HAL
open science

An Energy-Efficient Multiobjective Scheduling Model for Monitoring in Internet of Things

Basma Mostafa Hassan, Abderrahim Benslimane, Mohamed Saleh, Sally Kassem, Miklós Molnár

To cite this version:

Basma Mostafa Hassan, Abderrahim Benslimane, Mohamed Saleh, Sally Kassem, Miklós Molnár. An Energy-Efficient Multiobjective Scheduling Model for Monitoring in Internet of Things. IEEE Internet of Things Journal, 2018, 5 (3), pp.1727-1738. 10.1109/JIOT.2018.2792326 . lirmm-01818980

HAL Id: lirmm-01818980

https://hal-lirmm.ccsd.cnrs.fr/lirmm-01818980v1

Submitted on 14 Apr 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An Energy-Efficient Multiobjective Scheduling Model for Monitoring in Internet of Things

Basma Mostafa, Abderrahim Benslimane, *Senior Member, IEEE*,
Mohamed Saleh, Sally Kassem, and Miklos Molnar

***Abstract*—To ensure robustness in wireless networks, monitoring the network state, performance and functioning of the nodes and links is crucial, especially for critical applications. This paper targets Internet of Things (IoT) networks. In the IoT, devices (things) are vulnerable due to security risks from the Internet. Moreover, they are resource-constrained and connected via lossy links. This paper addresses the optimized scheduling of the monitoring role between the embedded devices in IoT networks. The objective is to minimize energy consumption and communication overhead of monitoring, for each node. Several subsets of the potential monitoring nodes are generated by solving a minimal vertex cover (VC) problem with constraint generation. Assuming periodical functioning, VCs are optimally assigned to time periods in order to distribute the monitoring role throughout the entire network. The assignment of VCs to periods is modeled as a multiobjective generalized assignment problem. To further optimize the energy consumption of the monitors, they are sequenced across time periods to minimize the state transitions of nodes. This part of the problem is modeled as a traveling salesman path problem. The proposed model is tested on randomly generated instances and the experimental results illustrate its effectiveness to optimize the scheduled monitoring for fault tolerance in IoT networks.**

***Index Terms*—Energy-efficient monitoring, generalized assignment problem (GAP), Internet of Things (IoT), robustness, scheduling, traveling salesman path (TSP) problem, vertex cover (VC).**

I. Introduction and Motivation

THE Internet of Things (IoT) is a persistently growing network that seamlessly interconnects a tremendous number of heterogeneous, smart devices (things) with the Internet. The connection does not require human-to-human or human-to-computer interaction. The IoT adopts novel processing and communication architectures and technologies [2]. IoT systems can rely on wireless sensor networks (WSNs) for data collection and their computational capabilities may be enhanced by employing cloud and fog/edge computing [3], [4]. As a result, numerous applications and services have been created; including smart cities, smart homes, smart grids, smart energy, smart agriculture, and environmental and health care monitoring [5].

The availability of devices, the reliability of communication, the quality of service (QoS), and security are all essential for the utilization of the IoT. Over time, the state of devices and the overall network may depreciate. This is due to the challenging and failure-prone nature of the IoT; consisting of a huge number of heterogeneous and resource-constrained things in terms of memory, energy, and computational capability. Furthermore, energy constraints impose hard duty cycles to maximise longevity, which in turn causes unreliable connectivity [6]. In addition to unknown and dynamic network topologies, and unreliable connectivity, this leads to incomplete information about the current network state [39]. The situation is considered a form of entropy, where a system deteriorates unless effort is invested in the development of monitoring and correction mechanisms to maintain a fault-tolerant system's performance [7].

Fault tolerance can be tackled in the network deployment stage by guaranteeing full network coverage and connectivity [8]. In the network layer, multipath routing has a major role in fault tolerance. It utilizes the high density in node deployment to incorporate redundancy in routing paths; which in turn increases the chance of data delivery [9], [10]. Nevertheless, constant network monitoring and taking corrective measures in case of failures is still necessary.

Consequently, the proliferation of IoT applications critically depends on effective monitoring methods and algorithms. To be effective, such methods and algorithms should do the following.

1) Ensure robustness and fault-tolerant operation of IoT ecosystems from end to end.
2) Guarantee good operation of sensors.
3) Verify the correctness of the big data programming models and the availability of resources in the network, in addition to.
4) Verify fog/edge and cloud datacenters, despite the high risk of faulty and nondeterministic environmental conditions.

Manuscript received September 14, 2017; revised December 14, 2017; accepted December 27, 2017. Date of publication January 11, 2018; date of current version June 8, 2018. This work was supported in part by the Academy of Scientific Research and Technology and in part by the French Ministry of Foreign Affairs for Granting the Imhotep Project MONIOT under Grant 33124VJ. *(Corresponding author: Abderrahim Benslimane.)*

B. Mostafa is with LIRMM, University of Montpellier, 34090 Montpellier, France, with the Faculty of Computers and Information, Cairo University, Cairo 12411, Egypt, and also with the CRI, French University in Egypt, El Shorouk, Egypt (e-mail: basma.mostafa@lirmm.fr).

