IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 17 February 2024, accepted 11 March 2024, date of publication 20 March 2024, date of current version 28 March 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3379425

RESEARCH ARTICLE

Dominating Sets-Based Approach for Maximizing Lifetime of IoT-Based Heterogeneous WSNs Enabled Sustainable Smart City Applications

BADER ALWASEL[1], AHMED SALIM[1,3], AHMED M. KHEDR[2,3], AND WALID OSAMY[1,4], (Member, IEEE)

[1]Unit of Scientific Research, Applied College, Qassim University, Buraydah 52571, Saudi Arabia
[2]Computer Science Department, University of Sharjah, Sharjah 27272, United Arab Emirates
[3]Mathematics Department, Zagazig University, Zagazig 44519, Egypt
[4]Computer Science Department, Faculty of Computers and Artificial Intelligence, Benha University, Banha 13511, Egypt

Corresponding author: Bader Alwasel (bwasel@qu.edu.sa)

This work was supported by Qassim University, Saudi Arabia, funded by the Deanship of Scientific Research, during the academic year 1445 AH/2023 AD under Grant 2023-SDG-1-BSRC35303.

ABSTRACT The versatility and diverse applications of IoT-based Heterogeneous WSN (HWSN) technologies make them valuable tools for achieving sustainability goals in Sustainable Smart Cities (SSCs). Proper management of underlying heterogeneous architecture is crucial for the successful operation of smart applications. To prolong the WSN's lifespan and avoid failures, effective energy management is essential. Although researchers are continually exploring heterogeneity in WSN, it gets more and more important to create cost-effective paradigms that cover multiple facets of SSC while ensuring their stability and reliability. The concept of Dominating Sets (DS) in a graph can be leveraged to minimize resource utilization in WSNs by arranging nodes into disjoint DS, with only one set executing duties at any given time. In this work, we propose a novel technique for IoT-based HWSNs-enabled SSC, called EADDSA, utilizing the DS concept to plan the sleep-and-awake scheme for heterogeneous nodes, based on their resource capabilities. We propose a new algorithm, called the Energy Attentive Algorithm (EAA), to find disjoint DSs that are energy-aware. EAA algorithm attentively tries to form the set that maximizes lifespan while adhering to DS conditions in each iteration. EADDSA further incorporates an effective DS scheduling strategy to enhance the HWSN lifetime by establishing operational guidelines for each round, taking into account the estimated lifetimes and the designated number of working rounds for each DS. This enables efficient allocation of data sensing and gathering tasks across the network minimizes resource usage, and extends network lifetime.

INDEX TERMS Dominating set, heterogeneous wireless sensor networks, network lifetime, sustainable urbanization, urban problems, energy consumption, sustainable smart city, modern cities.

I. INTRODUCTION

In the face of accelerating urbanization globally, the concept of a Sustainable Smart City (SSC) has emerged as a crucial solution to address the challenges brought about by rapid urban development. With over 50 percent of the world's population projected to reside in cities by 2050, SSCs play a pivotal role in providing essential amenities and services to meet the growing demands of urban living. To ensure the sustainable operation of different SSC components, such as Smart Governance, Smart Water and Waste Management, and Smart Connectivity, innovative strategies are imperative. One key technological enabler for achieving sustainability goals in SSCs is the Internet of Things (IoT)-based Wireless Sensor Networks (WSNs). These networks, consisting of heterogeneous nodes with varying capabilities, are responsible for acquiring and transferring data for smart applications. Utilizing a multi-hop scheme, these nodes

The associate editor coordinating the review of this manuscript and approving it for publication was Zhangbing Zhou.

establish connections to facilitate the transfer of accumulated data towards a designated sink node. Unlike the original data collection nodes in WSN, sink nodes are equipped with abundant energy, robust computing capabilities, extended communication ranges, and adequate memory [3], [5], [23], [24]. Despite the rapid expansion of real-world applications that include monitoring in agricultural areas, homes, healthcare, and animal and industrial contexts [8], WSNs face challenges related to limited battery capacity, propagation range, storage constraints, and computing resources [6], [7]. The cost associated with augmenting the energy of individual nodes is considerably lower than deploying additional nodes with homogeneous energy levels. This highlights the influence of energy heterogeneity in WSNs. Simultaneously, managing battery power and lifespan, as well as sustaining diverse quality of service parameters, pose significant challenges in Heterogeneous WSNs (HWSNs). The constraints on node energy contribute to survivability issues and potential failures. A key concern in WSNs is the refinement of energy efficiency, a factor that profoundly impacts the stability and lifespan of the network. Managing battery power and addressing survivability issues are fundamental considerations in enhancing the efficiency of WSNs. As SSCs continue to evolve, the interconnectedness between sustainable urban development and the versatile applications of WSN technologies remains a crucial aspect in shaping the future of smart cities. Furthermore, the network's capacity to withstand various challenges and failures, ensuring the accomplishment of designated tasks, remains a fundamental consideration in WSNs [8], [9].

Numerous strategies have been proposed in the existing body of literature to address the goal of conserving energy and prolonging the lifetime of networks. Sensing and communication operations stand out as the primary contributors to energy consumption, making effective energy management and scheduling crucial for extending the service life of WSNs. A common approach to energy preservation in networks involves transitioning nodes between sleep and awake states, as discussed in [25]. Given the typically dense deployment of WSN nodes, temporarily suspending the sensing and radio communication capabilities of nodes in the sleep state proves to be an efficient solution for energy conservation. In this approach, only nodes within the active set are tasked with data sensing and collection from the target area, while the remaining nodes remain in a low-power sleep state. The state transition mechanism plays a pivotal role in delivering prolonged service while minimizing power consumption.

This study leverages the Dominating Sets (DS) concept to regulate and manage the sleep/awake schedule of heterogeneous nodes based on energy considerations in WSNs. A graph's DS refers to a subset of nodes, where each node either belongs to the subset or is adjacent to at least one node within that subset. In the context of minimizing energy consumption, an effective strategy involves organizing nodes into disjoint sets, ensuring that only one set performs tasks at any given time [11]. This concept, known as sleep-awake scheduling, can be implemented by selecting disjoint dominating sets to prolong the network's lifespan and distributing data-gathering tasks across these sets. In this paper, we propose a DS-based approach, called EADDSA, for maximizing the lifetime of IoT-based HWSN-enabled SSC applications. We tackle the challenge of energy conservation in HWSN through the introduction of an algorithm for identifying energy-aware disjoint DSs. We also devise an effective scheduling strategy aimed at maximizing the HWSN lifetime. By choosing several disjoint DSs according to the energy level of nodes, our goal is to enhance the overall lifespan of a heterogeneous WSN with nodes possessing varying levels of initial energy. The careful choice of disjoint DSs can contribute significantly to extending the WSN lifespan, defined as the sum of the lifetimes of the disjoint DSs. Our primary focus is on incorporating energy considerations as a prominent factor in forming dominating sets, proposing algorithms that construct and schedule sets for prolonged lifetimes.

The main contributions of this paper are as given below:

- We tackle the challenge of energy conservation in heterogeneous WSN through the introduction of a novel DS-based technique, named EADDSA. With this technique, we aim to maximize the lifetime of IoT-based HWSN-enabled SSC applications.
- We propose a new algorithm, called the Energy Attentive Algorithm (EAA), to find disjoint DSs that are energy aware. The EAA algorithm attentively tries to form the set that maximizes lifespan while adhering to DS conditions in each iteration. This is achieved by prioritizing higher-energy nodes in the formation of each set, ensuring efficient energy distribution.
- We devise an effective DS scheduling strategy to enhance the HWSN lifetime by establishing operational guidelines for each round, taking into account the estimated lifetimes and the designated number of working rounds for each DS.

