}	50 nJ
Amplifier Energy	ε_{amp}	100 pJ
Sensor Battery	e_i	243 J

4.1. Toy Example

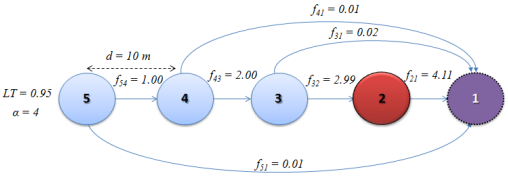
To illustrate the details of the basic trade-offs, we first present our analysis on a small scale network topology (*i.e.*, *toy example*) depicted in Fig. 1. This network topology consists of five nodes where the distance between adjacent nodes (d_{int}) is 10 meters, path loss exponent (α) is 4, and 80-bit security level is chosen. In Fig. 1, node 1 is designated as the base station and located at $(0, 0)$. Nodes 2, 3, 4, and 5 are located at $(-10, 0)$, $(-20, 0)$, $(-30, 0)$, and $(-40, 0)$, respectively. Maximum system lifetime is obtained when flows between nodes are balanced such that premature sensor deaths are avoided (*i.e.*, all the sensor nodes in the network cooperate in such a way that all the sensors drain out their energies approximately at the same time, which is the lifetime of the network).

The flows in Fig. 1a are obtained by using the case where no DS algorithm is utilized (*i.e.*, without DS case). The flows in Fig. 1b are obtained by using the aforementioned MIP model. The flows in Fig. 1c, Fig. 1d, and Fig. 1e are obtained by using the LP version of the MIP model. To convert the MIP model into an LP model, the binary variables are treated as parameters. If all nodes employ OTS then all a_1^i's are set to unity and all a_2^i's and a_3^i's are set to zero, therefore, only the continuous variables f_{ij}'s are left. Likewise, ECDSA and RSA are used solely by setting a_2^i's and a_3^i's to one, respectively.

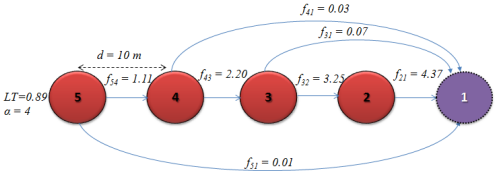
In summary, the optimization problem where no DS algorithm is used is obtained by the objective of maximizing t and Equations (2), (3), and (7). Furthermore, $S_{c,i}$ and $E_{c,i}$ are set to zero so that the cost associated with DS algorithms are annulled. As explained previously, the linearized MIP model (the node-level strategy) is obtained by the objective of maximizing t and Equations (2), (3), (6)–(14). The network-level strategy of utilizing OTS for all nodes is obtained by the objective of maximizing t and treating the binary variables as parameters. In the case of network-level strategy with OTS, all a_1^i's are set to unity and all a_2^i's and a_3^i's are set to zero. Therefore, in this strategy, Equations (2), (3), (4), (5), and (7) are used as constraints. The only difference for network-level strategies with ECDSA and RSA is setting only a_2^i's or a_3^i to unity and zeroing other binary variables.



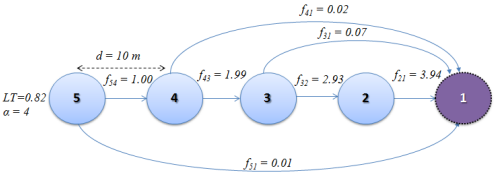
(a) Without DS case



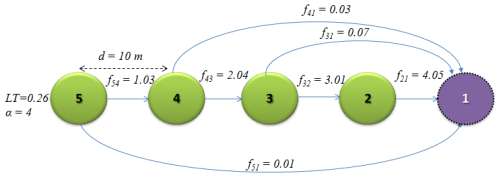
(b) Node level strategy



(c) OTS-80 is employed at network-level



(d) ECDSA-160 is employed at network-level



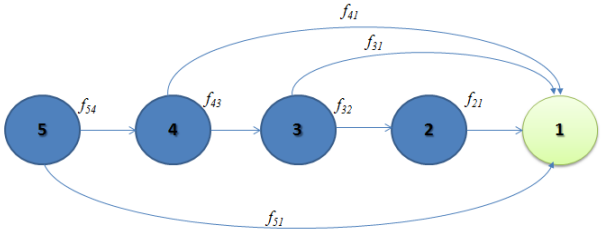
(e) RSA-1024 is employed at network-level

Figure 1: Normalized flows in a linear network topology and corresponding normalized lifetime (LT) values. Nodes visualized with red, blue, and green colors indicate that the node is using OTS-80, ECDSA-160, and RSA-1024, respectively.

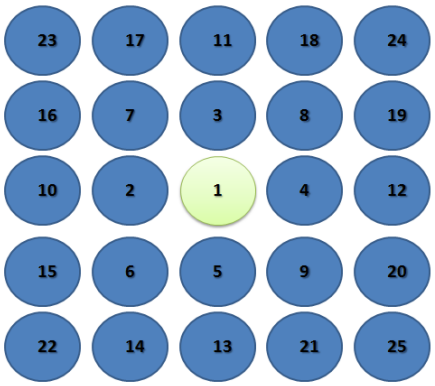
Absolute lifetime values are normalized (LT) with the base network lifetime[3] obtained by using the case where no DS algorithm is utilized. Note that in Fig. 1a, flows are normalized by assuming that each node generating unit amount of data. In Fig. 1a, all nodes except node 2 divide their data (the data they generate plus data flowing from other nodes) into two parts. Only a small fraction of the data is transmitted directly to the base station and the remaining large fraction of data is relayed to the closest node to the base station (*e.g.*, node 5 transmits only 1 % of its data to the base station and relays 99 % of its data to node 4). Such behavior is necessary to keep the cost of communicating data low while balancing energy dissipation throughout the network.

As illustrated in Fig. 1b, the node-level strategy assigns ECDSA to farther nodes from the base station and OTS to the closest node to the base station because the cost of communicating data to the base station is lower for the nodes closer to the base station (*i.e.*, OTS creates more overhead, yet, the computation cost is low). Note that node 5 uses ECDSA as its DS algorithm thus creating 1 % overhead (the sum of the data created by node 5 adds up to 1.01 units of which 0.01 units is due to the ECDSA overhead), however, if it were to use OTS algorithm then the overhead to be transported would be 12 % (the amount of data created by node 2 adds up to 1.12 units of which 0.12 units is due to OTS overhead). Node 5 splits its total flow (*i.e.*, total flow is the summation of a unit amount of generated data and signature overhead caused by ECDSA algorithm utilized on that node) into two parts instead of relaying all of its data via node 4 to the base station which results in less energy dissipation for node 5. The effects of signature overheads caused by DS algorithms can be observed in Fig.1b to 1e where it is evident that OTS has the highest signature overhead than other DS algorithms (*i.e.*, 3 times greater than RSA, and 12 times greater than ECDSA) while ECDSA has the lowest overhead.

[3]For our motivating scenario given in Section 1 (*i.e.*, WSN deployed for surveillance), the base network lifetime would refer to the ultimate lifetime goal corresponding to the hypothetical case of no data transmission overhead and no local processing.



(a) Illustration of a linear sensor network topology. Node 1 is the base station. Data flows from node i to node j are shown with f_{ij}'s.



(b) Illustration of a grid sensor network topology. Node 1 is the base station.

Figure 2: Linear and grid network topologies used throughout this paper.

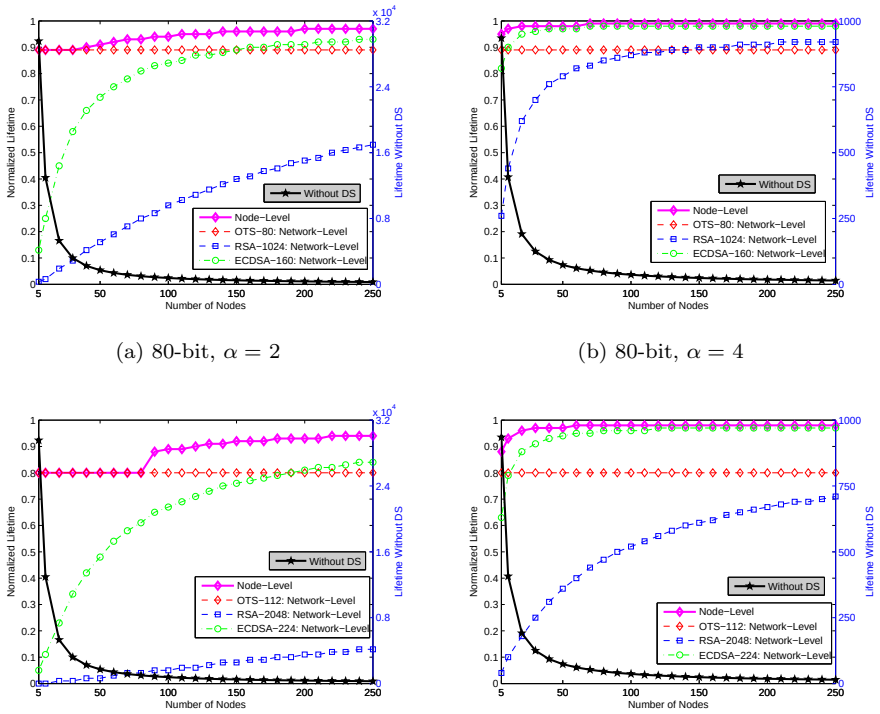
Normalized network lifetime values for the node-level strategy and network-level strategies using OTS-80, ECDSA-160, and RSA-1024 are 0.95, 0.89, 0.82, and 0.26, respectively which bolsters our hypothesis that an improvement in network lifetime is possible when hybrid usage of DS algorithms is allowed. Node level strategy can extend the lifetime up to 6.74% when compared to the best network-level strategy (OTS-80) for this topology. Furthermore, maximum lifetime is achieved if node 2 chooses OTS-80 while remaining nodes utilize ECDSA-160. RSA-1024 algorithm is not used by any sensor node in Fig. 1b because it has higher energy cost than ECDSA both in terms of computation and communication.