A. Benslimane is with LIA, University of Avignon, 84911 Avignon, France (e-mail: abderrahim.benslimane@univ-avignon.fr).

M. Saleh is with FCI, Cairo University, Cairo 12411, Egypt (e-mail: m.saleh@fci-cu.edu.eg).

S. Kassem is with Nile University, Giza, Egypt, and also with the Faculty of Computers and Information, Cairo University, Cairo 12411, Egypt (e-mail: s.kassem@fci-cu.edu.eg).

M. Molnar is with LIRMM, University of Montpellier, 34090 Montpellier, France (e-mail: miklos.molnar@lirmm.fr).

Digital Object Identifier 10.1109/JIOT.2018.2792326

Thus, addressing monitoring for fault tolerance and prevention in complex IoT can significantly improve the functioning of the network and will eventually pave way for the wide deployment of IoT services. This problem is interesting for both research and industrial domains, particularly for mission-critical, time-sensitive applications. Examples of such applications are critical control and fault detection, health monitoring and safety applications. Such applications require the availability, reliability, and the state of communicant objects to be constantly checked for fast restoration in cases of unexpected communication problems.

The literature is rich with researches on monitoring WSNs [8], [11]–[14], but fewer researches tackled the problem of monitoring IoT networks. Obviously the large foundation built for WSNs paves the way for successful monitoring in the IoT. However, due to the intrinsic differences between the two paradigms, providing a monitoring system customized specifically for IoT networks and their standardized protocols is worth investigating. Therefore, this paper is motivated by the lack of research that has addressed monitoring IoT efficiently, in terms of energy consumption and communication overhead.

The monitoring system operations allow continuous maintenance of the network state and the availability of components. The system needs to consider the resulting overhead on the network and minimize it. The ability to monitor a network in real-time with energy-efficient and dynamic algorithms helps in the early detection/prevention of faults. At the same time, the network is left unconstrained during normal operation. This objective is challenging, since network monitoring causes high energy losses due to the need for devices to be awake most of the time to maintain a correct network state [15]. On the other hand, the design of WSNs, and low power and lossy networks (LLNs), is ordinarily constrained by life span concerns. A prevalent approach toward expanding the life span of such networks is by using sleep scheduling. In sleep scheduling, nodes enter sleep state frequently, and intermittently wake up to check for action in the network [16], [17]. For critical and time-sensitive IoT applications, monitoring needs to be continuous regardless of the lack of activity in the network. Therefore, while employing monitoring with sleep scheduling, it is required to guarantee that each link in the network is always monitored by at least one monitoring node throughout the network lifetime. Ignoring the monitoring overhead may lead to increased congestion in the network at critical times, or battery drainage. These situations violate the requirements of mission-critical, time-sensitive IoT applications.

This paper investigates the optimized scheduling of the monitoring role of nodes in IoT networks. The problem is represented through a mathematical model. The developed model corresponds to a multiobjective optimization of the energy consumption and the overall communication overhead of monitoring the network. The proposed model is decomposed into a three-phase framework. The first phase requires generating multiple subsets of the nodes that cover the entire graph. These subsets are generated by solving a vertex cover problem (VCP) iteratively; while incorporating a constraint generation algorithm. Furthermore, an optimized scheduling of the vertex covers (VCs) is proposed. The scheduling aims to minimize the energy and communication costs incurred while monitoring the network. QoS-aware monitoring is targeted, where constraints are used to include/eliminate the nodes that can/cannot withstand the energy requirements of monitoring. The scheduling is addressed in the second and third phases of the computation, which are modeled as a multiobjective generalized assignment problem (GAP) and a traveling salesman path (TSP) problem, respectively. To verify the effectiveness of the proposed model, a branch-and-bound (BB) algorithm is used for the first and second phases and a dynamic programming algorithm is used for the third phase. This contribution is a step toward achieving optimized monitoring for improved robustness and fault tolerance. The monitoring process is performed with minimal energy consumption and communication overhead while providing load balancing. This is important for mission-critical IoT applications.

The rest of this paper is organized as follows. Section II discusses literature related to network monitoring in the IoT and WSNs. Section III is a description of the problem statement. Section IV presents the modeling of the monitoring optimization. Section V presents the mathematical formulation of the proposed framework, followed by implementation and analysis in Sections VI and VII, respectively. The conclusions and future research are presented in Section VIII.

II. Related Work

Several applications of the IoT are known for area, object, and health monitoring [18]–[20]. This research targets monitoring the IoT network itself (hereinafter referred to as network monitoring). Algorithms are developed for ensuring a reliable and fault-tolerant operation of the network. Fault tolerance can be addressed in each of the network (routing) layer, transport layer or application layer [8]. