Energy-Aware Activity Control for Wireless Sensing Infrastructure Using Periodic Communication and Mixed-Integer Programming

Piotr Arabas 1,* , Andrzej Sikora 1 and Wojciech Szynkiewicz 2

Abstract: This paper addresses the energy conservation problem in wireless sensor networks, in which sensor data are aggregated into packages and then transmitted periodically to the base station according to an established schedule. We formulate two mixed integer programming problems, first for minimum total energy usage and second for min-max per node energy usage. We present two algorithms for slot allocation that exploit the periodic nature of the data collection process by allocating a subset of nodes to subsequent frames. A mixed-integer solver solves the resulting mathematical programming task. The performed numerical experiments show that the proposed approach is appropriate for relatively small networks when minimizing the total energy consumption problem. Hopefully, the problem may be solved for medium size networks when a maximum node energy consumption performance index is used.

Keywords: wireless sensor network; energy; mixed integer programming

1. Introduction

The primary application of wireless sensor networks (WSNs) is collecting measurement data from the area of operation of the network. Typically, the data is transmitted multi-hop to a specified node, a so-called gateway, which sends the data further to a repository, e.g., in the cloud. Frequent transmissions, unfortunately, consume device power, which determines the lifetime of the wireless sensor network. One solution to this problem is the use of periodic sensor networks (PSNs), in which measurement data are aggregated into packets and then transmitted periodically to the gateway according to an established schedule. Networks of this type can be built from devices having a memory buffer, in which the collected data can be stored until the scheduled transmission. Examples of PSN network applications are presented in the paper [1].

One of the crucial challenges in PSNs is to provide energy-efficient routing to collect data from selected devices within an assumed time window. In the literature, approaches using network clustering or data preprocessing and aggregation are most commonly proposed [2–4]. These algorithms must consider the limited energy resources of the nodes, the limited data buffers and the radio transmission range [2]. In [3] the authors proposed a two-level data aggregation technique to minimize the amount of data transmitted, thereby saving power consumption and bandwidth reduction. Aslam et al. [4] presented an energy-saving clustering algorithm based on a multi-criteria optimization method that enables using three different metrics for single-hop clustering. A similar problem may be solved by optimizing sensor location if an application allows this—see for example [5]. Additionally to routing, it is possible to adjust power settings of the nodes to suit processing...
and transmission needs which may be done by solving mathematical programming tasks resembling these used in static, backbone networks [6,7].

We assumed that the location and energy profile of the sensors were known and that measurement data would be collected periodically. Thus, it is possible to formulate a mixed-integer programming problem (MIP) whose solution is the transmission schedule of the network nodes. In addition, the formulation considers the hidden node and exposed node problem to reduce interference and increase the quality and efficiency of network transmission.

A significant problem is the computational complexity of the task. However, its solution can be obtained centrally before physically running the network and collecting data on an external machine or computing environment with adequate resources. The obtained solution can then be deployed on the network and reused many times.

We analyze the infrastructure to monitor environmental data in some areas, e.g., farmland or dangerous waste dumps (see Figure 1 left). The sensors are located over the monitored site in a relatively dense, static network to facilitate the detection of anomalies like arid soil or change of toxic substance concentration. The sensors’ locations are known and do not change during network operation. The monitored processes are typically slow, varying so that the sensors may report their measurements with relatively low frequency. Furthermore, taking advantage of dense network topology, it is possible to sample only part of the sensors periodically and use collected data to interpolate measurements for the whole area. Such operation allows for the conservation of network energy and provides more prolonged operation.

Another application using periodic sampling of only part of the network to minimize communication and power consumption may be registering events for off-line analysis, e.g., monitoring animals moving around their territory (see Figure 1 right). The sensors activated by an animal may store data in their memory and transmit them in bulk at predefined time instants. It is important to note that energy consumption may be lowered by aggregating data in a single transmission and avoiding contention thanks to optimizing the transmission schedule.

We propose a periodic communication framework consisting of the number of frames divided into several time slots to coordinate sensor communication. In each frame, J sensors are allowed to transmit data. As the number of sensors sending data is N − 1 (sensor 0 is base station), it is necessary to have (N − 1) / J frames provided N − 1 is fully divisible by J. In such a framework, it is possible to precompute sensors to slot allocation in each frame and use it to configure sensors before deployment. If sensor software allows on-the-fly configuration, it is also possible to change allocations during network operation if necessary to accommodate battery discharge rates or sensor malfunction. Such a scheme is beneficial in two ways. First, it allows avoiding contention even if relatively short frames are used. The contention may be a fundamental problem in dense networks as close spacing increase the risk of interference. Short frame, in turn, makes synchronizing slots easier. The second benefit is the possibility to replace a single but formidable allocation task with a larger number of much simpler problems. Lower numbers of active nodes and slots to allocate make this task easier to solve. Taking these into account, the proposed framework may be considered a kind of heuristic, where nodes are allocated to frames using a simple algorithm—e.g., by random picking, and then allocations within the frames are found. As the resulting tasks are simpler to solve, we envisage that, at least for smaller networks, it is possible to use exact methods, i.e., MIP, and obtain high-quality solutions. Such a method was proposed in [8], where the authors presented a novel, highly accurate linear transformation for the MILP approach, aiming at optimal deployment of the switches in a distribution system to enhance its reliability.
The main contributions of this paper are: A novel mathematical formulation for determining energy-efficient routing and transmission scheduling in a PSN that takes into account:

- energy profiles of radio devices;
- performance characteristics of the PSN;
- effects of radio transmission in multi-hop networks.

We have introduced two formulations of two mixed integer programming problems and discuss numerical results to illustrate the effectiveness of the proposed optimization approaches. The solution of these problems for sample data is then presented along with conclusions and further research directions.

The rest of the paper is organized as follows. In Section 2 we present related work on energy-saving methods used in WSNs. Section 3 describes network and energy models. Section 4 provides the formulation of two mixed integer programming problems. In Section 5 we present and discuss numerical results to illustrate the effectiveness of the proposed optimization approaches. Finally, Section 6 concludes the paper and highlights future research directions.