4.2. Linear Topology

In subsection 4.1 we illustrate a comparative analysis of basic mechanisms for energy balancing. Although it is instructive to work on a small size network, investigation of trade-offs in larger networks is also necessary. Thus, we present an analysis of node-level and network-level strategies by using larger linear topologies in this section. Note that highway safety and traffic monitoring applications are possible utilization examples for linear sensor network topologies [3].

Linear topologies are assumed to consist of N sensor nodes that are equally spaced on a line and the base station is at one end of the line (Fig. 2a). The distance between adjacent nodes is kept consant ($d_{int} = 10m$). We compare network lifetimes obtained with node-level and network-level strategies for various network sizes, path loss exponents, and security levels.

In Figs. 3a and 3b, normalized network lifetimes as functions of number of nodes is presented with 80-bit security level (*i.e.*, OTS-80, RSA-1024, and ECDSA-160 algorithms are utilized) for $\alpha = 2$ and $\alpha = 4$, respectively. Normalization is achieved by dividing the lifetime values for DS algorithms with the lifetime value of the without DS case (on the left y axis). The absolute value of the without DS case is presented on the right y axis in terms of hours. For $\alpha = 2$ (Fig. 3a), node-level strategy can extend the lifetime up to 8.99% as compared to the network-level strategy (OTS-80 or ECDSA-160). In most cases, nodes closer to the base station use OTS-80 and nodes farther away prefer ECDSA-160 when node-level strategy is utilized. As proved in Lemma 1, no sensors choose RSA-1024 when the MIP model is used. Network lifetime is always prolonged by the node-level strategy in networks with more than 30 nodes. For $\alpha = 4$ (Fig. 3b), communication becomes more costly, thus, lifetime values drop harshly with respect to $\alpha = 2$ which indicates that transmission of signature overhead affects the overall energy dissipation more significantly. Hence, DS algorithms which have shorter signature sizes perform better. Nevertheless, node-level strategy can extend the lifetime up to 11.24% as compared to OTS-80 or ECDSA-160.



(a) 80-bit, $\alpha = 2$

(b) 80-bit, $\alpha = 4$

(c) 112-bit, $\alpha = 2$

(d) 112-bit, $\alpha = 4$

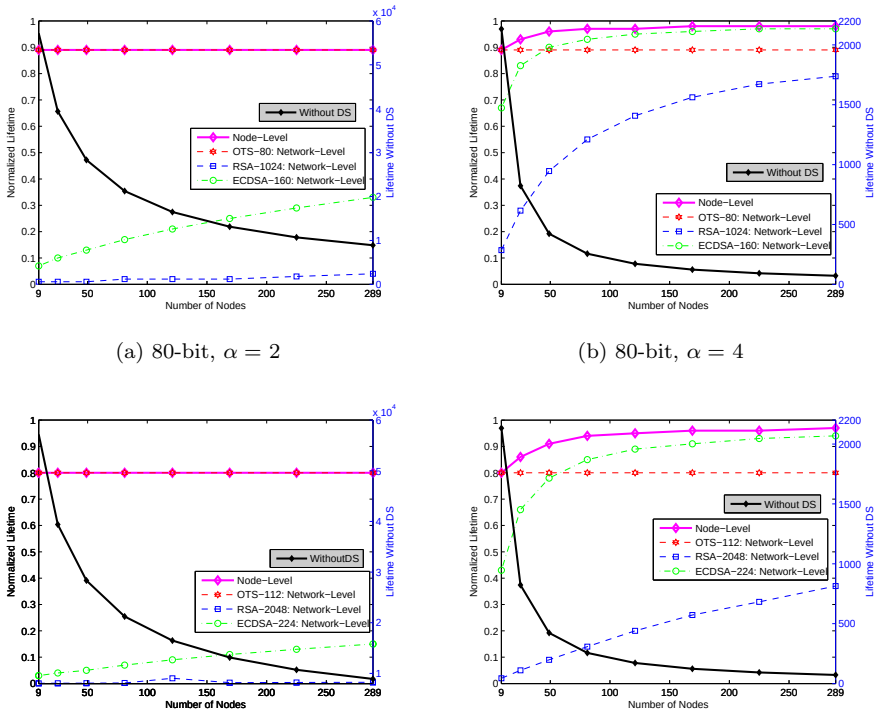
Figure 3: Normalized lifetimes for $\alpha = 2$ and $\alpha = 4$ in the linear network topology for 80-bit and 112-bit security levels.

To observe the effects of changing the security level from 80 bits to 112 bits, we repeat our analyses by using the same parameter sets and network topology. However, in this case, OTS-112, RSA-2048, and ECDSA-224 algorithms are utilized by selecting appropriate o_1^k and o_2^k parameters given in Table 3. For $\alpha = 2$ (Fig. 3c) node-level strategy can extend the lifetime up to 17.50% as compared to the OTS-112 or ECDSA-224. Nodes closer to the base station use OTS-112 and nodes farther away prefer ECDSA-224 when node-level strategy is employed. Network lifetime is always prolonged by the node-level strategy in networks with more than 80 nodes. For $\alpha = 4$ (Fig. 3d) network lifetime values obtained with node-level strategy is always higher than the network lifetime values obtained by using any network-level strategy. In fact, node-level strategy can extend the lifetime up to 22.50% when compared to OTS-112 or ECDSA-224.

4.3. Grid Topology

Even though it is instructive to analyze the effects of non-repudiation on network lifetime in linear WSN deployments, many practical sensor network topologies are envisioned to be deployed in two dimensional topologies. So it is necessary to perform analysis in two dimensional networks which grid topology is a canonical example. Note that in WSN literature the use of grid networks as representative two dimensional topologies is a commonly employed scenario [24, 43, 44, 45]. In this subsection, we repeat the analysis performed in the previous subsection in grid topologies where the base station is placed at the center of a grid area. The distance between adjacent nodes are fixed to 10 meters (Fig. 2b). The number of nodes are chosen as 9, 25, 49, 81, 121, 169, 225, and 289 such that an evenly distributed grid topology can be realized.

In Fig. 4a and Fig. 4b, path loss exponent values are $\alpha = 2$ and $\alpha = 4$, respectively. For $\alpha = 2$ lifetime values obtained with node-level strategy are exactly same as the network-level strategy with OTS-80 (*i.e.*, RSA-1024 and ECDSA-160 are never chosen by nodes at node-level strategy). For $\alpha = 4$ node-level strategy can extend the lifetime up to 10.11% when compared to OTS-80



(a) 80-bit, $\alpha = 2$

(b) 80-bit, $\alpha = 4$

(c) 112-bit, $\alpha = 2$

(d) 112-bit, $\alpha = 4$

Figure 4: Normalized lifetimes for $\alpha = 2$ and $\alpha = 4$ in the grid network topology for 80-bit and 112-bit security levels.

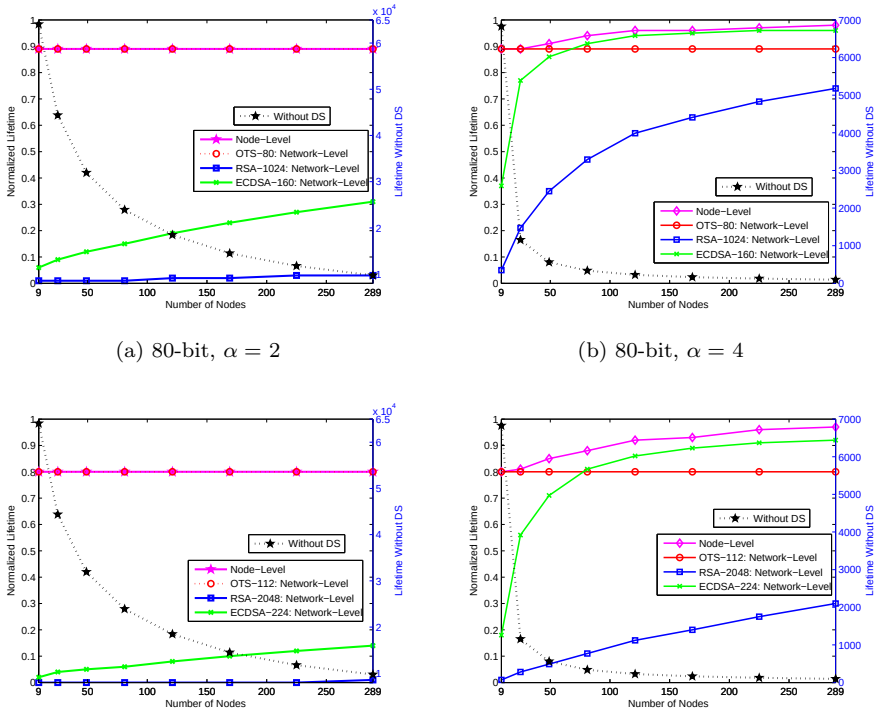
or ECDSA-160. System lifetime is always prolonged in networks which comprise more than 9 nodes by the node-level strategy.

To quantify the effects of changing the security level from 80 bits to 112 bits in grid topology we performed analysis by using OTS-112, RSA-2048, and ECDSA-224 algorithms (o_1^k and o_2^k parameters are presented in Table 3). For $\alpha = 2$ (Fig. 4c) network lifetime values obtained with node-level strategy are exactly same as the network-level strategy using OTS-112 algorithm which are also observed for 80-bit security level. For $\alpha = 4$ (Fig. 4d), node-level strategy can extend the lifetime up to 21.25% when compared to OTS-112 or ECDSA-224.