The remainder of this paper unfolds as follows: In Section II, a comprehensive review of related works is provided. The details of the proposed work are expounded in Section III. Section IV unveils the results of simulations conducted on the proposed scheme, accompanied by a performance comparison with existing approaches. Finally, Section V encapsulates the concluding remarks of the paper.

II. RELATED WORK

The rapid increase of smart sensing devices within SSCs has generated a great need for algorithmic approaches that are both resource-efficient and adaptive to the limited resources and heterogeneous characteristics of WSNs [1], [2], [3]. The successful realization of technological advancements in SSCs relies on a robust infrastructure as well as the integration of e-governance policies for efficient resource management [4], [8]. The integration of WSN and analogous technologies offering a wide array of applications within SSCs holds the potential to yield comprehensive strategies, fostering the development of sustainable communities [9], [10], [11]. Data insights provided by WSNs can be employed to make smart cities more sustainable by empowering city planners and decision-makers to make informed decisions that promote the development of sustainable urban environments. Increasing the lifetime of WSNs can reduce environmental impact, improve data reliability, and free up resources for other sustainable development initiatives, contributing to sustainable urban development [12], [13], [14]. Consequently, there is a growing urge to create paradigms that cover different facets of SSCs and conserve resources. While the majority of studies concentrate on developing energy-saving methods for WSNs with homogeneous devices in terms of their energy reserves, connection, and processing skills, heterogeneous networks feature devices with varying resource capabilities [15], [16], [17]. Conserving limited resources and managing heterogeneity is crucial for the successful implementation and sustainability of SSC applications. Efficient design and resource management can lead to reduced environmental impact, cost savings, and improved quality of life in smart cities [20], [21], [22].

Based on several aspects impacting power utilization, experts have suggested many solutions to address energy restrictions in WSNs. A sleep/awake scheme, according to several research studies [8], [9], [10], is an effective way of conserving energy in nodes. The phenomenon of sensing and transmission are the two main energy-intensive processes. Previous research has used the DS approach, in which the nodes are grouped into disjoint DSs with a single set executing duties at any given point, to reduce the utilization of energy [16], [17]. However, the problem of determining the minimum DS required to represent a WSN is regarded as NP-Hard [12]. The authors of [16] and [17] presented centralized approximation techniques, with approximation factors $O(\log n)$ and $O(n \log n^2)$, respectively. The DS technique employing dual greedy approaches to control the sleep/awake scheme in WSN was first presented in [16]. It was further enhanced by [17] employing the concept of maximum disjoint DSs expressed as Domatic Partition (DP) associated with Unit Disk Graphs (UDGs), to address the partitioning problem for homogeneous networks. To overcome the limitations of centralized schemes in large-scale networks, [18] and [21] made use of distributed algorithm properties to create DSs. In [20], DSs are generated using a distributed algorithm by employing the Maximal Independent Set (MIS) principle indirectly. By employing the three-stage approach and leveraging the concept of Pseudo DSs, the algorithm in [19] created smaller Connected DSs (CDS) while achieving improved network connectivity and coverage. By utilizing a non-trivial potential function, [9] improved the CDS connectivity. The suggested greedy algorithm generates three-connected components from Tutte's decomposition of a two-connected graph.

To enhance network performance, [13] proposed a clustering approach (EBDSC) that relies on DS. A novel decentralized technique for choosing the WSN nodes that will make up a DS was proposed by [14]. A Wait Before Start (WBS) method was created to enable any dominant node to announce itself according to the number of adjacent nodes. This helps the DS to be determined automatically. Every value is also time-weighted, allowing it to begin executing its software after an established period depending on the value it holds. To overcome the problem of DP, [22] presented a cell layout. To accomplish this, a clique was built for every cluster in WSN. They implemented a distributed nucleus algorithm (DNA) and demonstrated that it executes in definite iterations under the congested message-passing model.

All of the aforementioned algorithms solved the problem of identifying the minimum DS. While allowing various levels of starting energy for nodes, [11] created CDS using different kinds of local search strategies. However, the primary drawback is the level of complexity of their methods. One of the most significant constraints in WSNs is a scarcity of resources. As a result of having small, low-power capacity batteries, sensor nodes are prone to failure. The failure hurts the reliability and effectiveness of WSN-based services. This constraint poses several challenges and potential issues for QoS performance and reliability, especially in the context of smart cities that rely on IoT-based heterogeneous WSNs to support different applications with varying requirements like data rate, delay, and expended power.

Motivated by this, we use the concept of DS to regulate and establish an effective sleep-and-awake scheme for heterogeneous WSNs, where we design a new solution to create the disjoint DSs based on multiple criteria. Compared with existing algorithms, obtaining the maximum network lifetime is a significant achievement in the context of IoT-based heterogeneous WSNs-enabled SSC applications to ensure sustainable and prolonged operation while meeting application requirements. By proposing solutions that can enhance network lifetime, we will be able to enhance the overall efficiency, reliability, and service quality of the underlying application.

III. PROPOSED EADDSA TECHNIQUE

This section introduces the EADDSA technique, a novel approach developed in this study. The EADDSA technique incorporates the Energy Attentive Algorithm (EAA), which is designed to form the sets in each iteration that maximize the HWSN lifetime while adhering to DS conditions. This is achieved by prioritizing nodes with higher energy levels. Furthermore, the presented work incorporates a scheduling technique designed for the DSs generated by the EAA. Figure 1 illustrates the overall flow of the EADDSA technique.

We begin with a description of the system model and the formulation of the problem. Subsequently, we offer a detailed description of the proposed EADDSA technique.

A. SYSTEM MODEL AND PROBLEM FORMULATION

In our proposed framework, we represent the WSN as a graph, denoted as $G = (V, E)$, where V and E represent the set

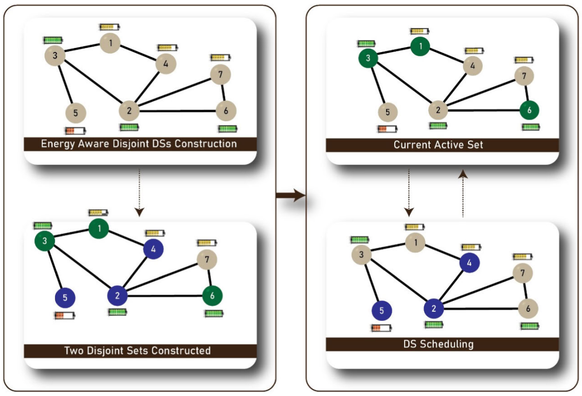

FIGURE 1. The overall flow of the proposed EADDSA technique.

of nodes and the communication links between them. If we take nodes i and $j \in V$, an edge between them exists upon i being within the communication range of j. A fundamental assumption in our model is the symmetry of communication links between nodes within the WSN, meaning that all nodes in the network share an identical communication range. Despite this symmetry, the battery-powered nodes exhibit heterogeneity in their initial energy levels. Additionally, we consider a constant energy usage rate across all nodes in the network.

A subset $D \subset V$ is identified as a DS if $\forall v \in V \backslash D$ there exists a neighbor in D. In the DS concept, nodes designated to execute tasks on behalf of their neighboring nodes form the DS, offering a means to enhance the network lifespan. A DS with a smaller size tends to exhibit superior energy efficiency, allowing more nodes to enter sleep mode. However, a notable drawback of this approach lies in the uneven distribution of energy among nodes. To address this imbalance, a strategy involves substituting an existing DS with another disjoint DS within the WSN and transitioning the nodes in the previous set to sleep mode. This entails employing as many disjoint DSs' as possible, utilizing each for a specific duration, and subsequently replacing it with another set, and so forth.