2. Related Work

There is a vast amount of literature relating to energy conservation to improve the lifetime of WSNs, as a plethora of methods and techniques has been proposed in the last few decades. As a result, there exist many surveys on energy-saving techniques in WSNs [9–12]. However, energy efficiency analysis is complicated because the network lifespan depends on many aspects, including network infrastructure and technology, computational load, routing protocols, channel characteristics and energy consumption model. Sensor nodes consume energy during sensing, processing and communication. Many approaches proposed in the literature assume that the energy consumed by the sensor in the communication module is much higher than the energy consumed for data sampling or processing [11]. Therefore, energy-saving in WSNs is focused mainly on optimizing the operation of the radio module. In [9], the authors presented a taxonomy of energy-saving strategies that mainly concentrated on work cycling and data-reduction techniques. Aziz et al. [10] reviewed approaches that strive for prolonging the network lifetime by Topology Control (TC). They proposed a novel definition of Topology Control and categorized the relevant techniques into different classes. In a more recent survey [12], the authors provided a survey of the existing energy-saving approaches in the conventional static WSNs that are not fitted with any energy harvesting or movable modules. They also proposed a general definition of energy efficiency for stationary WSNs.

The WSN topology is defined by the devices deployed in the environment and the links available between them. Links arise from the communication ranges of installed radio modules. By taking various actions, one can influence the network infrastructure and generate network topologies with desired characteristics (e.g., lifespan, robustness, coverage, connectivity) while increasing network capacity and reducing power consumption. Furthermore, energy-aware routing protocols can be em-
ployed to reduce transmission time and energy. To summarize, the following techniques can facilitate energy saving in WSNs: (1) transmission power control [13–15]—adjustment of the transmit power of the radio module to maintain a transmission range; (2) clustering and activity control [4,10,16,17]—clustering of nodes into clusters and temporal activation and deactivation of sensory nodes; (3) energy-aware routing [18,19]—calculating energy-aware routing tables, multi-path routing sink mobility; (4) energy-efficient data reduction [2,3,20]—aggregation and routing of data over a multi-hop network, processing data in intermediate nodes to reduce redundant data before transmission, adaptive sampling.

The approach proposed in this paper belongs to Activity Control (AC) algorithms. WSNs typically consist of a large number of sensor nodes distributed in the environment. Sensor data is typically transmitted over some time, either according to a schedule or in response to a new event. Therefore, in most scenarios, communication modules can remain inactive most of the time and are only activated to transmit measurements or receive messages from other sensor nodes. This fact is particularly advantageous since transmission is expensive and the power consumption of the communication module depends on the current operating mode. Activity control or sleep/wakeup schemes utilize dynamic control of radio modules of all nodes and adjust the operating mode according to the current situation in the network to reduce power consumption and thus increase the network lifetime. Therefore, implementing an appropriate strategy for transitioning to different states is critical for effective idle power control. It is a complex problem to formulate an optimal strategy for switching off a radio module. Duty cycling schemes are typically classified into three categories: on-demand, asynchronous and scheduled rendezvous [9].

Activity control protocols for WSNs use dynamic management of communication modules of all nodes in the network. They put the radio receiver to sleep mode at specified intervals [21]. The goal is to reduce energy consumption while minimizing the negative effect on network capacity and data routing efficiency. These protocols should be able to buffer traffic destined for sleeping nodes and forward data in a partial network defined by a coverage set. The membership of the coverage set must be rotated among all nodes to maximize network lifetime. Various types of protocols of this type are described in the literature.

Many AC algorithms have been proposed in the literature [9,21–23]. The majority of them combine two main techniques—clustering and algorithms for periodic radio module sleep [17,21,24]. Clustering-based algorithms manage the network load and balance the energy usage of the sensor nodes. In [21], a new sleep scheduling technique was proposed based on the offset scheduling to achieve low latency transmission in a large scale WSN employed in mission-critical applications. Cheng et al. [23] proposed an energy-aware centralized strategy for node scheduling problem for the service-oriented WSNs. They also proposed a distributed solution to this problem. In this approach, one header is assigned to each service, and the scheduling algorithm is performed in a distributed manner. The authors of [24] introduced multi-channel TDMA scheduling algorithms to minimize the total energy consumption in the WSN. The presented algorithms use multiple radio channels for efficient scheduling while eliminating collisions and eavesdropping. In [17], Rawat et al. presented a classification of the different clustering protocols based on their performance, network organization, and operation techniques. One of the most popular clustering routing protocols of WSN is LEACH (Low-Energy Adaptive Clustering Hierarchy) [25,26]—a self-organizing, adaptive protocol. LEACH uses a randomized rotation of cluster heads to evenly distribute the energy load among all nodes in the network. It also performs local data compression in each cluster to reduce overall communication. LEACH has two main disadvantages. One of them is the random selection of cluster heads. The distribution of cluster heads is random, which makes clustering unreliable [26]. Another is the significant energy consumption for large WSNs. Several clustering routing
protocols are based on the energy efficiency to select cluster heads, such as VH-LEACH [27], LEACH-T [28] and TB-LEACH [29] have been proposed.

Energy conservation is an important issue not only in WSNs but also in electric vehicles. For example, in [30] the authors presented a framework for establishing a strategic auction of wireless charging aggregators that serve wireless charging stations to provide charging services to electric vehicles on the move.

3. Periodic Wireless Sensor Network Model

In this section, we introduce the network model, energy profile of a network node and network topology model. The notation used in the following sections is summarized in Table 1.

| Table 1. Summary of Notation. |
|--------------------------------|
| \( \mathcal{V} \) | set of nodes (vertices) in the network |
| \( v_i \) | \( i \)-th node in the network \( v_i \in \mathcal{V} \) |
| \( F \) | time frame |
| \( Q_i \) | buffer capacity of node \( v_i \) |
| \( q_{it}^i \) | number of data packets in the buffer of node \( v_i \) at time \( t \in F \) |
| \( e_{kt}^i \) | \( e_{kt}^i \in \mathbb{B} \) equals 1 if the node \( v_i \) at time \( t \in F \) is in the energy state \( k \in \{1, \ldots, K\} \), 0 otherwise |
| \( \xi^i \) | current energy level in the node \( v_i \) |
| \( A^i_{jk} \) | \( A^i_{jk} \in \mathbb{B} \), equals 1 when the transmission from the node \( v_j \in S_D \), which is in the energy state \( k \in S_D \), otherwise equals 0 |
| \( B^i_{jk} \) | \( B^i_{jk} \in \mathbb{B} \) equals 1 if the signal received at the node \( v_i \) cannot be decoded correctly, 0 if no interference |
| \( \phi_k^i \) | energy consumption level in node \( v_i \) |
| \( \gamma \) | SINR threshold |
| \( S_k^i \) | energy model of node \( v_i \) |
| \( J_{total} \) | objective function in minimization of total energy consumption |
| \( J_{max} \) | objective function in minimization of the maximal energy consumed by a node |
| \( \mathbb{B} \) | set of binary numbers \([0,1]\) |