4.4. Random Topology

Random node deployment over a two dimensional sensing area is a frequently employed node deployment scenario in WSNs. Hence, in this section, we consider a network topology where we use a square shaped network consisting of N sensor nodes which are randomly placed within the square by utilizing a uniform distribution and a base station at the center. The results are averaged over 100 independent runs (*i.e.*, 100 randomly generated topologies). The number of deployed nodes is varied between 9 and 289 nodes as in the grid topology case (Section 4.3). In fact, the area of deployment when N is fixed is the same for both random square and deterministic grid topologies. For example, with $N = 9$, the edge of the grid is 20 meters in the deterministic grid topology which is also the case for the random node deployment (*i.e.*, nine sensor nodes are randomly placed within a 20 m by 20 m area). This pattern eventually provides an edge length of 160 meters for $N = 289$.

Normalized lifetimes for $\alpha = 2$ and $\alpha = 4$ in the random network topology for 80-bit and 112-bit security levels are presented in Fig. 5. For $\alpha = 2$ lifetime values obtained with node-level strategy are exactly same as the network-level strategy with OTS-80 and OTS-112 which suggest that when path loss exponent is low, it is preferable to use OTS. For $\alpha = 4$, the node-level strategy performs better than the network-level strategies. In fact, normalized network lifetimes



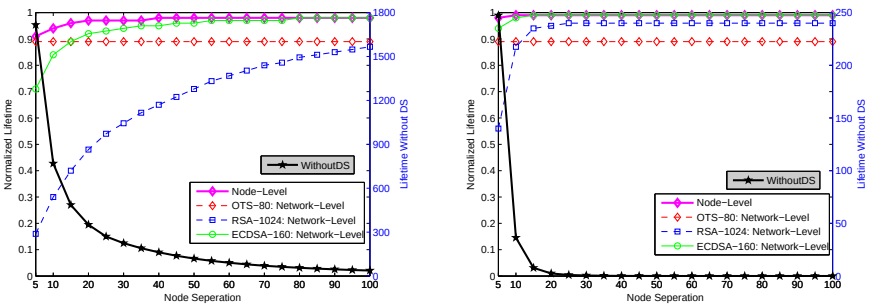
(a) 80-bit, $\alpha = 2$

(b) 80-bit, $\alpha = 4$

(c) 112-bit, $\alpha = 2$

(d) 112-bit, $\alpha = 4$

Figure 5: Normalized lifetimes for $\alpha = 2$ and $\alpha = 4$ in the random network topology for 80-bit and 112-bit security levels.



(a) $\alpha = 2$ (b) $\alpha = 4$

Figure 6: Normalized lifetimes as a function of node separation for $\alpha = 2$ and 4 in the linear network topology for 80-bit security level.

of node-level strategy and OTS-80 are 0.98 and 0.89, respectively, for $N = 289$. Furthermore, the difference in normalized lifetime increases as the level of security increases from 80 bits to 112 bits. For example, normalized network lifetimes with $N = 225$ are 0.96, 0.91, 0.80, and 0.25 for the node-level strategy, ECDSA-224, OTS-112, and RSA-2048, respectively. Nevertheless, the behaviors of the DS algorithms and strategies do not exhibit significant variations between the random and deterministic two dimensional node distributions.

4.5. Changing Network Density

In this subsection, we investigate the effects of inter-node separation (d_{int}) on the performance of the DS algorithms. In Fig. 6a and Fig. 6b, lifetime values obtained by using network-level strategies (OTS-80, ECDSA-160, and RSA-1024) and node-level strategy are plotted against the inter-node distance for $\alpha = 2$ and $\alpha = 4$, respectively. The number of nodes is set to 100.

Increasing inter-node distance has similar effects as increasing the number of nodes. While the network gets sparser, communication energy becomes more dominant when using $\alpha = 4$; hence, computational overhead (o_1^k) of ECDSA-160 and RSA-1024 algorithms becomes less significant. For $d_{int} = 100m$ with $\alpha = 2$ and $\alpha = 4$, normalized lifetimes obtained with network-level strategy are found

as 0.98 and 0.99, respectively which are $10.11\% - 11.24\%$ higher than the case where OTS-80 is employed.

4.6. Changing Signing Rate

Signing rate has a profound impact on the energy dissipation of sensor nodes, especially for the strategies that have higher signing cost. Therefore, in this subsection we explore the impact of changing the signing rate on the lifetime of WSNs utilizing the node-level strategy and network-level strategies. We used three signing rates ($r = 1/253440$, $r = 1/25344$, and $r = 1/2534.4$) in a linear topology with 100 nodes and 80-bit security level.

The top panel in Figs. 7 presents network lifetime in terms of hours when $\alpha = 2$ and bottom panel presents this case when $\alpha = 4$, respectively. For each strategy, three bars with grey, white and black colored are presenting the network lifetimes by using three signing rates.

Changing the signing rate has a significant impact on the lifetime depending on which algorithm is employed at network-level. For example, decreasing signing rate by an order of magnitude leads to up to 2.5 times longer lifetimes when RSA-1024 is employed at network-level. However, for ECDSA-160 and OTS-80, this improvement is not that significant. The most important effect of increasing the signing rate for the node-level strategy (*i.e.* MIP) is that more nodes utilize ECDSA-160 instead of OTS-80, which is especially significant for $\alpha = 4$.

4.7. Limited Transmission Range

Up to this point we assume that all nodes can reach each other (*i.e.*, there is no transmission range limitation), however, in many practical WSNs, transmission ranges are limited. Furthermore, having non-zero flows with only immediate neighbors is a commonly employed strategy. To analyze such a case, we revisit the grid topology with inter-node distance of 10 m presented in Subsection 4.3 by employing a maximum transmission range of 15 m instead of an unlimited transmission range (*i.e.*, $f_{ij} = 0$ if $d_{ij} \geq 15$ m). By using such an additional

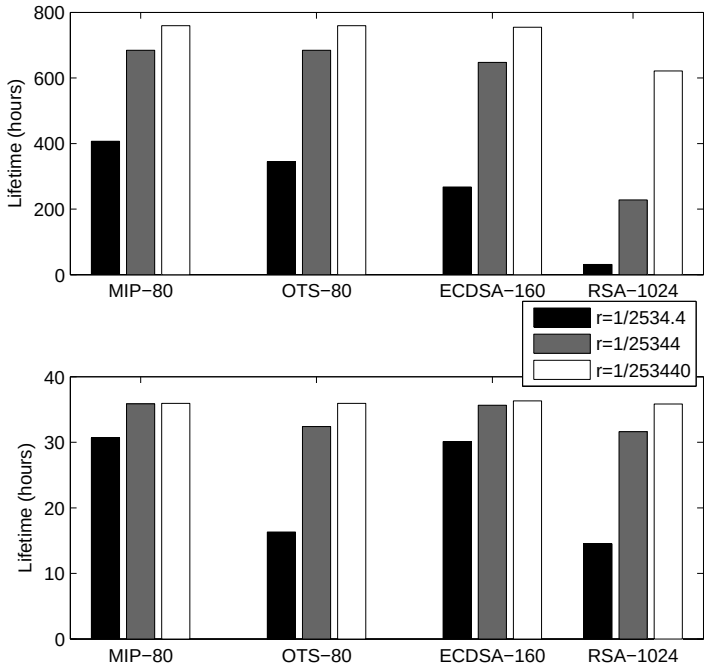
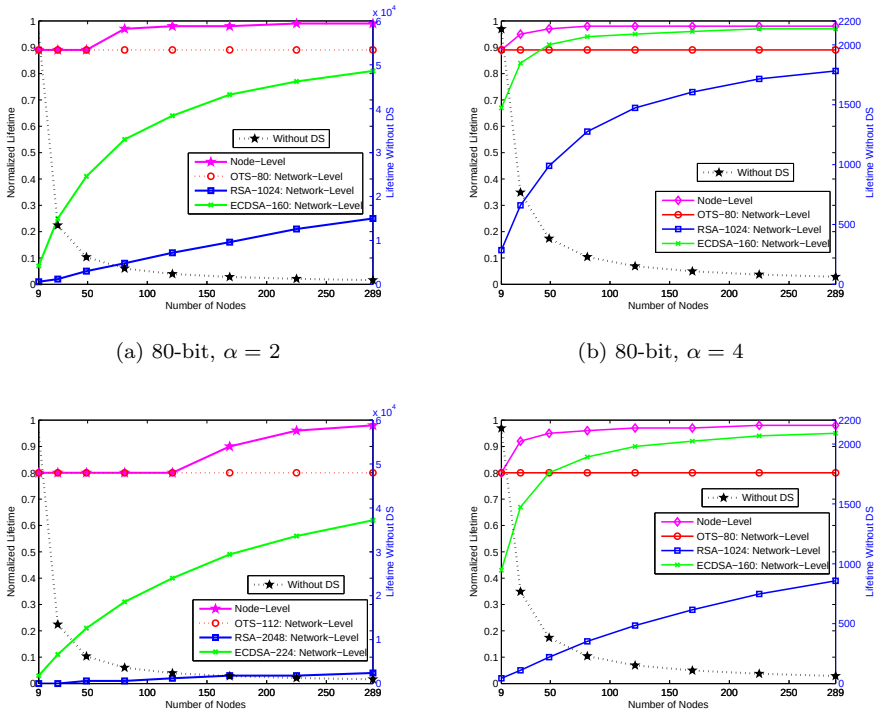


Figure 7: Lifetimes for different signing rates *(r in bits^{-1})* when $\alpha = 2$ (top) and $\alpha = 4$ (bottom) in the linear network topology for 80-bit security level.

constraint, each node can have at most 8 neighbors. For example, in Fig.2b, node 4 can reach to only nodes 1, 3, 5, 8, 9, 12, 19, and 20 provided that the inter-node separation is 10 m and the maximum transmission range is 15 m.

In Fig. 8, we present the lifetimes obtained with all strategies for $\alpha = 2$ and $\alpha = 4$ as well as for 80-bit and 112-bit security levels with 15 m transmission range limit. The limited transmission range assumption does not change the general trends we observed in unlimited transmission case (Fig. 4). Node level strategy performs better than network-level strategy with OTS which in turn performs better than the network-level strategy with ECDSA. RSA leads to the worst performance in terms of lifetime achieved. In fact, node-level strategy even fares better with the limited transmission range for $\alpha = 2$ when compared to the case of unlimited transmission range (*e.g.*, for $N = 289$, lifetimes achieved by node-level strategy with limited transmission range is just 2 % lower than lifetime without DS whereas with unlimited transmission range assumption lifetimes with node-level strategy are approximately 10 % lower). Indeed, with the limited transmission range the lifetime of the baseline case (no DS strategy) reduces when compared to the unlimited transmission range assumption due to the reduction in the number of available links to balance the energy dissipation throughout the network. Some nodes cannot dissipate their energies completely in a way that reduces the energy dissipations of more heavily burdened relay nodes. For example when we examine Fig. 1a, we see that a small fraction of data from node 3, 4, and 5 are sent directly to the base station to balance the energy dissipation although the energy costs of these flows are much higher than the transmission to closer neighbors. On the other hand, the node-level strategy could use the remaining energy for computation to reduce the transmission overhead. For example, in Fig. 8a, the node-level strategy uses not only OTS but also ECDSA, however, in the unlimited transmission case only OTS is used hence the transmission overhead is larger (Fig. 4a).



(a) 80-bit, $\alpha = 2$

(b) 80-bit, $\alpha = 4$

(c) 112-bit, $\alpha = 2$

(d) 112-bit, $\alpha = 4$

Figure 8: Normalized lifetimes for $\alpha = 2$ and $\alpha = 4$ in the grid network topology for 80-bit and 112-bit security levels with 15 m transmission range limit.

5. Heuristic Methods

In a wide variety of application domains (*e.g.*, telecommunications, logistics, and transportation problems [46]), optimization problems can be formulated as MIP models. However, solving MIPs (especially large instances) takes formidably high amount of computing due to the combinatorial nature of the binary/integer variables used within these models. Well known MIP solution techniques including Branch & Bound or Cutting Planes require excessive computing times even for small scale problems (networks) [13, 14]. This encouraged the researchers to develop heuristic algorithms which provide approximately optimal feasible solutions in a way faster than finding the exact optimal solutions to the problems. We are also interested in developing efficient heuristic algorithms to mitigate the computational burdens of the exact solution methods. In subsection 5.1 and subsection 5.2, we present Golden Section Search (GSS) and Simulated Annealing (SA) heuristics, respectively.

5.1. Golden Section Search (GSS) Heuristic

From Lemma 1, we know that RSA algorithm is never used in the optimal solutions. On the other hand, through the experimental study we made in Section 4, we observed that in most of the cases depending on the distances of the nodes to the base, either OTS or ECDSA are used: Closer nodes use OTS, whereas distant ones use ECDSA. As a consequence, the cutting edge between the OTS and ECDSA algorithms must be determined which can be done by implementing one of the line search algorithms such as the Golden Section Search *(GSS)* or Bisection Search. In this study, we decided to use the former one which has a better average case performance.

In our heuristic, we will search for the number of nodes to use the OTS (ECDSA) algorithm, which must take an integer value, over a finite set of nodes. Then, if the objective function value can be evaluated in polynomial time, then the resulting GSS method will also be polynomial. Note that, if we know which node uses which DS algorithm (values for a_k^i), then the linearized MIP model

derived in Section 3 reduces to an ordinary LP by removing the equations (9) to (14) and re-substituting w_k^i with $t \times a_k^i$. Since LPs can be solved in polynomial time, we conclude that the resulting algorithm is also polynomial.

GSS method determines the extremum by successively altering the range of values inside which the extremum is known to exist [47, 48]. The pseudo-code for this algorithm is presented in Algorithm 1. Since this heuristic is a mathematical programming based algorithm, it is implemented in GAMS IDE, which includes its own programming capabilities [42]. CPLEX 12.4 is used as the LP and MIP solver within GAMS.

Input: $k \leftarrow$ Number of Nodes;

Output: Lifetime, OTS and ECDSA nodes

Initialize the lower bound ($n \leftarrow 0$), the upper bound ($m \leftarrow k-1$), and the golden ratio $\big(\phi \leftarrow (-1+\sqrt{5})/2\big)$;

Compute $\lambda_1 \leftarrow \lceil m - \phi(m-n) \rceil$ and $\lambda_2 \leftarrow \lfloor n + \phi(m-n) \rfloor$;

while $|m-n| \geq 1$ **do**

  Set the first $\lambda_1 + 1$ nodes to OTS and the remaining ones to ECDSA.

  Compute the corresponding lifetime. Let this lifetime be β_1;

  Set the first $\lambda_2 + 1$ nodes to OTS and the remaining ones to ECDSA.

  Compute the corresponding lifetime. Let this lifetime be β_2;

  if $\beta_1 < \beta_2$ **then**

    Narrow the interval from the right by updating

    $m \leftarrow \lambda_2; \quad \lambda_2 \leftarrow \lambda_1; \quad \lambda_1 \leftarrow \lceil m - \phi(m-n) \rceil;$

  else

    Narrow the interval from the left by updating

    $n \leftarrow \lambda_1; \quad \lambda_1 \leftarrow \lambda_2; \quad \lambda_2 \leftarrow \lfloor n + \phi(m-n) \rfloor;$

  end

end

Result: $OTS \leftarrow 2 \leq i \leq n$;

Result: $ECDSA \leftarrow (n+1) \leq i \leq k$;

Algorithm 1: Pseudo-Code for the Golden Section Search

The positive number k is used to indicate the number of nodes in the WSN. n and m are used as boundary variables which contain the extremum. Initially,

we set $n = 0$ which indicates that no sensors use OTS, and $m = k - 1$ indicates that all sensors use ECDSA. The key parameter is the golden ratio denoted by ϕ and calculated accordingly [47, 48].

GSS method requires to determine two intermediate points λ_1 and λ_2 ($n < \lambda_1 < \lambda_2 < m$). They indicate the number of sensors that will use OTS algorithm in the corresponding solution. To be more explicit, consider $\lambda_1 = 5$. Then, from the nearest node to the base station (node 2) up to $2 + 5 = 7$th node (*i.e.*, node 2, 3, 4, 5, and 6), sensors are forced to use OTS. From $\lambda_1 + 1$st node (or $\lambda_2 + 1$st node) to the farthest node from the base station (node k), remaining sensors are forced to use ECDSA, respectively.

The resulting problems can be solved by using the LP formulation as already mentioned. The objective function values that are evaluated with λ_1 and λ_2 are saved as β_1 and β_2, respectively. Seeking for the maximum lifetime, when $\beta_1 < \beta_2$, GSS narrows the interval from $[n, m]$ to $[n, m = \lambda_2]$. Otherwise, if $\beta_1 \geq \beta_2$, GSS narrows the interval from $[n, m]$ to $[n = \lambda_1, m]$. Ultimately, new λ_1, λ_2, n, and m values are calculated for the new bounds and this process continues until the interval containing the extremum is less than 1, $m = n$. At the end of the algorithm n (or m) gives the optimum threshold for the sensors which use OTS to obtain the maximum network lifetime.

5.2. Simulated Annealing (SA) Heuristic

Note that, the assertion that the closer nodes use OTS and the farther nodes use ECDSA in the optimal solution is not always valid since counter examples can be found. Due to this reason and to test the performance of the GSS algorithm against a similar heuristic, we also develop a Simulated Annealing (SA) algorithm. In this algorithm, as in the GSS we search for different assignments of OTS and ECDSA algorithms to the nodes. However, we do not presume that the closer nodes use OTS and farther ones use ECDSA. Consider a vector with $k - 1$ elements (representing the sensor nodes in the network other than the base) having the value 1 or 2. If the value is 1, then this means the corresponding sensor uses OTS and if it is 2 it uses ECDSA.

Once we know this vector, then we can determine the optimal lifetime and flows corresponding to this vector by solving an LP. The SA is used to search for the optimal assignment of these algorithms (OTS or ECDSA) to nodes. The pseudo-code for this algorithm is presented in Algorithm 2.

Input: $k \leftarrow$ Number of Nodes, $q \leftarrow$ Initial Temparature, $\alpha \leftarrow$ Cooling Parameter, and $\beta \leftarrow$ Iteration Count Limit);

Output: Lifetime, OTS and ECDSA nodes

Initialize the iteration count ($IterCount \leftarrow 0$), best lifetime ($BestTime \leftarrow 0$);

Find an initial solution and set it as $CurrentSolution$. Calculate its lifetime by solving the LP and set as $CurrentTime$. Also set these as $BestSolution$ and $BestTime$, respectively;

while $IterCount \leq \beta$ **do**

Set the temperature $q \leftarrow q \times \alpha$. Find a $CandidateSolution$ as a neighbor of the $CurrentSolution$ by complementing one element of the vector.