The primary goal is to improve the lifespan of HWSN by carefully selecting disjoint DSs. The lifespan of a DS is dependent upon the node with the lowest energy, factoring in its energy consumption rate. The cumulative network lifetime is calculated as the sum of the lifetimes of all disjoint DSs.

In defining the constraints, the following considerations are taken into account:

1) Energy Constraint: The total expended energy of all nodes in a DS should not exceed the complete available energy of the WSN, denoted as E_{total}: $\sum_{k=1}^{j} s_{ik}.E \leq E_{total}$, where $s_{ik}.E$ represents the energy of node i in DS $D_k \subseteq V$, and j is the disjoint DSs count.
2) Disjointness Constraint: Each pair of DSs should have no common nodes: $D_i \cap D_j = \emptyset$ for all D_i, $D_j \subseteq V$, ensuring that no nodes are shared between different DSs.
3) Connectivity Constraint: Every node outside of a DS should have at least one neighbor in some DS: For each $v \in V \setminus D$, there should exist a DS D so that $D \subset V$ and v has at least one neighboring node in D.
4) Positive Energy Constraint: The energy of each node in a DS should be greater than zero, considering a positive constant δ: $s_{ik}.E > 0$ for all $s_{ik} \in D_k \subseteq V$, with the energy consumption rate $\delta > 0$.

To achieve the objective of maximizing the network's lifespan, we formulate the objective function F as below:

$$F = \max\left(\sum_{k=1}^{j} \frac{E_{min}}{\delta}\right),$$
$$E_{min} = \min(s_{1k}.E, s_{2k}.E, \ldots, s_{ik}.E) \tag{1}$$

Within the objective function, the term $s_{ik}.E$ denotes the energy of node i within the DS $D_k \subseteq V$. The variable j signifies the count of disjoint dominating sets, and δ represents the energy consumption rate. The objective function is designed to maximize the sum of the minimum energy levels among nodes within each DS, with this sum then divided by the energy consumption rate δ. By expressing the constraints and the objective function in this manner, the aim is to optimize the configuration of energy-aware disjoint DSs, ultimately maximizing the lifespan of the WSN.

1) EXAMPLE

To illustrate, let's consider a HWSN with seven nodes, each having varying energy levels, as depicted in Figure 2. In this, we can identify two disjoint DSs: $D_1 = [1, 2, 5]$ and $D_2 = [3, 4, 6]$. In D_1, node 5 holds the minimum energy, while node 4 in D_2 possesses the lowest energy. Assuming a constant rate of energy utilization, given by 0.05, the lifetime of D_1 is calculated as 0.03 / 0.05 = 0.6, and the lifetime of D_2 is 0.18 / 0.05 = 3.6. Consequently, the total WSN is 4.2. Optimizing and enhancing the total WSN lifetime is achievable by selecting alternative DSs, such as $D_1 = [2, 4, 5]$ and $D_2 = [1, 3, 6]$. In this scenario, the overall WSN lifetime becomes (0.03 / 0.05) + (0.76 / 0.05) = 0.6 + 15.2 = 15.8. This demonstrates that careful consideration and optimization of DSs can significantly improve the total WSN lifespan.

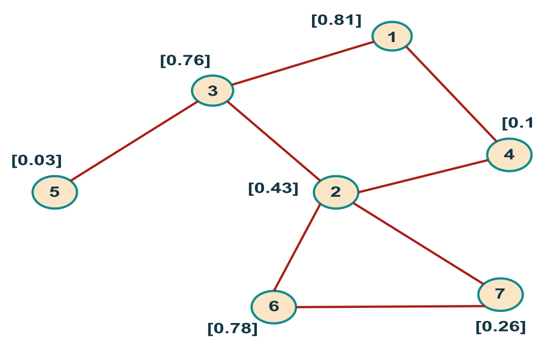

FIGURE 2. Heterogeneous WSN example.

B. EAA: ENERGY ATTENTIVE ALGORITHM

We discuss the proposed Energy Attentive Algorithm (EAA) for Energy-Aware Disjoint DSs construction. EAA operates in iterations, in an iteration $r(\ r > 1)$, we determine a DS S_j, where $S_j \bigcap S_k = \phi$, for $j \neq k,\ j, k \geq 1$, and S_j, $S_k \subseteq V$.

In iteration 1, $S'_1 = V$, where S'_1 is a temporary set and gradually will be shrunk as nodes are eliminated from it while the algorithm execution proceeds. The nodes in S'_1 are sorted in ascending manner according to their energies. Select node $v \in S'_1$ with minimum energy then check if $S'_1 \setminus v$ will violate DSs constraints. If not, eliminate v from the set S_1 and insert it into V'_1 (V'_1 is the set of candidate nodes for the next iteration). Then, update the neighbors of v by eliminating it from its neighbors. Otherwise, v is kept as a dominated node in S'_1. We repeat this process until the entire nodes in the set V'_1 are visited in ascending manner. Assign $S_1 = S'_1$ and color each node in S_1. At this stage, we formed the first DS S_1 and we have the candidate nodes in set V'_1 for the next iteration that may form a new DS. The algorithm stops when we cannot find a new disjoint DS.

Now, we describe how a DS S_r is created at iteration $r > 2$ in general. First $S'_r = V'_r$ (S'_r denotes a temporary set at iteration r). As before, nodes in S'_r are sorted in ascending manner according to their energies. Then iteratively check if the elimination of each node v violates the dominating set condition or not. The nodes with minimum energy are eliminated only if they do not cause a violation of the DS condition. The removed nodes are then moved to V'_{r+1}. The set S_r is formed from the set S'_r which contains nodes with maximum energy at this iteration and does not violate the DS condition. This process is continued until one of the following conditions is satisfied:

- The remaining nodes violate the condition of dominating sets and cannot form new sets.
- There are no remaining nodes, i.e., all nodes have formed the disjoint dominating sets.

The details of the EAA algorithm are given in Algorithm 1. DS scheduling technique then operates on the set of DSs to schedule them.

1) CLARIFICATION EXAMPLE

For clarification, consider the example given in Figure 2. The energy of each node is shown below its ID. In the first iteration, S'_1 contains all sorted nodes according to their energies in ascending order $S'_1 = \{5, 4, 7, 2, 3, 6, 1\}$. Since S'_1 is a valid dominating set after eliminating node 5 (minimum energy node), S'_1 will be updated to $S'_1 \setminus 5$ and node 5 will be inserted into the next candidate set V_1. The same process will be repeated till we have the first valid dominating set $S_1 = \{1, 3, 6\}$. In the second iteration the 2^{nd} temporary set contains $S'_2 = \{5, 4, 7, 2\}$ (this set is initialized as V_1). Again eliminate node 5 (node with minimum energy) and check if the S'_2 is still a valid dominating set or not. Node 5 is kept since we can not eliminate it. The same thing for nodes 4 and 2, we cannot eliminate them. As in previous iteration, we have $V_2 = \{7\}$ and $S'_2 = \{4, 2, 5\}$. Therefore, this iteration ends with $S_2 = \{2, 4, 5\}$ and 3^{rd} iteration starts with $S'_3 = \{7\}$. The algorithm stops with S_1, S_2. Since S_3 violates the dominating set conditions, it cannot be considered for output. The result will be the dominating sets $\{1, 3, 6\}\ and\ \{2, 4, 5\}$ with lifetime 15.80.

Algorithm 1 Energy Attentive Algorithm (EAA)

1: V**:** set of nodes.
2: r**:** iteration number (initially equals 1).
3: S**:** set of disjoint dominating sets and initially equals ϕ.
4: S_r**:** DS at iteration r.
5: S'_r**:** temporary dominating set at iteration r.
6: V_r**:** set of candidate nodes at iteration i initially equals ϕ.
7: **while** $r > 0$ **do**
8:     $S_r = V$
9:     Sort S_r ascending according to energy
10:     **for** each node $s \in S_r$ **do**
11:         $S'_r = S_r \setminus s$
12:         **if** S'_r do not violate the dominating set conditions **then**
13:             update s neighbors by eliminating s from their list
14:             $S_r = S'_r$
15:             $V_r = V_r \cup s$
16:         **end if**
17:     **end for**
18:     **if** S_r do not violate the dominating set conditions and $S_r \neq \phi$ **then**
19:         $S = S \bigcup S_r$
20:         $V = V_r$
21:         Increment r by 1
22:     **else**
23:         break
24:     **end if**
25: **end while**

The description of the Algorithm 1 is as follows: Lines (1 to 6) are initialization steps. Initialize the set of nodes, V, which represents all the nodes in the network (line 1). Set the iteration number, r, to 1 and Initialize the set of disjoint dominating sets, S, as an empty set (lines 2, 3). Define S_r as the dominating set at iteration r and Define S'_r as the temporary dominating set at iteration r (lines 4, 5). Define V_r as the set of candidate nodes at iteration r, initially empty (line 6). Enter the while loop with the condition $r > 0$ (line 7). Set S_r to the set of all nodes (V) (line 8). Sort the nodes in S_r in ascending order according to their energy levels (line 9). Iterate through each node s in S_r (lines 10 -17). Create a temporary dominating set, S'_r, by removing node s from S_r(line 11). Check if S'_r satisfies the conditions of being a DS (line 12). In Line 13, if S'_r satisfies the DS conditions then execute lines 13-15. Update the neighbor lists of node s (line 13), set S_r to S'_r (line 14), and add node s to V_r (line 15). Line 18 checks if S_r satisfies the dominating set conditions and is not an empty set. If S_r satisfies the conditions, add S_r to S, update V to V_r, and increment r by 1 (lines 19-21); else exit the while loop (lines 22-25). The algorithm terminates when no valid DSs are found or r becomes 0. The output is the set of DSs, S, found during the iterations.

C. DS SCHEDULING ALGORITHM

Upon identifying all DSs, we focus on devising an effective scheduling strategy aimed at maximizing the HWSN lifetime. A commonly employed method for scheduling the final set of disjoint DSs is the Round-Robin scheduling approach, due to its simplicity and low complexity. This technique entails each DS in the set taking turns to perform tasks in a circular order until all DSs have completed their assignments. The number of rounds directly impacts the network lifetime, constrained by the count of disjoint DSs. However, this method poses an issue by potentially causing premature energy depletion of the first node. This arises due to the evaluation of each DS being based on the minimum energy level within it. To further enhance the network lifetime, an alternative heuristic scheduling mechanism can be employed. This mechanism ensures optimal DS scheduling for each round by assessing the lowest node energy within each DS during the present time slot, effectively serving as an indicator of the DS's remaining lifetime.

For clarification, let's consider a network with two DSs, ds_1 and ds_2, where ds_2 has the highest estimated lifetime and ds_1 has the lowest lifetime. If ds_1 and ds_2 operate as active DSs for an equal number of rounds, ds_1 will experience premature depletion, leading to a shorter network lifetime compared to ds_2. This disparity significantly influences the overall lifespan of the network.

The DS scheduling algorithm is presented in Algorithm 2, which determines the working policy for the next round.

Algorithm 2 DS Scheduling Algorithm

m: DSs count within the network.
r: the round.
C: set of working DSs in the following round, initialized as $C = \phi$.
Determine each DS's lifetime $(\tau_1, \tau_2, \ldots, \tau_m)$.
for each DS ds_i **do**
    Calculate $NWR_i = \lceil \tau_i \rceil$.
end for
Sort DSs based on NWR_i in descending order.
for each DS $ds_i, i \ldots m-1$ **do**
    Add ds_i to C for NWR_i-1 rounds {schedule ds_i to work in round r}.
end for
Add ds_m to C for NWR_m rounds {schedule ds_m to work in round r}.
Add $ds_i, i \ldots m-1$ to C for one round{schedule ds_i to work in round r}.

The BS formulates the working policy for the rounds through the following steps:

1) The BS initiates the process by estimating the lifetime of each DS, denoted as $\tau_1, \tau_2, \ldots, \tau_m$, where m represents the total DSs count in the network, and τ_i signifies the lifetime of DS i.
2) Subsequently, the BS calculates the number of working rounds (NWR) for each DS_i using the formula $NWR_i = \lceil \tau_i \rceil$.
3) The DSs are then organized in descending order based on their respective NWR values.
4) The BS proceeds to schedule the DSs according to the following criteria:
   - Assign DS_1 to operate for $NWR_1 - 1$ rounds.
   - Assign DS_2 to operate for $NWR_2 - 1$ rounds.
   - Assign DS_i, where i ranges from 3 to $m-1$, to operate for $NWR_i - 1$ rounds.
   - Assign DS_m, characterized by the least lifetime, to operate for NWR_m rounds.
   - Finally, for each DS_i, where i ranges from 1 to $m-1$, assign it to operate for 1 round.

Based on this, the BS establishes the operational guidelines for the upcoming round, taking into account the estimated lifetimes and the designated number of working rounds for each DS.

1) EXAMPLE

To clarify, consider a network with three DSs: $ds1$, $ds2$, and $ds3$. Let's assume the lifetimes of $ds1$, $ds2$, and $ds3$ are $\tau_1 = 10$, $\tau_2 = 7$, and $\tau_3 = 3$, respectively.

1) The BS estimates the lifetime of each DS:
$$\tau_1 = 10$$
$$\tau_2 = 7$$
$$\tau_3 = 3$$
2) The BS then estimates the number of working rounds (NWR) for each DS_i:
$$NWR_1 = \lceil \tau_1 \rceil = \lceil 10 \rceil = 10$$
$$NWR_2 = \lceil \tau_2 \rceil = \lceil 7 \rceil = 7$$
$$NWR_3 = \lceil \tau_3 \rceil = \lceil 3 \rceil = 3$$
3) Sorting the DSs in descending order based on their NWR values:
   - DS1: $NWR_1 = 10$ rounds
   - DS2: $NWR_2 = 7$ rounds
   - DS3: $NWR_3 = 3$ rounds
4) The BS schedules the DSs as follows:
   - Assign DS1 to work for $NWR_1 - 1 = 10 - 1 = 9$ rounds.
   - Assign DS2 to work for $NWR_2 - 1 = 7 - 1 = 6$ rounds.
   - Assign DS3 to work for $NWR_3 = 3$ rounds.
   - Assign DS1 to work for 1 round.
   - Assign DS2 to work for 1 round.

As each DS is assessed based on the minimum node energy within it, considering the aforementioned values reveals that the initial node's depletion happens after 18 rounds of the node with the minimal energy in ds3. In contrast, utilizing the Round-Robin scheduling method results in the initial node's exhaustion occurring after 9 rounds. Figure 3 illustrates the depiction of our scheduling approach compared to Round-Robin scheduling.

D. ANALYSIS

Lemma 1: EAA algorithm guarantees that it can always generate the DS S.

Proof: The proof will be by contradiction. A set S is a DS if and only if there exists no node s in the graph G that

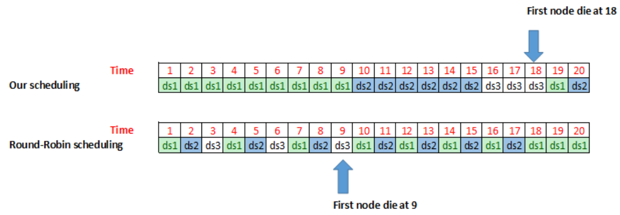

FIGURE 3. Depiction of ds1, ds2 and ds3 scheduling.

is not connected to a node in S (or covered by a node in S). Consider S as a DS and suppose that at least one node s in G is not connected to S. This implies that node s or some of its neighboring nodes are not connected to a node in S, which means S is not a DS. This leads to a contradiction if we assume that all nodes are connected in G and S is not a DS. Given that all nodes are interconnected, each node is influenced or dominated by some nodes in the set S. This contradicts the initial assumption that S is not a DS.

The EAA starts with set S1, which covers all nodes in graph G. To form a valid DS, set S1 is updated attentively by eliminating nodes that have minimum energy and do not violate the dominating set conditions. The eliminated nodes form a new set of candidate nodes for the next iteration. The process is repeated considering the new candidate node set to generate a new valid DS. This means that the algorithm's operation is performed to keep the DS valid.

Lemma 2: Given a WSN modeled as a connected graph G(V, E), EAA will create the set of DSs in polynomial time.

Proof: Assuming that the overall amount of nodes within the network is n ($|V| = n$) and the average count of nodes of a candidate DS is k, the time analysis of executing EAA will be as follows:
The while loop will be executed r iterations in the worst case. In each iteration,

- In step 9 (sorting a set): if the nodes count in S is $k < |V|$, then the time for this step will be $\mathcal{O}(k * log\ k)$ using the sorting algorithm.
- In steps 10-17, EAA updates neighbor info. this update is performed in $\mathcal{O}(k)$ time.
- In steps 18-24, EAA checks the conditions of DS in $\mathcal{O}(n-k)$.

Therefore, the overall time complexity of EAA is $\mathcal{O}(r * k^2 * (n-k) * logk)$.

IV. RESULTS AND DISCUSSION

This section initially presents the simulation settings, followed by the simulation results and comparative analysis with the other baseline algorithms that include minimum dominating set algorithm (MDP) [17], max energy first minimum dominating set algorithm (MFMDP), local search (LSearch), variable depth (VDepth) and fixed depth (FDepth) [11]. MFMDP follows MDP algorithm with some modifications in which the highest energy node in the network is selected first.

A. SIMULATION SETTINGS

MATLAB R2015a is employed for the performance evaluation of the proposed EEA. MAX-DPA technique [15] is adopted for random deployment of WSNs. We considered different deployment scenarios during simulation, where 50, 100, 150, and 200 nodes are considered with varied network densities. For each node deployment, sparse and dense network graphs are considered, and the starting energy of each node is allotted a random value $r \in (0, 1]$. Moreover, as in [11], the rate of energy depletion is set to 0.05. The results are captured by averaging 50 different randomly generated WSNs using MAX-DPA algorithm [15]. An example of the generated dense and sparse network graphs is depicted in Figure 4.

B. EVALUATION RESULTS

In this test, we have 4 scenarios, each scenario undergoes 50 executions, and the obtained results are averaged for analysis.

Figures 5-9 show varying network densities and the corresponding network lifetime for 50, 100, 150, and 200 nodes. In the 50-node scenario (as in Figure 5), the MDP consistently has values ranging from 16.24 to 28.94 across the network densities. The MFMDP also consistently has values ranging from 9.24 to 13.34 across the network densities. It exhibits relatively stable performance but generally has lower values compared to MDP. LSearch has values ranging from 27.21 to 56.7 across the network densities. It shows a wider range of values and generally performs better than both MDP and MFMDP. VDepth has values ranging from 33.1 to 66.24 across the network densities, demonstrating higher values compared to the previous algorithms, which suggests better performance. FDepth has values ranging from 32.68 to 57.5 across the network densities, similar to VDepth in terms of performance, and it consistently performs better than MDP and MFMDP. The proposed EADDSA has values ranging from 30.1137 to 77.887 across the network densities which exhibits the highest values among all the above algorithms.

In the 100-node scenario (as in Figure 6), In terms of network lifetime, MDP has values ranging from 18.7 to 28.81, and MFMDP has values ranging from 12.42 to 18.1, which are lower than those of MDP, suggesting a shorter network lifetime ad it is clear that MFMDP performs worse than MDP. LSearch performs better than MFMDP but worse than MDP in terms of network lifetime. The VDepth shows better performance compared to LSearch. FDepth performs similarly to VDepth but slightly worse than LSearch. The proposed EADDSA has values ranging from 35.4694 to 141.5906, which is higher than all the above algorithms, i.e., EADDSA shows the best performance among the above algorithms in terms of network lifetime.

In the 150-node and 200-node scenarios (as in Figures 7 and 8), the EADDSA algorithm consistently demonstrates the highest values for network lifetime across all network densities, it outperforms all other algorithms and shows the

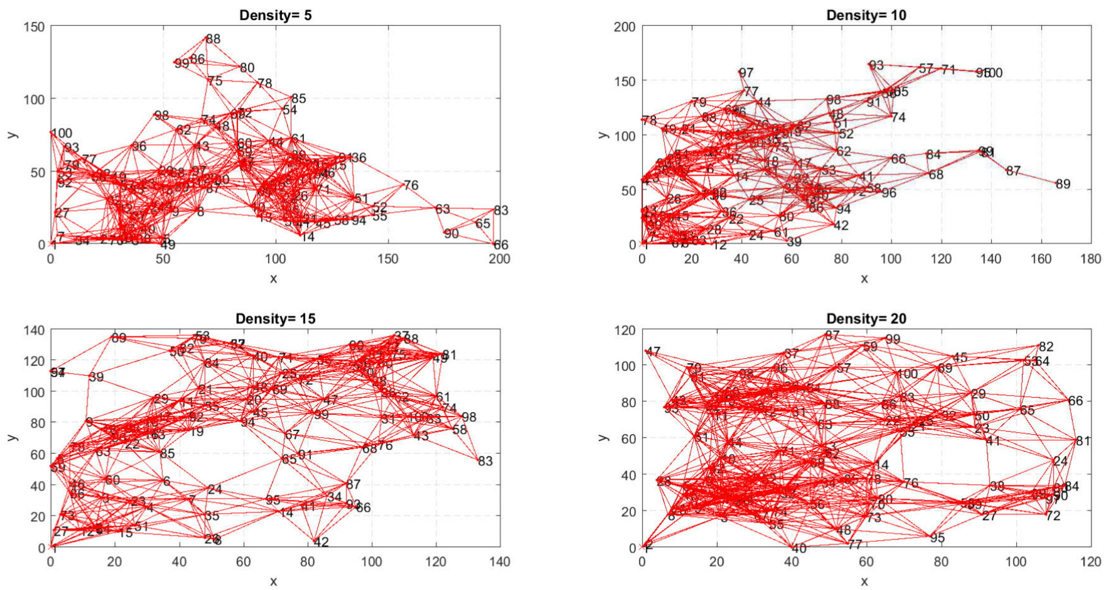

FIGURE 4. Example: randomly generated sparse and dense graphs with 100 nodes.

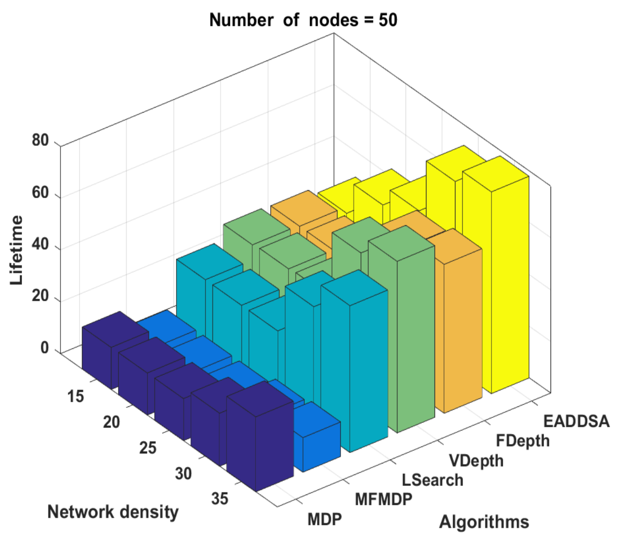

FIGURE 5. Comparison: varying network densities and resultant network lifetime (scenario: 50 nodes).

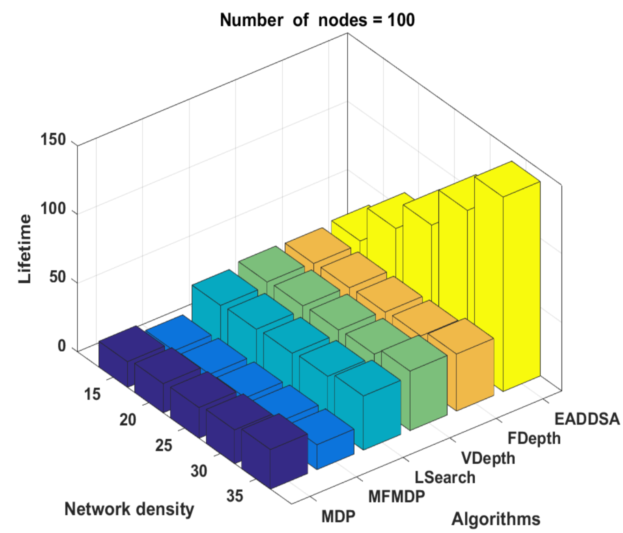

FIGURE 6. Comparison: varying network densities and resultant network lifetime (scenario: 100 nodes).

best performance in terms of network lifetime. The VDepth performs consistently well, showing better performance than most of the other algorithms but falling behind EADDSA. The FDepth algorithm also performs well, showing a similar level of performance as VDepth but slightly lower. The LSearch is performing relatively well but being outperformed by both VDepth and FDepth. The MDP performing adequately well but falling behind the previously mentioned algorithms. The MFMDP has the lowest values for network lifetime, and it shows the lowest performance among the provided algorithms.

It is apparent that as the network density increases, the network lifetime values for all algorithms generally increase as well. Moreover, from Figure 9, it can be observed that MDP and MFMDP generally exhibit a positive correlation between network lifetime and the number of nodes, indicating higher performance with larger networks. LSearch's performance remains consistent, while VDepth and FDepth show a slight decrease in performance with more nodes. EADDSA consistently displays the highest network lifetime across different numbers of nodes.

Furthermore, from Figures 5-9, it is evident that the network lifetime performance of MFMDP, and MDP schemes are less than LSearch, VDepth, and FDepth. The reason behind the improved performance of LSearch, VDepth, and FDepth is that they allow swapping among the nodes of the different DSs which leads to improved network lifetime. It is evident from these Figures that the proposed EADDSA

TABLE 1. Times of improvement of EADDSA over other algorithms.

Network density	MDP	MFMDP	LSearch	VDepth	FDepth
Number of nodes =50					
15	0.85	2.26	0.11	-0.09	-0.08
20	1.66	3.67	0.59	0.28	0.32
25	2.13	4.50	0.87	0.49	0.56
30	2.52	4.94	0.55	0.21	0.32
35	1.69	4.84	0.37	0.18	0.35
Number of nodes =100					
20	0.90	1.86	0.14	0.04	0.07
30	1.97	3.54	0.95	0.79	0.85
40	2.76	4.52	1.51	1.33	1.40
50	3.60	5.71	2.22	2.05	2.14
60	3.91	6.82	2.59	2.27	2.41
Number of nodes =150					
30	1.80	3.37	0.68	0.60	0.64
40	2.83	4.93	1.33	1.24	1.28
50	3.70	5.85	1.99	1.83	1.87
60	4.48	6.74	2.51	2.29	2.44
70	4.37	7.28	2.70	2.49	2.62
Number of nodes =200					
40	2.69	4.88	1.33	1.14	1.20
50	3.48	6.15	1.81	1.72	1.75
60	4.82	7.47	2.54	2.42	2.44
70	5.31	7.91	3.04	2.74	2.85
80	5.24	8.61	3.30	3.12	3.28

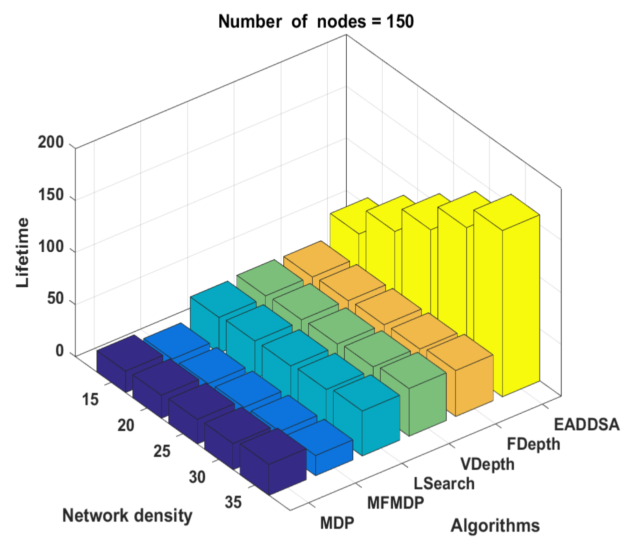

FIGURE 7. Comparison: varying network densities and network lifetime for the algorithms (scenario: 150 nodes).

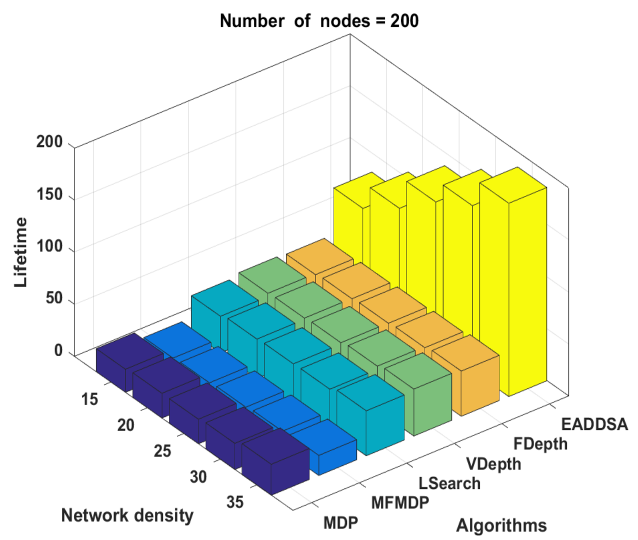

FIGURE 8. Comparison: varying network densities and resultant network lifetime (scenario: 200 nodes).

offers higher network performance than other algorithms because EADDSA not only considers the sensor node energy while generating the CDS but also considers the lifetime of each CDS.

Figure 9 shows that the EADDSA has the best network lifetime over the different network densities and using different number of nodes. Because, in each iteration, the EADDSA aims to group the nodes with high energy in the same CDS and the nodes with low energy in the same CDS. By doing this, the CDS lifetime has been increased, which indirectly results in a longer lifetime for a network as well.