We consider a wireless ad-hoc network consisting of a set of \( N \) sensors \( v_i \) \( (i \in \{1, \ldots, N\}) \) monitoring a selected environmental parameter at a uniform, constant sensor data reading rate and one base station (gateway) \( v_0 (i = 0) \) whose purpose is to collect measurement data.

The running time of the sensory network was divided into a series of cyclically repeated time frames \( F = \{0,1,\ldots,T\}, F', F'', \ldots \)—each of length \( T \) of consecutive time slots \( t \). The initial state \( t = 0 \) for the next frame \( F' \) is the final state \( t = T \) of the previous frame \( F \). For simplicity, the collected and transmitted data are divided into equally sized packets. Each of these data packets can be transmitted over the network between two directly neighboring nodes \( v_i \) and \( v_j \) during one unit of time \( t = F \). Each node in the network has a data buffer that can store up to \( Q_i \) packets of data at a time. The base station’s data buffer size is assumed to be unlimited \( Q_0 = \infty \).

Each time unit \( t \in F \) was assigned a variable \( q_{it}^i \) denoting the number of data packets in the buffer of node \( v_i \) at time \( t \in F \). It was also assumed that at the initial time \( t = 0 \), at least one packet of monitored data coming from the sensor \( (q_{it}^i > 0) \) enters the buffer of each node \( v_i \).

3.1. Energy Profile of a Network Node

The second variable associated with time \( t \) is the binary variable \( e_{kt}^i = 0,1 \) taking the value 1 when the node \( v_i \) at time \( t \in F \) is in the energy state \( k \in \{1, \ldots, K\} \). Otherwise, the variable takes the value 0.
It is assumed that the energy resource available to the device forming the network at any given moment is limited to the charge level of the battery supplying a node of the network.

It is also assumed that each node in the network can operate in one of \( k = 1, 2, \ldots, K \) energy states \((K \geq 3)\) associated with different energy consumption and transmitted signal power levels. The first index \( k = 1 \) is reserved for the sleep state, in which the radio module is not involved in communication—it is turned off. The index \( k = 2 \) denotes the listening state, during which a network node \( v_i \) is prepared to receive data from another node \( v_j \). The remaining energy states \((2 < k \leq K)\) are used during data transmission with different RF power levels assigned to different energy states. As the transmission signal’s power level increases, the range of wireless transmission increases, which may cause changes in the topology of the considered ad-hoc network. Each of the energy states of the node \( v_i \) has been assigned an energy consumption level \( \phi_i^k \) generated by the radio communication module and a power level of the radio signal \( P_{t_i}^k \) transmitted by the device in the state \( k \).

For the sleep state and the listening state, the power level of the transmitted signal is equal to 0 W. It is additionally assumed that as the state index \( k \) increases, the power level of the transmitted signal increases: \( P_{t_2}^l < \ldots < P_{t_K}^l \). With the transmission of a signal, there is a corresponding energy consumption \( \phi_i^k \) measured in units of [mA]. Using the introduced designations, the energy model \( S_{i,E} \) of the device \( v_i \) can be formulated:

\[
S_{i,E} = \{\xi_i, e_i^k : k = 1, \ldots, K_i\},
\]

(1)

where the binary variable \( e_i^k = \{0, 1\} \) takes the value 1 when the node \( v_i \) at time \( t \) is in the energy state \( k \in \{1, \ldots, K\} \). Otherwise, the variable takes the value 0.

\[
e_i^k = \begin{cases} 
1 & \text{when node } v_i \text{ is in energy state } k \\
0 & \text{when node } v_i \text{ is not in energy state } k 
\end{cases}
\]

(2)

A node \( v_i \) can only operate in one energy state at any given time \( t \). Therefore, the following constraint has been introduced:

\[
\forall v_i \in S_D \sum_{k=1}^{K} e_i^k = 1.
\]

(3)

The energy consumption of device modules other than the radio module is not considered in this paper. This approach can only be used when a significant amount of energy is consumed for wireless communication and exceeds the total energy used by the network node for other functions (e.g., monitoring, computation, traffic). In all cases, the analysis of the energy states of the radio module is used to determine the ad-hoc network topology. The Figure 2 presents the energy state parameters of the CC2420 [31] radio communication module, which is often used in IEEE 802.15.4 compliant devices. It is worth noting that the device in the sleep state \((k = 1)\) also consumes energy, while the listen state \((k = 2)\) generates the highest energy demand.
3.2. Network Topology Model

It is assumed that a wireless network node $v_i \in S_D$ can correctly receive data from another wireless network node $v_j \in S_D$ when it is in listening mode ($e^2_i = 1$) and the power level $Pr^i$ of the signal received from node $v_j$ is greater than an assumed threshold value—receiver sensitivity—$Ps^j$ and greater by a value $\gamma$ than the noise and interference power $Pn^i$ ([32]), where $\gamma$ corresponds to the minimum value of the coefficient $SNIR_{\min}$ assumed by the designer. These conditions are defined by parameter $A^i_{jk}$, which takes value 1 when the transmission from the node $v_j \in S_D$, which is in the energy state $k \in S_D$, to the node $v_i \in S_D$ operating in the listening mode is possible. Otherwise, the parameter takes a value equal to 0. The situation where too low a power level of the signal received by node $v_i$ does not allow the signal to be decoded correctly and at the same time increases the level of interference when receiving other signals is described by the parameter $B^i_{jk} = 1$.

$$Pm^i = \max\{Ps^j, Pn^i + \gamma\}$$  \hspace{1cm} (4)

$$A^i_{jk} = \begin{cases} 1 & Pr^j \geq Pm^i : k \in \{3, \ldots, K\}, \\ 0 & \text{wireless communication is not possible}, \end{cases}$$ \hspace{1cm} (5)

$$B^i_{jk} = \begin{cases} 1 & kTB < Pr^j < Pm^i : k \in \{3, \ldots, K\} \\ 0 & \text{no interference}, \end{cases}$$ \hspace{1cm} (6)

4. Mathematical Programming Problem Formulation

This section presents two MIP formulations, first for minimum total energy usage and second for min-max per node energy usage. In the first problem, the objective function aims to reduce the energy used by the network as a whole. The second problem is formulated as a minimization of energy consumed by the most energy-consuming sensor node. The notation used in these formulations is summarized in Table 1.