Calculate its lifetime as $CandidateTime$;

if $CandidateTime > CurrentTime$ **then**

$CurrentTime \leftarrow CandidateTime$,

$CurrentSolution \leftarrow CandidateSolution$

if $CandidateTime > BestTime$ **then**

$BestTime \leftarrow CandidateTime$,

$BestSolution \leftarrow CandidateSolution$, $IterCount \leftarrow 0$.

else

$IterCount \leftarrow IterCount + 1$

$\Delta Obj \leftarrow CurrentTime - BestTime$. Accept $CandidateSolution$ as new $CurrentSolution$ with probability $e^{(-\Delta Obj/q)}$;

end

end

Algorithm 2: Pseudo-Code for the Simulated Annealing

Simulated Annealing heuristic starts with an initial solution. We found this solution by assigning OTS to half of the nodes (closer ones) and ECDSA to the remaining. At each iteration, a neighbor solution is determined and evaluated. This solution is found by randomly selecting a node and complementing its DS algorithm. That is, if the current one is OTS it is changed to ECDSA and

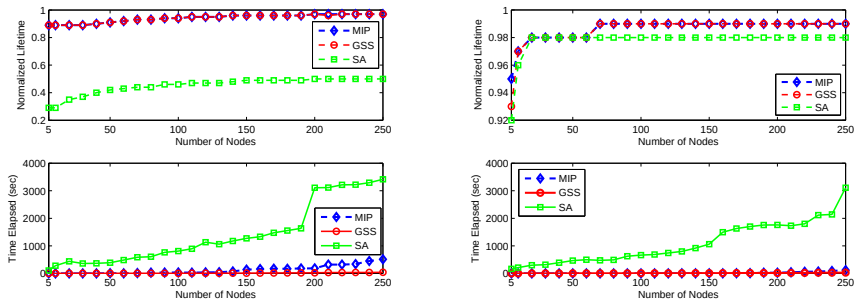
vice versa. Then this neighbor solution is evaluated. If it has a better objective function then it is accepted as the new current solution. Otherwise, it is accepted with a probability calculated using the current temperature and the difference in the lifetime values of the current and the candidate solutions. This probability is higher at the initial iterations of the search in order to satisfy diversification. As the temperature is cooled down at each iteration the probability of accepting a non-improving move is reduced.

In order to set the parameters of the Simulated Annealing we performed a computational study. Using the result of the computational study, we determined the initial temperature (q) as the lifetime value of the initial solution, cooling parameter (α) as 0.95, and iteration count limit (β) as 100.

5.3. Performance Evaluation

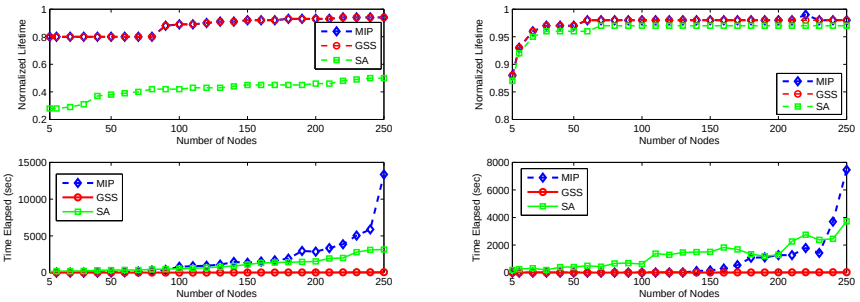
In this subsection, we investigate the performance of the heuristic method in terms of the time elapsed to obtain a solution and the deviation from the optimal value. For this purpose, we generate different problems using different parameter combinations. Each problem is solved by the MIP model and the heuristic algorithms and the results are compared with each other. In the top panels of Figs. 9 to 11, normalized lifetime values given by the exact solution of the MIP model and the heuristic methods are plotted as functions of number of nodes (up to 250 nodes for linear and 289 nodes for two dimensional topologies, respectively). In the bottom panels computation times are plotted.

Solving the MIP model for larger topologies with more than 300 nodes requires prohibitively excessive computational time, thus, we approximated the solution time for larger networks by using Cubic Spline Extrapolation (CSE) [49]. In Tables 5, 6, and 7, we present the approximate solution times driven by CSE for the MIP model, and the solution times obtained by the heuristics for linear, deterministic grid, and random square topologies, respectively. We use four relatively larger networks (*i.e.*, the larger networks comprise of 350,



(a) 80-bit, $\alpha = 2$

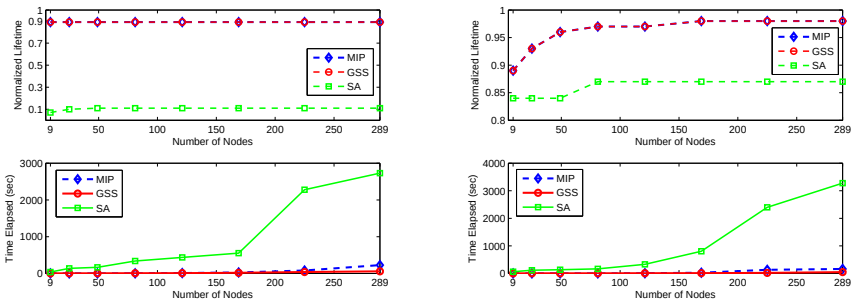
(b) 80-bit, $\alpha = 4$



(c) 112-bit, $\alpha = 2$

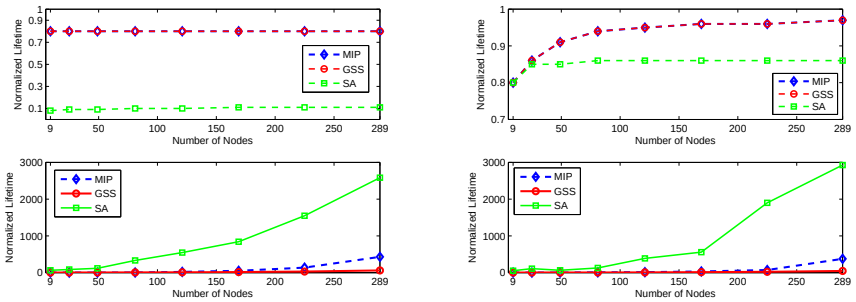
(d) 112-bit, $\alpha = 4$

Figure 9: Normalized lifetime and performance comparison of MIP vs. heuristics when $\alpha = 2$ and 4 in the linear network topology for 80-bit and 112-bit security level.



(a) 80-bit, $\alpha = 2$

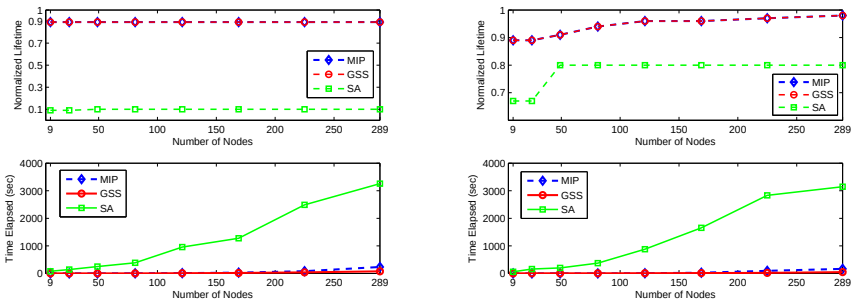
(b) 80-bit, $\alpha = 4$



(c) 112-bit, $\alpha = 2$

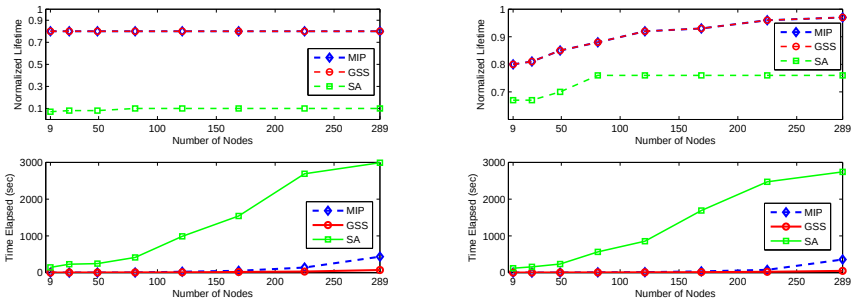
(d) 112-bit, $\alpha = 4$

Figure 10: Normalized lifetime and performance comparison of MIP vs. heuristics when $\alpha = 2$ and 4 in the grid network topology for 80-bit and 112-bit security level.



(a) 80-bit, $\alpha = 2$

(b) 80-bit, $\alpha = 4$



(c) 112-bit, $\alpha = 2$

(d) 112-bit, $\alpha = 4$

Figure 11: Normalized lifetime and performance comparison of MIP vs. heuristics when $\alpha = 2$ and 4 in the random network topology for 80-bit and 112-bit security level.

Table 5: Performance Evaluations of the MIP and heuristics in terms of computation times for linear topology

α	NN	80-bit SL			112-bit SL		
		MIP (s)	GSS (s)	SA (s)	MIP (s)	GSS (s)	SA (s)
2	750	4.13×10^6	1761	2.90×10^5	2.13×10^8	1602	1.57×10^5
	1000	13.61×10^6	2975	3.46×10^5	6.96×10^8	2160	1.78×10^5
4	350	1.68×10^3	99	1.52×10^3	0.48×10^6	178	4.98×10^3
	500	1.20×10^4	306	6.88×10^3	7.49×10^6	473	6.79×10^3

Table 6: Performance Evaluations of the MIP and heuristics Method in terms of computation times for grid topology

α	NN	80-bit SL			112-bit SL		
		MIP (s)	GSS (s)	SA (s)	MIP (s)	GSS (s)	SA (s)
2	729	80.84×10^3	1342	5.05×10^4	20.45×10^3	1682	1.95×10^4
	1089	32.02×10^4	2707	1.30×10^5	85.16×10^3	7476	7.24×10^5
4	729	26.57×10^3	1041	2.25×10^4	38.58×10^3	1466	3.05×10^4
	1089	13.25×10^4	3106	5.36×10^4	17.39×10^4	4234	1.27×10^5

500, 750, and 1000 nodes[4]) to address the performance evaluations on linear topology whilst we opt two different networks (*i.e.*, networks comprising of 729 and 1000 nodes) designated as large grid topologies. We report the performance evaluations for both 80-bit and 112-bit security levels (SL) with two different propagation environments ($\alpha = \{2, 4\}$). Eventually, all solution times obtained by CSE and heuristics are presented in seconds.

Performance comparisons in linear networks with 80-bit security is presented in Figs. 9a and 9b for $\alpha = 2$ and $\alpha = 4$, respectively. Normalized lifetimes obtained with the MIP model and GSS have a difference less than 0.99% (for

[4]In these tables NN denotes the number of nodes in the network.