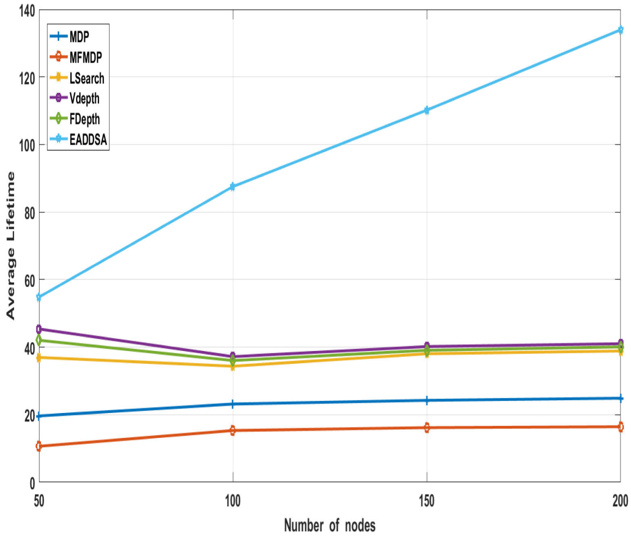

FIGURE 9. **Comparison: average network lifetime for the algorithms across all densities.**

Moreover, we can conclude from the results that the low performances of LSearch, VDepth, and FDepth can largely be attributed to the greedy heuristic that is employed in generating initial solutions for the local process in these approaches when producing their initial results. Given that the method employed for generating initial solutions was originally designed for the minimum DS problem, limitations arise in LSearch, VDepth, and FDepth approaches as they strive to enhance these initial solutions. Specifically, these methods aim to improve their solutions by swapping nodes belonging to different disjoint DSs. However, the number of solutions returned by the greedy heuristic can only be manipulated to a certain extent.

Table 1 shows the times of improvement of EADDSA over different algorithms using different number of nodes and different network densities.

V. CONCLUSION

This paper introduces a novel solution aimed at enhancing the HWSN lifetime performance. The proposed EADDSA approach incorporates the consideration of nodes with heterogeneous energy in the WSN and introduces an attentive method (EAA) that takes into account the energy heterogeneity during the construction of DSs. Moreover, the effective DS scheduling strategy enhances the HWSN lifetime by establishing operational guidelines for each round, taking into account the estimated lifetimes and the designated number of working rounds for each DS. A comparative analysis conducted with related algorithms, including MDP, MFMDP, LSearch, VDepth, and FDepth, reveals promising results. The findings demonstrate that the proposed EADDSA method outperforms these algorithms, leading to significant improvements in WSN lifetime. Specifically, EADDSA achieves superior average lifetime improvements compared to MDP, MFMDP, LSearch, VDepth, and FDepth, as indicated by improvement factors of 1.79, 4.15, 0.48, 0.21, and 0.30 times with 50 nodes, 3.55, 5.82, 1.90, 1.74, and 1.82 times with 100 nodes, 2.78, 4.72, 1.55, 1.35, and 1.43 times with 150 nodes, and 4.38, 7.16, 2.45, 2.27, and 2.34 times with 200 nodes, respectively. In the future, we plan to introduce and evaluate a novel hybrid optimization algorithm to further improve network lifetime through the formation of disjoint DSs that are sensitive to both energy and load balancing considerations.

ACKNOWLEDGMENT

The authors gratefully acknowledge Qassim University, represented by the Deanship of Scientific Research, on the financial support for this research under the number (2023-SDG-1-BSRC35303) during the academic year 1445 AH/2023 AD.

REFERENCES

[1] M. A. Ramírez-Moreno, S. Keshtkar, D. A. Padilla-Reyes, E. Ramos-López, M. García-Martínez, M. C. Hernández-Luna, A. E. Mogro, J. Mahlknecht, J. I. Huertas, R. E. Peimbert-García, R. A. Ramírez-Mendoza, A. M. Mangini, M. Roccotelli, B. L. Pérez-Henríquez, S. C. Mukhopadhyay, and J. de Jesús Lozoya-Santos, "Sensors for sustainable smart cities: A review," *Appl. Sci.*, vol. 11, no. 17, p. 8198, 2021.

[2] M. V. Ramesh, R. Prabha, H. Thirugnanam, A. R. Devidas, D. Raj, S. Anand, and R. K. Pathinarupothi, "Achieving sustainability through smart city applications: Protocols, systems and solutions using IoT and wireless sensor network," *CSI Trans. ICT*, vol. 8, no. 2, pp. 213–230, Jun. 2020.

[3] A. Aziz, K. Singh, W. Osamy, and A. M. Khedr, "Effective algorithm for optimizing compressive sensing in IoT and periodic monitoring applications," *J. Netw. Comput. Appl.*, vol. 126, pp. 12–28, Jan. 2019.

[4] A. Aziz, W. Osamy, A. M. Khedr, A. A. El-Sawy, and K. Singh, "Grey Wolf based compressive sensing scheme for data gathering in IoT based heterogeneous WSNs," *Wireless Netw.*, vol. 6, pp. 3395–3418, Feb. 2020, doi: 10.1007/s11276-020-02265-8.

[5] W. Osamy, A. A. El-Sawy, and A. M. Khedr, "Effective TDMA scheduling for tree-based data collection using genetic algorithm in wireless sensor networks," *Peer-to-Peer Netw. Appl.*, vol. 13, pp. 796–815, May 2020.

[6] A. M. Khedr, Z. Al Aghbari, and P. P. V. Raj, "MSSPP: Modified sparrow search algorithm based mobile sink path planning for WSNs," *Neural Comput. Appl.*, vol. 35, no. 2, pp. 1363–1378, Jan. 2023.

[7] P. V. Pravija Raj, Z. Al Aghbari, and A. M. Khedr, "ETP-CED: Efficient trajectory planning method for coverage enhanced data collection in WSN," *Wireless Netw.*, vol. 29, no. 5, pp. 2127–2142, Jul. 2023.

[8] C. Mahapatra, Z. Sheng, P. Kamalinejad, V. C. M. Leung, and S. Mirabbasi, "Optimal power control in green wireless sensor networks with wireless energy harvesting, wake-up radio and transmission control," *IEEE Access*, vol. 5, pp. 501–518, 2017.

[9] Z. Zhang, J. Zhou, Y. Mo, and D.-Z. Du, "Performance-guaranteed approximation algorithm for fault-tolerant connected dominating set in wireless networks," in *Proc. IEEE INFOCOM*, Apr. 2016, pp. 1–8.

[10] V. Padmapriya and D. N. Sujatha, "Future of sustainable smart cities: An insight," in *Blockchain for Smart Cities*. Amsterdam, The Netherlands: Elsevier, 2021, pp. 17–34.

[11] T. Pino, S. Choudhury, and F. Al-Turjman, "Dominating set algorithms for wireless sensor networks survivability," *IEEE Access*, vol. 6, pp. 17527–17532, 2018.

[12] A. H. Karbasi and R. E. Atani, "Application of dominating sets in wireless sensor networks," *Int. J. Secur. Appl.*, vol. 7, no. 4, pp. 185–202, 2013.

[13] X. Kui, S. Zhang, J. Wang, and J. Cao, "An energy-balanced clustering protocol based on dominating set for data gathering in wireless sensor networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Ottawa, ON, Canada, Jun. 2012, pp. 193–197, doi: 10.1109/ICC.2012.6363775.

[14] M. Bezoui, A. Bounceur, R. Euler, F. Lalem, and L. Abdelkader, "A new algorithm for finding a dominating set in wireless sensor and IoT networks based on the wait-before-starting concept," in *Proc. IEEE SENSORS*, Oct. 2017, pp. 1–3.