4.1. Network Definition—Constraints

Constraint (7) describes sensor node buffer evolution, i.e., accumulation when receiving messages ($l = 2$) and reduction when in sending mode ($k = 3, \ldots, K$). The inequality (8) models buffer capacity. Let us assume that node 0 (base station) has a buffer sufficient to collect all data sent by other nodes. Equation (9) restricts node operation to single-mode—sleeping, receiving or transmitting—in every moment $t$. Inequalities (10) and (11) describe media access, namely hidden and exposed node problem [33,34] respectively. Constraint (10) ensures that every receiving node $i$ must be able to hear the signal from at most one transmitting node $j$. To complement this, the constraint (11) ensures that at least one node $i$ must be able to receive the transmission of node $j$. Equation (12) guarantees that all data present in the initial time instant
\( t = 0 \) must be transmitted to node 0, while the attendant Equation (13) imposes empty buffers in all nodes except base station at last time slot \( t = T \).

\[
\forall i = 0, \ldots, N, \quad l = 2, \\
q^i_l = q^i_{l-1} + e^i_{lT} - \sum_{k=3}^{K} e^i_{kT},
\]

\[\forall i = 0, \ldots, N, \quad l = 2, \quad 0 \leq q^i_l \leq Q_i,\] (8)

\[
\forall i = 0, \ldots, N, \quad l = 1, \ldots, T, \\
\sum_{k=1}^{K} e^i_{kT} = 1,\] (9)

\[
\forall i = 0, \ldots, N, \quad l = 1, \ldots, T, \quad k = 3, \ldots, K, \\
e^i_{kT} + N(1 - e^i_{lT}) \geq \sum_{j=1}^{N} (A^i_{jk} + B^i_{jk}) e^j_{kT},
\]

\[\forall i = 1, \ldots, N, \quad l = 1, \ldots, T, \quad k = 3, \ldots, K, \\
e^i_{kT} \leq \sum_{j=0}^{N} A^i_{jk} e^j_{lT},\] (10)

\[
\sum_{i=1}^{N} q^0_i = q^0_T - q^0_0,
\]

\[
\sum_{i=1}^{N} q^T_i = 0.
\] (13)

### 4.2. Minimization of Total Energy Consumption

Constraints (7)–(13) define operating principles of the network. It is necessary to describe the goal to be attained first to find the optimal slot allocation. One may consider minimization of the overall power consumption of the network. Such a goal will promote conserving the power consumption of the network as a whole, regardless of the power consumption of a particular node. In dense topologies where some nodes may replace exhausted ones, it may lead to longer network operation. Formally it can be done by defining the performance index (see Equation (14)) in the form of the sum of energy consumed by all nodes in subsequent time slots of the frame:

\[ J_{\text{total}} = \sum_{i=1}^{T} \sum_{k=1}^{K} \phi^i_k e^i_{kT}. \] (14)

The resulting optimization problem may be defined by formula (15):

\[
\min_{q^i_l, e^i_{kT}} J_{\text{total}},
\] (15)

subject to constraints (7)–(13).

### 4.3. Minimization of the Maximal Power Consumed by a Node

For sparse networks, it may be reasonable to redefine the energy-saving problem as minimizing the energy consumed by the busiest node. Such an approach aims to prevent premature exhaustion of important node energy source, rendering the network inoperable. Formally it may be done by minimizing of energy consumed by the most energy-consuming node. Introducing an additional variable \( z \) one can avoid min-max problem formulation by adding the following constraint:

\[
\forall i = 1, \ldots, N, \quad \sum_{l=1}^{T} \sum_{k=1}^{K} \phi^i_k e^i_{kT} \leq z,\] (16)
Continuous variable $z$ is not less than the energy consumed by the most energy-consuming node within the whole frame. Using this variable, it is possible to define the performance index by Equation (17):

$$J_{\text{max}} = z.$$  \hspace{1cm} (17)

The resulting optimization problem may be defined then by Equation (18):

$$\min_{q_i, e_{ik}, z} J_{\text{max}},$$  \hspace{1cm} (18)

subject to constraints (7)–(13) and (16).

5. Experiments

Numerical experiments verified the proposed approaches. Their aim was:

1. To find out practical network size for which the task may be effectively solved using state of the art mixed-integer (MIP) solver;
2. To find out appropriate algorithm parameters, especially number of nodes sending data in a frame;
3. To compare the effectiveness of both proposed algorithms;
4. To analyze solution features and propose how they can be exploited to speed up computations.

The experiments involved solving mathematical programming tasks generated for a set of scenarios described in the following subsections.

5.1. Scenarios

Application of proposed algorithms to periodic sensor networks involves computation of a set of frames coordinating node activity, allowing all nodes to send their data to the base node. An appropriate scenario consisting of a selected number of frames was generated by randomly drawing indexes of nodes having data to send to provide realistic results for each network. There was an equal number of data sending nodes in every frame when the number of nodes was fully divisible by the number of slots. Otherwise, it was accepted that the last scenario served fewer nodes—see Figure 3 for an example of such a scenario.