Table 7: Performance Evaluations of the MIP and heuristics in terms of computation times for random topology

α	NN	80-bit SL			112-bit SL		
		MIP (s)	GSS (s)	SA (s)	MIP (s)	GSS (s)	SA (s)
2	729	92.05×10^3	1534	85.24×10^3	22.67×10^3	1832	19.57×10^3
	1089	41.02×10^4	2875	28.45×10^4	88.45×10^3	7654	69.21×10^3
4	729	28.13×10^3	1423	16.44×10^3	41.67×10^3	1656	20.12×10^3
	1089	19.25×10^4	3346	17.25×10^4	21.45×10^4	4521	20.03×10^4

$\alpha = 2$) and 1.29% (for $\alpha = 4$). Computation time of GSS outperforms the MIP model in larger networks which comprise of at least 60 nodes for $\alpha = 2$ and 200 nodes for $\alpha = 4$.

In Fig. 9c and Fig. 9d we present the performance evaluations by using 112-bit security level in linear topology for $\alpha = 2$ and $\alpha = 4$, respectively. Normalized lifetimes obtained with MIP model and GSS have a difference less than 0.05% (for $\alpha = 2$) and 0.95% (for $\alpha = 4$). GSS outperforms the MIP model in terms of solution time for larger networks which comprise, at least, of 70 nodes (for $\alpha = 2$) and 140 nodes (for $\alpha = 4$).

In Figs. 10a and 10b, we present a comparison of the MIP model and the heuristic methods in grid topologies with 80-bit security level for $\alpha = 2$ and $\alpha = 4$, respectively. Solutions found by GSS matches the solutions found by the MIP model in both propagation environments. Computation time of GSS outperforms the MIP model for larger networks which have at least 169 nodes in total for both propagation environments.

Performance evaluations performed in a grid topology with 112-bit security level is presented in Fig. 10c and Fig. 10d for $\alpha = 2$ and $\alpha = 4$, respectively. Computation time of GSS outperforms the MIP model for larger networks which have at least 121 nodes in both propagation environments.

In Figs. 11a-11d, we present a comparison of the MIP model and the heuristic methods in random topologies for both 80-bit and 112-bit security levels for $\alpha =$

2 and $\alpha = 4$, respectively. As in the case of the deterministic two dimensional topologies, in random topologies the difference between the MIP and GSS are negligibly low while GSS solutions are performing better, especially for larger networks, in terms of computation times.

When we compare the two heuristic algorithms, GSS outperforms SA under all settings both in terms of achieved network lifetime and computation time. In fact, simplicity of SA is the reason for its unsatisfactory performance (*i.e.*, only three parameters are needed in SA). Furthermore, the iteration rules of the SA algorithm is very much dependent on randomness, yet, GSS is based on deterministic iteration rules. In all topologies and scenarios we investigated, deterministic iteration rules of the GSS lead to better solutions in lower solution times when compared to those of SA.

Although, GSS gives network lifetimes within 2% neighborhood of the MIP solutions, it is desirable to develop a polynomial time algorithm that can find the exact optimal solution. However, existence of such an algorithm for our problem is not known even for deterministic regular topologies (*i.e.*, line and grid topologies). Therefore, the investigation of the possibility of creating an exact polynomial time solution procedure for our problem is left as an open research question.

We constructed several special network topologies and scenarios where the basic rule of GSS (*i.e.*, the closer nodes use OTS and the farther nodes use ECDSA in the optimal solution) does not hold. However, such topologies are pathological and their occurrence probabilities are very low (*e.g.*, we have not encountered any such topology in random networks). Even in such topologies GSS outperforms SA.

6. Simulations

Up to this point, we investigated the optimal solutions based on LP or MIP models and presented a heuristic algorithm which approximates the optimal solutions. In this section, we complete our analysis by proposing two greedy

algorithms which maximize the network lifetime both with the network-level and the node-level strategies. We perform simulations using the greedy algorithms and compare the simulation results against the optimal solutions attained by mathematical programming models.

For the network-level strategies, we propose Updated Shortest Cost Path (USCP) algorithm (see Algorithm 3) which is a modified version of Flow Augmentation (FA) algorithm proposed in [44]. USCP algorithm calculates the lowest cost paths to the base station from each node which are then used to transport data to the base station. The important part of the algorithm is the selection of an appropriate Cost Function (CF) and its implementation. We note that if the CF is calculated only for once (*i.e.*, the lowest cost path is determined for each node i for the entire lifetime of WSN) then all data generated by each node i is transported to the base station on the same path and energy of the nodes on that path drains out more quickly than the others. Therefore, USCP algorithm calculates shortest cost paths periodically at each round using the following formula:

$$CF_{ij} = \frac{E_{tx,ij}}{e_i^r} + \gamma \frac{E_{rx}}{e_j^r} \tag{17}$$

CF_{ij} represents the cost of a link between node i and node j by considering transmission energy cost ($E_{tx,ij}$), reception energy cost (E_{rx}), energy remaining at batteries (e_i^r), path lost exponent (α), and the distance of node i to the base station in terms of average number of hops (ANH) in the network as in Equation(17).

In Equation(17), γ is a scaling factor and it is determined according to α and ANH. As the ANH increases, γ also increases. For example, when $\alpha = 2$, (ANH,γ) values are (1,0), (2,0), (3,1), (4,2), (5,3), (6,4), (7,5), and (8,6) for grid topologies with 9, 25, 49, 81, 121, 169, 225, and 289 nodes, respectively. As α increases the cost of sending data on each link increases, sharply. More precisely, when $\alpha = 4$, γ values are 0, 4^2, 5^2, 6^2, 7^2, 8^2, 9^2, and 10^2 for 9, 25, 49, 81, 121, 169, 225, and 289 nodes, respectively.

For the node-level strategy, we propose extended USCP (USCP-ex) algo-

1 Lifetime = 0

2 Calculate the cost function for each node i using

$$CF_{ij} = \frac{E_{tx,ij}}{e_i^r} + \gamma \frac{E_{rx}}{e_j^r}$$

if *residual energy in battery of each node i is enough,* **then**

3 | Calculate lowest cost paths for this round using a shortest path algorithm (Dijkstra [50], BellmanFord [51])

4 **end**

5 **if** *any node i can't find a path to the base station,* **then**

6 | Return (Lifetime);

7 | Stop;

8 **else**

9 | Continue;

10 **end**

11 Send generated data (s_i) and signature overhead data ($S_{c,i}$) through the base station using calculated lowest cost paths.

12 Calculate the dissipated energies for each node i using

$$E_{rx} \sum_{\substack{j \in W \\ j \neq i}} f_{ji} + \sum_{\substack{j \in V \\ j \neq i}} f_{ij} \; E_{tx,ij} + E_{c,i}$$

13 Update residual energies of nodes in their batteries.

14 Lifetime = Lifetime + 1

15 Go to 2.

Algorithm 3: USCP Algorithm

rithm (see Algorithm 4). In the USPC-ex algorithm the nodes not only determine the path to send their data but they also determine the DS algorithm to be used throughout the network lifetime. Due to the high energy cost of RSA, in the node-level strategy, nodes select either OTS or ECDSA as the DS algorithm.

For the selection of DS algorithm, we consider the distance of node i to the base station (d_{i-BS}) and α. When α is small, energy cost of sending data becomes smaller. Therefore, the energy cost of computation for signing data becomes more important than the transmission energy cost pertaining to the signature size. On the other hand, when α is large, energy cost of signing data becomes less important than the signature size. Hence, when α=2, nodes tend to select OTS and when α=4, nodes tend to select ECDSA. Nevertheless, the proper selection also depends on the distance between the signing node and the base station. A threshold value (DS_{th}) is parameterized for the selection of the DS algorithm. If we assume that all sensor nodes generate data at a constant rate of s_i = 3600 bits/hour = 1 bit/second, then the lifetime value in seconds is numerically equal to the number of bits generated per node.

In Fig. 12, Fig. 13, and Fig. 14, we present the simulation results obtained by using the USPC and USCP-ex algorithms in linear, grid, and random network topologies, respectively. The parameters and the topologies used to obtain the results in Fig. 12, Fig. 13, and Fig. 14 are the same with those in Subsection 4.2, Subsection 4.3, and Subsection 4.4, respectively. The only difference is that the results in Fig. 3, Fig. 4, and Fig. 5 are obtained by centralized solutions of LP and MIP models and the results in Fig. 12, Fig. 13, and Fig. 14 are obtained by simulations of the greedy algorithms. By comparing these two sets of results, we see that the general trends in the simulation results are consistent with the trends we observed in optimal results. The decreases in network lifetime values are as expected in Fig. 12, Fig. 13, and Fig. 14 when compared to the lifetime values presented in Fig. 3, Fig. 4, and Fig. 5. Indeed, USPC and USPC-ex give lower network lifetime values than the network lifetimes obtained by the optimal cases. However, the extent of the lifetime decrease for the linear, grid,

1 Lifetime = 0

2 Calculate the cost function for each node i using

$$CF_{ij} = \frac{E_{tx,ij}}{e_i^r} + \gamma \frac{E_{rx}}{e_j^r}.$$

3 **if** *residual energy in battery of each node i is enough,* **then**

4 | Calculate lowest cost paths for this round using a shortest path algorithm (Dijkstra [50], BellmanFord [51])

5 **end**

6 **if** *any node i can't find a path to the base station,* **then**

7 | Return (Lifetime);

8 | Stop;

9 **else**

10 | Continue;

11 **end**

12 **for** *each node i* **do**

13 | **if** $d_{i-BS} > DS_{th}$ **then**

14 | | use ECDSA;

15 | **else**

16 | | use OTS;

17 | **end**

18 **end**

19 Send generated data (s_i) and signature overhead data ($S_{c,i}$) through the base station using calculated lowest cost paths.