[15] F. A. Onat and I. Stojmenovic, "Generating random graphs for wireless actuator networks," in *Proc. IEEE Int. Symp. a World Wireless, Mobile Multimedia Netw.*, Jun. 2007, pp. 1–12.
[16] S. Guha and S. Khuller, "Approximation algorithms for connected dominating sets," *Algorithmica*, vol. 20, no. 4, pp. 374–387, Apr. 1998.
[17] K. Islam, S. G. Akl, and H. Meijer, "Maximizing the lifetime of wireless sensor networks through domatic partition," in *Proc. IEEE 34th Conf. Local Comput. Netw.*, Oct. 2009, pp. 436–442.
[18] I. Stojmenovic, M. Seddigh, and J. Zunic, "Dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 13, no. 1, pp. 14–25, Jan. 2002.
[19] J. P. Mohanty, C. Mandal, C. Reade, and A. Das, "Construction of minimum connected dominating set in wireless sensor networks using pseudo dominating set," *Ad Hoc Netw.*, vol. 42, pp. 61–73, May 2016.
[20] K. Islam, S. G. Akl, and H. Meijer, "A constant factor localized algorithm for computing connected dominating sets in wireless sensor networks," in *Proc. 14th IEEE Int. Conf. Parallel Distrib. Syst.*, Dec. 2008, pp. 559–566.
[21] J. Wu, F. Dai, M. Gao, and I. Stojmenovic, "On calculating power-aware connected dominating sets for efficient routing in ad hoc wireless networks," *J. Commun. Netw.*, vol. 4, no. 1, pp. 59–70, Mar. 2002.
[22] J. Yu, Q. Zhang, D. Yu, C. Chen, and G. Wang, "Domatic partition in homogeneous wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 37, pp. 186–193, Jan. 2014, doi: 10.1016/j.jnca.2013.02.025.
[23] M. Wollschlaeger, T. Sauter, and J. Jasperneite, "The future of industrial communication: Automation networks in the era of the Internet of Things and Industry 4.0," *IEEE Ind. Electron. Mag.*, vol. 11, no. 1, pp. 17–27, Mar. 2017.
[24] A. Aziz, K. Singh, W. Osamy, and A. M. Khedr, "An efficient compressive sensing routing scheme for Internet of Things based wireless sensor networks," *Wireless Pers. Commun.*, vol. 114, no. 3, pp. 1905–1925, Oct. 2020, doi: 10.1007/s11277-020-07454-4.
[25] R. Wan, N. Xiong, and N. T. Loc, "An energy-efficient sleep scheduling mechanism with similarity measure for wireless sensor networks," *Human-Centric Comput. Inf. Sci.*, vol. 8, no. 1, pp. 1–22, Jun. 2018.
[26] L. Ma, A. Khreishah, Y. Zhang, and M. Yan, *Wireless Algorithms, Systems, and Applications*, vol. 10251. Cham, Switzerland: Springer, 2017.
[27] J. Wu and H. Li, "On calculating connected dominating set for efficient routing in ad hoc wireless networks," in *Proc. 3rd Int. workshop Discrete Algorithms Methods Mobile Comput. Commun.*, Aug. 1999, pp. 7–14.
[28] B. Das, R. Sivakumar, and V. Bharghavan, "Routing in ad hoc networks using a spine," in *Proc. 6th Int. Conf. Comput. Commun. Netw.*, Sep. 1997, pp. 34–39.
[29] J. Yu, N. Wang, G. Wang, and D. Yu, "Connected dominating sets in wireless ad hoc and sensor networks—A comprehensive survey," *Comput. Commun.*, vol. 36, no. 2, pp. 121–134, 2013.
[30] T. Shi, S. Cheng, Z. Cai, and J. Li, "Adaptive connected dominating set discovering algorithm in energy-harvest sensor networks," in *Proc. 35th Annu. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Apr. 2016, pp. 1–9.
[31] B. W. Kernighan and S. Lin, "An efficient heuristic procedure for partitioning graphs," *Bell Syst. Tech. J.*, vol. 49, no. 2, pp. 291–307, Feb. 1970.

BADER ALWASEL received the B.S. degree in computer science from King Saud University, Qassim Branch, Saudi Arabia, the M.S. degree in computer security from the University of Birmingham, Birmingham, U.K., and the Ph.D. degree from Royal Holloway, University of London, London, U.K. Since 2022, he has been an Associate Professor with the Applied Natural Science Department. He is currently the Vice Dean for Applied Programs with the Applied College, Qassim University. His main research interests include cyber-physical systems security, network and distributed systems security, control systems, graph theory, and models for critical infrastructure protection.

AHMED SALIM received the B.Sc. degree in computer science and the M.Sc. degree in the area of distributed computing from Zagazig University, Egypt, in June 2001 and December 2006, respectively, and the Ph.D. degree in systems, network and telecommunication devices from the Bonch-Bruvich University of Telecommunication, Saint Petersburg, Russia, in October 2010. From January 2012 to December 2012, he was a Consultant of ICTP with Zagazig University, where he was an Assistant Professor with the Department of Mathematics, Faculty of Science, from January 2011 to May 2019. Since June 2019, he has been an Associate Professor with the Department of Mathematics, Faculty of Science, Zagazig University. Since January 2014, he has been an Assistant Professor with the Department of Mathematics, Faculty of Science and Art, Qassim University, Al-Mithnab, Saudi Arabia. His research interests include decomposable algorithms, computing, and wireless networks.

AHMED M. KHEDR received the B.Sc. degree in mathematics and the M.Sc. degree in the area of optimal control from Zagazig University, Egypt, in June 1989 and July 1995, respectively, and the M.Sc. and Ph.D. degrees in computer science and engineering from the University of Cincinnati, OH, USA, in July 1999 and March 2003, respectively. From March 2003 to January 2004, he was a Research Assistant Professor with the ECECS Department, University of Cincinnati. From January 2004 to May 2009, he was an Assistant Professor with Zagazig University. From September 2009 to September 2010, he was an Associate Professor with the Department of Computer Science, College of Computers and Information Systems, Taif University, Saudi Arabia. Since December 2014, he has been a Professor with Zagazig University. From September 2010 to December 2019, he was an Associate Professor, and since January 2020, he has been a Professor with the Department of Computer Science, College of Computing and Informatics, University of Sharjah, United Arab Emirates. He was awarded the State Prize of Distinction in Advanced Technology, Sharjah Islamic Bank Prize of Distinction in Research, and the University of Sharjah Prize of Distinction in Research (two times), in June 2009, May 2013, April 2014, and April 2021, respectively. His research interests include computational intelligence, wireless sensor networks, the Internet of Things, and distributing computing.

WALID OSAMY (Member, IEEE) received the B.Sc. degree (Hons.) in computer science and the M.S. and Ph.D. degrees in the area of computer science from the Faculty of Science, Zagazig University, Egypt, in June 2000, June 2006, and September 2010, respectively. From July 2001 to March 2009, he was involved in the projects of network infrastructure, and management information systems. He is a member of ICTP Projects with Zagazig University. From December 2010 to January 2019, he was an Assistant Professor with the Department of Computer Science, College of Computers and Informatics, Benha University, Egypt, where he has been an Associate Professor with the Department of Computer Science, College of Computers and Artificial Intelligence, since February 2019. Since March 2015, he has been an Assistant Professor with Qassim University, Buridah, Saudi Arabia. His research interests include computational intelligence and the IoT [mobile computing and wireless sensor networks (WSNs)].