It must be noted that the base station does not send data, so the number of data sending nodes is $N - 1$. When an equal number of data sending nodes was required, the additional nodes were randomly duplicated. Square grid topology was adopted to make tests synthetic and easy to interpret. The node can operate in one of five states: sleeping, receiving and transmitting with low, medium or high power settings. The corresponding ranges and power consumption are presented in Table 2.

| Energy State | Transmission Range [m] | Power [mW] |
|--------------|------------------------|------------|
| Number       | Description            | Effective  | Interference |          |
| 1            | sleeping               | –          | –            | 0.004     |
| 2            | receiving              | –          | –            | 120       |
| 3            | transmitting—low       | 65         | 85           | 85        |
| 4            | transmitting—medium    | 95         | 115          | 98        |
| 5            | transmitting—high      | 125        | 145          | 112       |

The radio characteristics of the nodes are based on Texas Instruments CC2420 [31] chip used commonly in sensor nodes, e.g., in TelosB family. The ranges shown in Table 2 are used to infer model parameters $A_{ik}^j$ and $B_{ik}^j$, while power levels are used as values of...
parameter $\phi_k$ to calculate the performance index. The grid edge length was set at 50 m, so starting from a $3 \times 3$ grid, at least some of the nodes require multi-hop communication to reach the base node.

The experiments were carried out using scripts written in Perl language driving input data preparation and output data collection for CPLEX solver (version 20.1.0.0). The mathematical programming model and data were encoded using AMPL language. The machine used for computations was equipped with an Intel Xeon E3-1240 v.3 processor with eight cores operating at 3.4GHz and 8GB RAM.

![Figure 3](image)

**Figure 3.** Example scenario for the network of 16 nodes and four frames. Nodes having data to send are marked in red, base station is in a larger black one. Note that every node except the base station can send its data in one of the frames.

### 5.2. Influence of Number of Nodes Having Data to Send Within the Frame on Algorithm Performance

The number of nodes having data to send in a time frame and their placement in the network topology determines the path length and ability of nodes to be activated concurrently. Therefore, scenarios consisting of less data sending nodes may result in significantly simpler mathematical programming tasks and lower computation time. On the other hand, as all nodes must have the ability to communicate with the base station, activating a too-small number of nodes implies more data collection rounds and slows down the process. To assess the appropriate sender number, a set of experiments was carried out where its number was equal to $\sqrt{N}$ in every time frame. As all nodes except node 0 (base station) must be served, allocations for $\sqrt{N}$ of frames were calculated by selecting randomly without returns $\sqrt{N}$ nodes having data to send. A lacking node was randomly duplicated to provide an equal number of sending nodes in every frame. The frame of eight slots was used in all experiments.

The data presented in Tables 3 and 4 show results aggregated for $\sqrt{N}$ calculated frames. For the algorithm using total power consumption as the performance index, the dependence of computation time on sending nodes number is visible even if a problem size (see Table 5) is considered. The resulting problems were so demanding that the calculation of one of the frames exhausted all computer memory and had to be terminated. It must be noted that a problem with more nodes having data to send is harder to solve as not only more paths must be calculated, but what is supposed to be more critical, the paths must be coordinated to prevent interference. The last condition is challenging as all paths must end in a single base node, and the number of slots in the frame is limited. The computation times for the algorithm using the maximal power consumption index are much more favorable. However, still, there is a strong influence, visible especially in maximal computation time values. For a relatively broad number of scenarios, the sending nodes number for further experiments was set to 4 to compare both algorithms.
Table 3. Results for algorithm using total energy consumption index, (*) optimization of one of the frames for 25 nodes scenario was terminated after 180 min as computer memory was nearly exhausted, the maximum time and performance index values are related to this frame.

| Number of Nodes | Computation Time [s] | Performance Index | Frames Number |
|-----------------|----------------------|-------------------|---------------|
|                 | Total                | Having Data       | Average       | Min. | Max. | Average | Min. | Max. | Number |
| 9               | 3                    |                   | 0.07          | 0.024| 0.16 | 996.3   | 874.3| 1118.2| 3      |
| 16              | 4                    |                   | 0.1           | 13.3 | 1334.5| 1079.5  | 1583.5| 4     |
| 25              | 5                    |                   | 3813          | 2.9  | 10,800*| 1688.1  | 1541.7| 2238.7| 5      |

Table 4. Results for algorithm using maximal energy consumption index.

| Number of Nodes | Computation Time [s] | Performance Index | Frames Number |
|-----------------|----------------------|-------------------|---------------|
|                 | Total                | Having Data       | Average       | Min. | Max. | Average | Min. | Max. | Number |
| 9               | 3                    |                   | 0.09          | 0.047| 0.137| 93.7    | 85   | 98   | 3      |
| 16              | 4                    |                   | 7.8           | 2    | 18   | 112     | 112  | 112  | 4      |
| 25              | 5                    |                   | 18.4          | 1.38 | 35.7 | 112     | 112  | 112  | 5      |

Table 5. Problem sizes for algorithm using total energy consumption index, for algorithm using maximal energy consumption index number of variables and the number of constraints must be incremented by 1. Note that problem size does not depend on the number of nodes having data to send.

| Number of Nodes | Number of Variables | Number of Constraints |
|-----------------|---------------------|-----------------------|
| 9               | 432                 | 452                   |
| 16              | 768                 | 802                   |
| 25              | 1200                | 1252                  |
| 36              | 1728                | 1802                  |
| 49              | 2352                | 2452                  |

5.3. Algorithms Comparison

The next set of experiments was aimed at comparing the performance of both algorithms for networks of varying sizes. For every network, the entire scenario was computed, i.e., the problems were solved for all frames providing that all nodes except the base station were able to send their data. As the number of nodes sending data within a frame was set to 4, the number of frames was \(\lceil (N - 1)/4 \rceil\), and in the case when \(N - 1\) was not fully divisible by 4, the last frame consisted of less sending nodes.

It is important to note that the number of variables and constraints grows significantly with network size—see Table 5. In total energy formulation, all variables are binary or integer. The maximum node energy version uses a single additional continuous variable. Even using highly efficient solvers like CPLEX problems, over a thousand variables may be considered hard, especially when a specific structure of bounds is present.

Results of this may be easily observed in Tables 6 and 7. The tables present average and extreme values of time and performance index for all frames forming subsequent scenarios. One of the most visible features is that computation times for algorithm using total energy consumption index (Table 6) tends to be, at least for larger networks, relatively high, making its application for networks with more than 16 nodes questionable. For algorithms using maximal energy consumption index the computation times seems to be growing
much slower. They may be considered acceptable when the slot allocation is repeated rarely, e.g., on a time scale of several hours or days.