20 Calculate the dissipated energies for each node i using

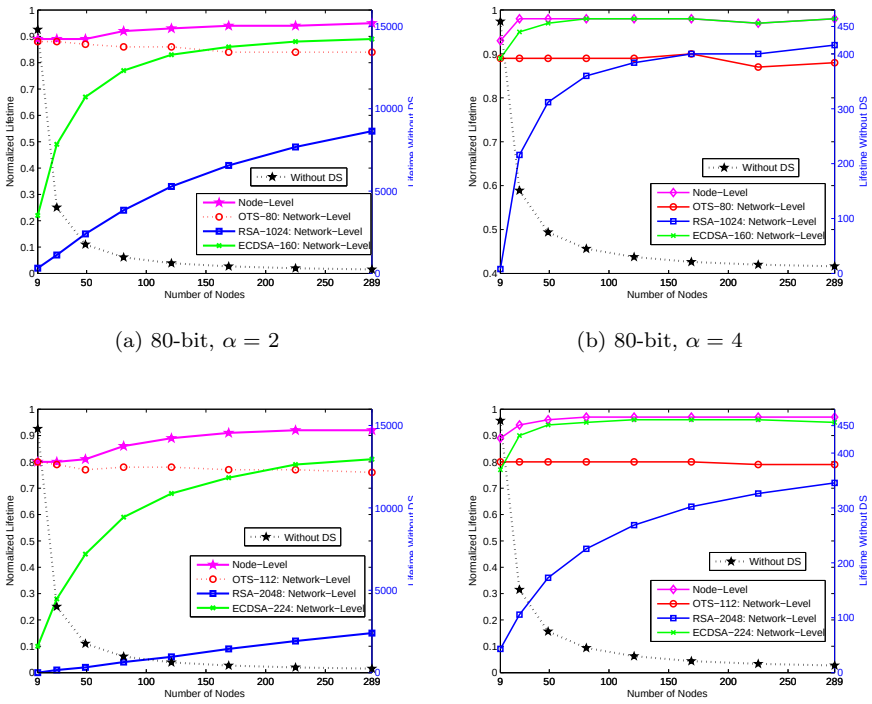
$$E_{rx} \sum_{\substack{j \in W \\ j \neq i}} f_{ji} + \sum_{\substack{j \in V \\ j \neq i}} f_{ij} \; E_{tx,ij} + E_{c,i}$$

21 Update residual energies of nodes in their batteries.

22 Lifetime = Lifetime + 1

23 Go to 2.

Algorithm 4: USCP-ex Algorithm



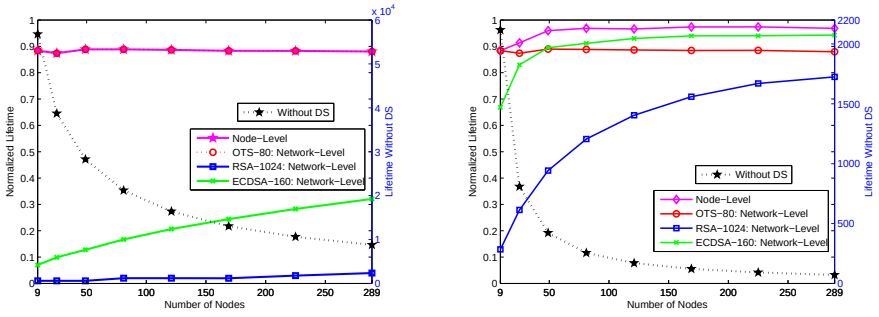
(a) 80-bit, $\alpha = 2$

(b) 80-bit, $\alpha = 4$

(c) 112-bit, $\alpha = 2$

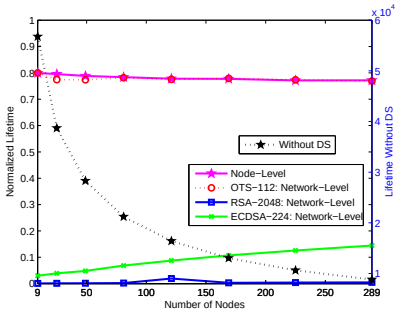
(d) 112-bit, $\alpha = 4$

Figure 12: Normalized lifetimes obtained by USCP and USPC-ex algorithms for $\alpha = 2$ and $\alpha = 4$ in the linear network topology (d_{int}=10m) for 80-bit and 112-bit security levels.

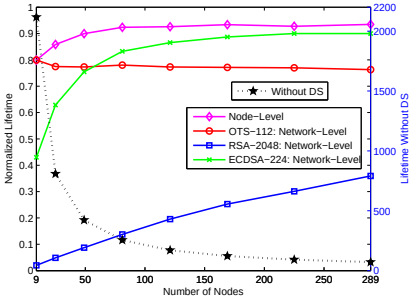


(a) 80-bit, $\alpha = 2$

(b) 80-bit, $\alpha = 4$

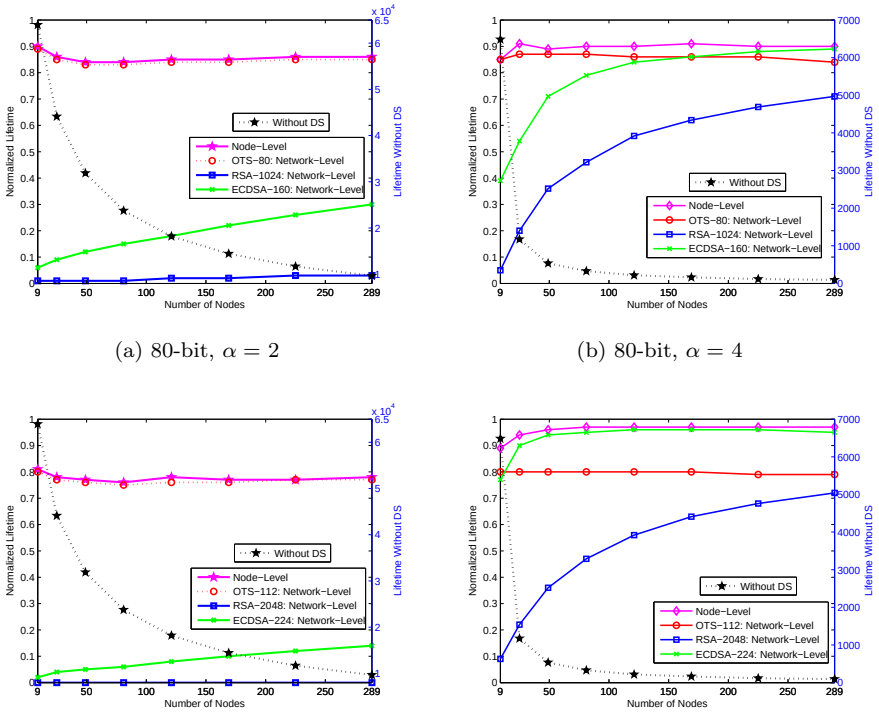


(c) 112-bit, $\alpha = 2$



(d) 112-bit, $\alpha = 4$

Figure 13: Normalized lifetimes obtained by USCP and USPC-ex algorithms for $\alpha = 2$ and $\alpha = 4$ in the grid network topology for 80-bit and 112-bit security levels.



(a) 80-bit, $\alpha = 2$

(b) 80-bit, $\alpha = 4$

(c) 112-bit, $\alpha = 2$

(d) 112-bit, $\alpha = 4$

Figure 14: Normalized lifetimes obtained by USCP and USPC-ex algorithms for $\alpha = 2$ and $\alpha = 4$ in the random network topology for 80-bit and 112-bit security levels.

and random topologies are limited by 5.96% (*i.e.*, the lifetime obtained by USPC algorithm for OTS-80 in a 289 node network with $\alpha = 2$ is 5.96% lower than the optimal network lifetime), 4.75% (*i.e.*, the lifetime obtained by USPC algorithm for ECDSA-224 in a 25 node network with $\alpha = 4$ is 4.75% lower than the optimal network lifetime), and 9.98% (*i.e.*, the lifetime obtained by USPC-ex algorithm for node level strategy in a 289 node network with $\alpha = 4$ is 9.98% lower than the optimal network lifetime), respectively. Furthermore, the majority of network lifetime values for the linear, grid, and random topologies are within 1.75%, 2.50%, 3.25% neighborhood of the corresponding optimal values, respectively.

7. Conclusions

In typical WSN applications, there are two main categories of energy dissipation: communication energy dissipation and computation energy dissipation. Limited battery power of sensor nodes necessitates optimal balancing of the computation and communication energy dissipations to prolong the network lifetime. The trade-offs involved in such optimization problems can be investigated by the techniques of mathematical programming. We revisit the network lifetime optimization problem in WSNs exemplified by a non-repudiation service through digital signature algorithms using network-level strategies introduced in [4]. Our key finding is the identification of a more fine grained node-level strategy that can outperform the network-level strategies. For this purpose, we build a novel MIP framework to optimize network lifetime by using the node-level strategy. We present two polynomial time heuristic algorithms to reduce the computational complexity of the MIP model in large networks. We investigate the performance gains brought by the node-level strategy and the heuristic method by exploring the parameter space. Finally, we perform simulations using the proposed greedy algorithms and compare the simulation results against the optimal solutions. Since the motivation for this paper is posed as a sequel of questions in Section 1, we present our main conclusions in reply to these questions itemized as follows:

1. The node-level strategy, which provides flexibility for nodes to choose the optimum DS algorithm, gives higher network lifetimes when compared to the network-level strategies where all sensors strictly use a single algorithm (up to 22.50% lifetime increase). In general, the configuration in which the nodes closer to the base station use OTS and the nodes farther away from the base station use ECDSA is the optimal for maximizing the network lifetime.
2. We first build a nonlinear programming model to formulate the node-level strategy. Later, we convert the nonlinear programming model to an MIP model without making any simplifying assumptions. The MIP model is solvable in reasonable time provided that the number of nodes in the network is not more than a few hundred.
3. The MIP model cannot be solved in large networks consisting of more than a few hundred nodes, furthermore, the solution time increases drastically as a function of network size. Therefore, we develop a heuristic algorithm (*i.e.*, GSS) that surpasses the MIP model in terms of solution time, especially in large networks. Computational study results show that, GSS algorithm can solve large problems that cannot be solvable by the MIP model in reasonable times and the maximum deviation of the heuristic solution from the optimal solution is less than 1.3%.