Table 6. Results for algorithm using total energy consumption index.

| Number of Nodes | Computation Time [s] | Performance Index |
|-----------------|----------------------|-------------------|
|                 | Average   | Min. | Max. | Average | Min. | Max. |
| 9               | 0.062     | 0.024| 0.1  | 996.3   | 874.3| 1118.2|
| 16              | 67.2      | 0.064| 266  | 1276.5  | 655.5| 1774.5|
| 25              | 1476.9    | 0.26 | 8162 | 1485    | 597.7| 2264.7|

Table 7. Results for algorithm using maximal energy consumption index.

| Number of Nodes | Computation Time [s] | Performance Index |
|-----------------|----------------------|-------------------|
|                 | Average   | Min. | Max. | Average | Min. | Max. |
| 9               | 0.42      | 0.26 | 0.57 | 112     | 112  | 112  |
| 16              | 7.45      | 0.23 | 12.4 | 108.5   | 98   | 112  |
| 25              | 17        | 3.67 | 41.2 | 112     | 112  | 112  |
| 36              | 12.9      | 1.77 | 36.2 | 110.4   | 98   | 112  |
| 49              | 16.9      | 4.1  | 64   | 112     | 112  | 112  |

For a deeper analysis of the results, it may be helpful to compare a gap between initial approximations and optimal solutions presented in Tables 8 and 9 for algorithms using total energy performance index and maximal energy performance index, respectively. Data in the tables are values of the relaxed (continuous) solution and the initial integer upper bound found in the root node of the branch and bound procedure. It must be noted that the algorithm could not find an integer solution in the root node for some cases. The distance between two approximated solutions and optimal solution gives some idea of the general complexity of the task. Specifically, there are some favorable cases when the number of branch and bound iterations may be low or even zero, like nine nodes scenarios. In general, the algorithm using the total energy consumption performance index needs more branch and bound iterations than that using maximal energy consumption, and the number of iterations grows rapidly with the problem size. For the algorithm using a maximal energy consumption performance index, this growth is not so apparent. In this case, the influence of a gap between the initial upper bound on the number of branch and bound iterations is more visible. When the upper bound cannot be found in the root node—as in some cases marked with (*) and (**) in the Tables 8 and 9—the branch and bound procedure becomes more difficult as the possibility to skip some branches early is limited.

Table 8. Comparison of average relaxed solution and upper bounds in the root node for algorithm using total energy consumption index. The optimal solution is found after the number of branch and bound iterations presented in the last column. Value marked with (*) is the mean of 4 of 6 cases as CPLEX did not provide an initial approximation in the root node for all tasks.

| Number of Nodes | Average Solution | Average Branch and Bound Iterations |
|-----------------|------------------|-------------------------------------|
|                 | Root: Relaxed    | Root: Upper Bound                  | Optimal          |
| 9               | 914.8            | 996.3                              | 996.3            |
| 16              | 914              | 1441.2                             | 1276.5           |
| 25              | 994.2            | 1771.5 *                           | 1485             |

|                     |                  |                                     | 2,592,057.2      |

| 16              | 914              | 1441.2                             | 1276.5           |
| 25              | 994.2            | 1771.5 *                           | 1485             |
Table 9. Comparison of average relaxed solution and upper bounds in the root node for algorithm using maximal energy consumption index. The optimal solution is found after the number of branch and bound iterations presented in the last column. Value marked with (*) and (**) are means of 5 of 6 cases and 2 of 9 cases, respectively as CPLEX did not provide an initial approximation in the root node for all tasks.

| Number of Nodes | Average Solution | Average Branch |
|-----------------|------------------|----------------|
|                 | Root: Relaxed    | Root: Upper Bound | Optimal and Bound Iterations |
| 9               | 85.7             | 112             | 112                        | 2351.5           |
| 16              | 85               | 224.8           | 108.5                      | 32,149            |
| 25              | 85               | 160.25 *        | 112                        | 48,665.2          |
| 36              | 86               | 112 **          | 110.4                      | 26,929.3          |
| 49              | 85               | 112 **          | 112                        | 39,067.7          |

Another important observation is that computation time may vary significantly within a scenario. This suggests that coordinating data transmission in some frames may be more difficult than in others.

An example of such a situation may be seen in Table 6 for the 25 nodes scenario. The solution for one of the frames may be found in about a quarter of a second, while the other needs over 2 hours to be solved. Furthermore, the solution of the first case requires no branch and bound iterations, while the second needs nearly 400 million iterations. This clearly shows that for some favorable data locations integer solution may be found easily. The initial locations of data to be sent are depicted in Figure 4. In the case marked “1”, the data are located close to the base station, and most of them may be sent in one hop. On the contrary, in the case marked “2”, the nodes are located far from the base station and at such distance from each other that they can easily interfere. There are still several possible paths for the data. However, taken into account the radio range, they all require high power setting of the transmitter.

![Figure 4](image)

Comparison of solution quality for both algorithms is not straightforward as they have different goals and performance indices. When comparing values of indices in Tables 6 and 7, two differences may be noticed. First, the spread of minimal and maximal values tends to be greater for algorithm minimizing total energy consumption. Next, values of the total energy consumption index tend to grow with the size of the network. In contrast, for algorithm minimizing maximal energy consumption, the performance index values are nearly constant. To make comparison easier, average values of the second index for both algorithms were computed and presented in Figure 5. The algorithm using the total energy consumption index minimizes the energy consumed by all network nodes in all frame slots. As the performance is a sum, it may be significantly lower for simpler
(e.g., having 3 sending nodes instead of 4 or having nodes located close to the base station) frames.

Figure 5. Average values of total energy performance index (left graph) and maximum node energy performance index (right graph) for both algorithms.

On the contrary, the performance index of the second algorithm is equal to the power needed by the node transmitting the data with the highest power setting. As it is value for a single node, it does not grow with the size of the network, and there is also significantly less variability in it. Namely, it can take one of three values related to the three power levels of the sensor. The algorithm using maximal node energy index tries to maximize network life by using many nodes transmitting the data with low power settings. This way, the battery use is balanced among most sensors. Such a strategy typically, at least for small to medium networks, leads to solutions where the nodes transmit data no more than once—it may be clearly seen in Table 7 where the maximal value of performance index is 112 mW, which equals energy consumption for the highest transmission power. On the other hand, algorithm minimizing total energy allows multiple transmissions through the same node—this fact is reflected in the right graph of Figure 5.

The overall network energy consumption for algorithm minimizing total energy may be significantly lower. However, it has two drawbacks. First, the resulting mathematical programming task is much harder to solve. It is necessary to take into consideration all nodes to minimize the total energy. In the second algorithm, only the most energy-consuming node is essential while others can operate in any state, making it easier to find a feasible solution. Next, calculating it independently for several frames does not ensure that overall network life will be long enough. Adversely, minimizing maximal energy should typically provide longer network life thanks to the transmitting nodes being spread more evenly.

6. Conclusions

The paper presents two algorithms for slot allocation in a periodic sensor network. Both take advantage of the periodic nature of the data collection process by allocating a subset of nodes to subsequent frames. They can be considered a kind of centralised activity control mechanism coordinating data transmission in subsequent frames. The resulting schedule satisfies two main targets: it reduces power consumption and provides contention-free routing. The communication rate may be relatively low, thanks to data buffering in nodes. Precomputing schedule makes using additional signaling communication channels and synchronization unnecessary. It is important to note that the solution takes into consideration interference (i.e., hidden node problem). As a schedule allows all nodes to send their data, the scheme may be used when periodic data acquisition is necessary. The proposed scheme is restricted to static typologies. In varied environments, decentralized and reactive activity control mechanism provides better performance and reliability.
We conducted numerical experiments to assess the performance of the proposed algorithms. They show that the proposed method of the solution—with the help of branch and bound MIP solver—is appropriate for network size in a range of 50 nodes when a maximum node energy consumption performance index is used. Minimization of total energy consumption index results in a much more complicated structure of constraints, so the solution is possible for networks smaller than 20 nodes. However, it is possible (as it was to some extent shown in experiments) that slot allocations found using maximum node energy consumption performance index can provide longer network operation. This is achieved thanks to spreading the load evenly and preventing premature battery exhaustion in nodes of greater centrality. To fully exploit the method using the total energy consumption index, it is necessary to include battery capacity as an additional state variable in the network model. Such modification will further complicate the mathematical programming task so that an effective solution could be possible, probably only using the heuristic method.

To enhance method performance and allow finding a solution for more extensive networks, two methods are envisaged. The first may be based on using effective heuristics. Comparing total vs. maximal energy consumption index algorithms shows that they need many branch and bound iterations. One of the reasons may be specific task structure resulting in relatively poor quality of initial upper bounds, especially for more complex initial data locations. So, the heuristic providing any feasible solution may speed up the branch and bound method by providing upper bound and cutting many unnecessary variants. For the same reason, application of any method providing better relaxed solutions like cut and branch or branch and price may be effective. The second approach may be considered complementary to the previous ones. It is important to note that in the experiments, the node to frame allocation was generated randomly. Such a procedure allowed one to test a variety of cases. However, it resulted in very high computation time variance. As it was shown in Section 5, unfavorable initial data locations may cause the increase of computation time in the order of ten times for maximal node energy index variant and even over a thousand times for the total energy index method. Some initial work on classifying initial data location variants was done during the analysis of experiments’ results. In the cases which allow binding sending nodes to frames freely, it may be proposed that the allocations should be generated following some rules. At this moment, it is possible to summarize them as dividing the network into several sectors lying in growing distance from the base station and spreading nodes evenly among these sectors. Such distribution should allow separating the nodes enough to eliminate interference and promote concurrent multi-hop transmission.

Author Contributions: Conceptualization, P.A., A.S. and W.S.; methodology, P.A., A.S. and W.S.; software, P.A.; validation, P.A. and W.S.; formal analysis, P.A., A.S. and W.S.; investigation, P.A. and A.S.; writing—original draft preparation, P.A., A.S. and W.S. All authors have read and agreed to the published version of the manuscript.

Funding: Research was partially funded by the Centre for Priority Research Area Artificial Intelligence and Robotics of Warsaw University of Technology within the Excellence Initiative: Research University (IDUB) programme.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References
1. Al-Qurabat, A.K.M.; Abdulzahra, S.A. An Overview of Periodic Wireless Sensor Networks to The Internet of Things. *IOP Conf. Ser. Mater. Sci. Eng.* 2020, 928, 032055. [CrossRef]
2. Fasolo, E.; Rossi, M.; Widmer, J.; Zorzi, M. In-network aggregation techniques for wireless sensor networks: A survey. *IEEE Wirel. Commun.* 2007, 14, 70–87. [CrossRef]

3. Bahi, J.M.; Makhoul, A.; Medlej, M. An Optimized In-Network Aggregation Scheme for Data Collection in Periodic Sensor Networks. In *Ad-hoc, Mobile, and Wireless Networks*; Li, X.Y., Papavassiliou, S., Ruehrup, S., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; pp. 153–166.

4. Aslam, N.; Phillips, W.; Robertson, W.; Sivakumar, S. A multi-criterion optimization technique for energy efficient cluster formation in wireless sensor networks. *Inf. Fusion* 2011, 12, 202–212. [CrossRef]

5. Niewiadomska-Szynkiewicz, E.; Sikora, A.; Kołodziej, J.; Szynkiewicz, P. Modelling and simulation of secure energy aware fog sensing systems. *Simul. Model. Pract. Theory* 2020, 101, 102011. [CrossRef]

6. Karpowicz, M.; Niewiadomska-Szynkiewicz, E.; Arabas, P.; Sikora, A. Energy and power efficiency in cloud. In *Resource Management for Big Data Platforms: Algorithms, Modelling, and High-Performance Computing Techniques*; Computer Communications and Networking Series; Springer: Berlin/Heidelberg, Germany, 2016; pp. 97–127. [CrossRef]

7. Karpowicz, M.P.; Arabas, P.; Niewiadomska-Szynkiewicz, E. Energy-Aware Multilevel Control System for a Network of Linux Software Routers: Design and Implementation. *IEEE Syst. J.* 2018, 12, 571–582. [CrossRef]

8. Shahbazian, A.; Fereidunian, A.; Manshadi, S.D. Optimal Switch Placement in Distribution Systems: A High-Accuracy MILP Formulation. *IEEE Trans. Smart Grid* 2020, 11, 5009–5018. [CrossRef]