References

[1] S. Gandham, M. Dawande, R. Prakash, S. Venkatesan, Energy efficient schemes for wireless sensor networks with multiple mobile base stations, in: Proc. IEEE Global Telecommunications Conference (GLOBECOM), Vol. 1, 2003, pp. 377–381.

[2] C. Zhu, L. Shu, T. Hara, L. Wang, S. Nishio, L. T. Yang, A survey on communication and data management issues in mobile sensor networks, Wireless Communications and Mobile Computing 14 (1) (2014) 19–36.

[3] Z. Cheng, M. Perillo, W. Heinzelman, General network lifetime and cost models for evaluating sensor network deployment strategies, IEEE Transactions on Mobile Computing 7 (2008) 484–497.

[4] K. Bicakci, I. E. Bagci, B. Tavli, Communication/computation tradeoffs for prolonging network lifetime in wireless sensor networks: The case of digital signatures, Information Sciences 188 (2012) 44–63.

[5] S. Ergen, P. Varaiya, On multi-hop routing for energy efficiency, IEEE Communications Letters 9 (2005) 880–881.

[6] Q. Dong, Maximizing system lifetime in wireless sensor networks, in: Proc. International Symposium on Information Processing in Sensor Networks (IPSN), 2005, pp. 13–19.

[7] B. Tavli, K. Bicakci, R. Zilan, J. M. Barcelo-Ordinas, A survey of visual sensor network platforms, Multimedia Tools and Applications 60 (3) (2012) 689–726.

[8] K. Piotrowski, P. Langendoerfer, S. Peter, How public key cryptography influences wireless sensor node lifetime, in: Proc. ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN), 2006, pp. 169–176.

[9] A. Wander, N. Gura, H. Eberle, V. Gupta, S. Shantz, Energy analysis of public-key cryptography for wireless sensor networks, in: Proc. IEEE International Conference on Pervasive Computing and Communications (PerCom), 2005, pp. 324–328.

[10] R. L. Rivest, A. Shamir, L. Adleman, A method for obtaining digital signatures and public-key cryptosystems, Communications of the ACM 21 (1978) 120–126.

[11] D. Hankerson, A. J. Menezes, S. Vanstone, Guide to Elliptic Curve Cryptography, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2003.

[12] L. Lamport, Constructing digital signatures from a one-way function, Tech. rep., SRI International Computer Science Laboratory (1979).

[13] M. R. Garey, D. S. Johnson, Computers and Intractability; A Guide to the Theory of NP-Completeness, W. H. Freeman & Co., New York, NY, USA, 1990.

[14] C. H. Papadimitriou, K. Steiglitz, Combinatorial optimization: algorithms and complexity, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1982.

[15] R. Bixby, E. Rothberg, Progress in computational mixed integer programminga look back from the other side of the tipping point, Annals of Operations Research 149 (2007) 37–41.

[16] E. A. Silver, E. A. Silver, An overview of heuristic solution methods, in: Proc. International Conference on Industrial Engineering Theory, Applications and Practice, Vol. 55, 2004, pp. 936–956.

[17] S. Seys, B. Preneel, Power consumption evaluation of efficient digital signature schemes for low power devices, in: Proc. IEEE International Conference on Wireless And Mobile Computing, Networking And Communications (WiMob), Vol. 1, 2005, pp. 79–86.

[18] Y. B. Saied, A. Olivereau, D. Zeghlache, M. Laurent, Lightweight collaborative key establishment scheme for the internet of things, Computer Networks 64 (0) (2014) 273 – 295.

[19] P. Szczechowiak, A. Kargl, M. Scott, M. Collier, On the application of pairing based cryptography to wireless sensor networks, in: Proc. ACM Conference on Wireless Network Security (WiSec), 2009, pp. 1–12.

[20] Y. W. Law, Z. Gong, T. Luo, S. Marusic, M. Palaniswami, Comparative study of multicast authentication schemes with application to wide-area measurement system, in: Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security, ASIA CCS '13, ACM, New York, NY, USA, 2013, pp. 287–298.

[21] L. Wolsey, Integer Programming, Wiley Interscience Publication, 1998.

[22] F. Ishmanov, A. S. Malik, S. M. Kim, Energy consumption balancing (ECB) issues and mechanisms in wireless sensor networks (WSNs): a comprehensive overview, European Transactions on Telecommunications 22 (2011) 151–167.

[23] A. Gogu, D. Nace, A. Dilo, N. Meratnia, Review of optimization problems in wireless sensor networks, in: J. Hamilton Ortiz (Ed.), Telecommunications Networks - Current Status and Future Trends, InTech, 2012, pp. 153–180.

[24] A. Alfieri, A. Bianco, P. Brandimarte, C.-F. Chiasserini, Maximizing system lifetime in wireless sensor networks, European Journal of Operational Research 181 (2007) 390–402.

[25] K. Bicakci, H. Gultekin, B. Tavli, The impact of one-time energy costs on network lifetime in wireless sensor networks, IEEE Communications Letters 13 (2009) 905–907.

[26] B. Tavli, M. Kayaalp, O. Ceylan, I. Bagci, Data processing and communication strategies for lifetime optimization in wireless sensor networks, AEU - International Journal of Electronics and Communications 64 (2010) 992–998.

[27] B. Tavli, I. Bagci, O. Ceylan, Optimal data compression and forwarding in wireless sensor networks, IEEE Communications Letters 14 (2010) 408–410.

[28] A. C. Santos, F. Bendali, J. Mailfert, C. Duhamel, K. M. Hou, Heuristics for designing energy-efficient wireless sensor network topologies, Journal of Networks 4 (2009) 436–444.

[29] A. Hoang, M. Motani, Collaborative broadcasting and compression in cluster-based wireless sensor networks, in: European Wireless Sensor Networks Workshop (EWSN), 2005, pp. 197–206.

[30] B. Fateh, M. Govindarasu, Joint scheduling of tasks and messages for energy minimization in interference-aware real-time sensor networks, Mobile Computing, IEEE Transactions on 14 (1) (2015) 86–98.

[31] Y. Gu, M. Pan, W. Li, Joint sleep scheduling and routing for lifetime optimization in delay-sensitive sensor networks, in: Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2013 10th Annual IEEE Communications Society Conference on, 2013, pp. 263–263.

[32] A. A. El-Sherif, A. Mohamed, V. C. Leung, Optimum power and rate allocation in video sensor networks, in: Global Communications Conference (GLOBECOM), 2013 IEEE, 2013, pp. 480–486.

[33] M. Cheng, Q. Ye, L. Cai, Cross-layer schemes for reducing delay in multihop wireless networks, Wireless Communications, IEEE Transactions on 12 (2) (2013) 928–937.

[34] H. U. Yildiz, K. Bicakci, B. Tavli, Communication/computation tradeoffs in wireless sensor networks: Comparing network-level and node-level strategies, in: Wireless Sensors and Sensor Networks (WiSNet), 2014 IEEE Topical Conference on, IEEE, 2014, pp. 49–51.

[35] W. Heinzelman, A. Chandrakasan, H. Balakrishnan, An application-specific protocol architecture for wireless microsensor networks, IEEE Transactions on Wireless Communications 1 (2002) 660–670.

[36] C.-F. Chiasserini, E. Magli, Energy consumption and image quality in wireless video-surveillance networks, in: Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Vol. 5, 2002, pp. 2357–2361.

[37] W. Heinzelman, A. Chandrakasan, H. Balakrishnan, An application specific protocol architecture for wireless microsensor networks, IEEE Transactions on Wireless Communications 1 (2002) 660–670.

[38] K. Bicakci, H. Gultekin, B. Tavli, The impact of one-time energy costs on network lifetime in wireless sensor networks, IEEE Communications Letters 13 (2009) 905–907.

[39] B. Tavli, W. Heinzelman, Energy and spatial reuse efficient network wide real-time data broadcasting in mobile ad hoc networking, IEEE Transactions on Mobile Computing 10 (2006) 1297–1312.

[40] T. Kothmayr, W. Hu, C. Schmitt, M. Bruenig, G. Carle, Poster: Securing the internet of things with dtls, in: Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems, ACM, 2011, pp. 345–346.

[41] NIST report on cryptographic key length and crypto-period.
URL `http://www.keylength.com/en/4/`

[42] General Algebraic Modeling System (GAMS).
URL `http://www.gams.com`

[43] M. Bhardwaj, A. Chandrakasan, Bounding the lifetime of sensor networks via optimal role assignments, in: Proc. Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM), Vol. 3, 2002, pp. 1587–1596.

[44] J.-H. Chang, L. Tassiulas, Maximum lifetime routing in wireless sensor networks, IEEE/ACM Transactions on Networking 12 (2004) 609–619.

[45] S. Jiang, Y. Xue, Optimal wireless network restoration under jamming attack, in: Proc. Internatonal Conference on Computer Communications and Networks (ICCCN), 2009, pp. 1–6.

[46] I. Gamvros, B. Golden, S. Raghavan, D. Stanojevi, Heuristic search for network design, in: H. G (Ed.), Tutorials on Emerging Methodologies and Applications in Operations Research, Vol. 76 of International Series in Operations Research & Management Science, Springer New York, 2005, pp. 1–46.

[47] J. Kiefer, Sequential minimax search for a maximum, Proceedings of the American Mathematical Society 4 (1953) 502–506.

[48] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, Numerical Recipes 3rd Edition: The Art of Scientific Computing, Cambridge University Press, 2007.

[49] J. Stoer, R. Bulirsch, Introduction to Numerical Analysis, 3rd Edition, Springer, New York, 2002.

[50] E. W. Dijkstra, A note on two problems in connexion with graphs, Numerische Mathematik 1 (1959) 269–271.

[51] R. Bellman, On a routing problem, Quarterly of Applied Mathematics (1959) 87–90.