9. Anastasi, G.; Conti, M.; Di Francesco, M.; Passarella, A. Energy conservation in wireless sensor networks: A survey. *Ad Hoc Netw.* 2009, 7, 537–568. [CrossRef]

10. Aziz, A.A.; Sekerciglu, Y.A.; Fitzpatrick, P.; Ivanovich, M. A Survey on Distributed Topology Control Techniques for Extending the Lifetime of Battery Powered Wireless Sensor Networks. *IEEE Commun. Surv. Tutor.* 2013, 15, 121–144. [CrossRef]

11. Rault, T.; Bouabdallah, A.; Challal, Y. Energy efficiency in wireless sensor networks: A top-down survey. *Comput. Netw.* 2014, 67, 104–122. [CrossRef]

12. Lin, D.; Wang, Q.; Min, W.; Xu, J.; Zhang, Z. A Survey on Energy-Efficient Strategies in Static Wireless Sensor Networks. *ACM Trans. Sens. Netw.* 2020, 17. [CrossRef]

13. Gomez, J.; Campbell, A.T. Variable-Range Transmission Power Control in Wireless Ad Hoc Networks. *IEEE Trans. Mob. Comput.* 2007, 6, 87–99. [CrossRef]

14. García Villalba, L.J.; Sandoval Orozco, A.L.; Triviño Cabrera, A.; Barenco Abbas, C.J. Routing Protocols in Wireless Sensor Networks. *Sensors* 2009, 9, 8399–8421. [CrossRef]

15. Singh, V.P.; Kumar, K. Literature Survey on Power Control Algorithms for Mobile Ad-hoc Network. *Wirel. Pers. Commun.* 2011, 60, 679–685. [CrossRef]

16. Avril, F.; Bernard, T.; Bui, A.; Sohier, D. Clustering and Communications Scheduling in WSNs using Mixed Integer Linear Programming. *J. Commun. Netw.* 2014, 16, 421–429. [CrossRef]

17. Rawat, P.; Chauhan, S. Clustering protocols in wireless sensor network: A survey, classification, issues, and future directions. *Comput. Sci. Rev.* 2021, 40, 100396. [CrossRef]

18. Del-Valle-Soto, C.; Mex-Perera, C.; Nolazco-Flores, J.A.; Velázquez, R.; Rossa-Sierra, A. Wireless Sensor Network Energy Model and Its Use in the Optimization of Routing Protocols. *Energies* 2020, 13, 728. [CrossRef]

19. Saba, T.; Haseeb, K.; Ud Din, I.; Almogren, A.; Fati, S.M. EGCR: Energy-Aware Graph Clustering and Intelligent Routing Using Supervised System in Wireless Sensor Networks. *Energies* 2020, 13, 4072. [CrossRef]

20. Fitzgerald, E.; Pióro, M.; Tomaszwski, A. Energy-Optimal Data Aggregation and Dissemination for the Internet of Things. *IEEE Internet Things J.* 2018, 5, 955–969. [CrossRef]

21. Guo, P.; Jiang, T.; Zhang, Q.; Zhang, K. Sleep Scheduling for Critical Event Monitoring in Wireless Sensor Networks. *IEEE Trans. Parallel Distrib. Syst.* 2012, 23, 345–352. [CrossRef]

22. Akylidiz, I.; Vuran, M.C. Wireless Sensor Networks: John Wiley & Sons, Ltd.; Chichester, UK, 2010.

23. Cheng, H.; Guo, R.; Su, Z.; Xiong, N.; Guo, W. Service-Oriented Node Scheduling Schemes with Energy Efficiency in Wireless Sensor Networks. *Int. J. Distrib. Sens. Netw.* 2014, 10, 247173. [CrossRef]

24. Kumar, S.; Kim, H. Energy Efficient Scheduling in Wireless Sensor Networks for Periodic Data Gathering. *IEEE Access* 2019, 7, 11410–11426. [CrossRef]

25. Heinzelman, W.; Chandrakasan, A.; Balakrishnan, H. Energy-efficient communication protocol for wireless microsensor networks. In Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, Maui, HI, USA, 4–7 January 2000; Volume 2, pp. 1–10. [CrossRef]

26. Daanoune, I.; Abdennaceur, B.; Ballouk, A. A comprehensive survey on LEACH-based clustering routing protocols in Wireless Sensor Networks. *Ad Hoc Netw.* 2021, 114, 102409. [CrossRef]

27. Mehmoond, A.; Lloret, J.; Song, H. Improvement of the Wireless Sensor Network Lifetime using LEACH with Vice-Cluster Head. *Ad-Hoc Sens. Wirel. Netw.* 2015, 28, 1–17.

28. Cao, J.G. An Improvement Routing Protocol Based LEACH for Wireless Sensor Networks. *Appl. Mech. Mater.* 2014, 614, 472–475. [CrossRef]

29. Junping, H.; Yuhui, J.; Liang, D. A Time-based Cluster-Head Selection Algorithm for LEACH. In Proceedings of the 2008 IEEE Symposium on Computers and Communications, Marrakech, Morocco, 6–9 July 2008; pp. 1172–1176. [CrossRef]
30. Manshadi, S.D.; Khodayar, M.E. Strategic Behavior of In-Motion Wireless Charging Aggregators in the Electricity and Transportation Networks. *IEEE Trans. Veh. Technol.* **2020**, *69*, 14780–14792. [CrossRef]

31. Chipcon CC2420 Datasheet, Texas Instruments. 2007. Available online: [http://focus.ti.com/lit/ds/symlink/cc2420.pdf](http://focus.ti.com/lit/ds/symlink/cc2420.pdf) (accessed on 6 August 2021).

32. Rappaport, T. *Wireless Communications: Principles and Practice*, 2nd ed.; Prentice Hall PTR: Upper Saddle River, NJ, USA, 2001.

33. Kumar, S.; Raghavan, V.S.; Deng, J. Medium Access Control Protocols for Ad Hoc Wireless Networks: A Survey. *Ad Hoc Netw.* **2006**, *4*, 326–358. [CrossRef]

34. Hekmat, R. *Ad-hoc Networks: Fundamental Properties and Network Topologies*; Springer Netherlands: Dodlerk, The Netherlands, 2006. [CrossRef]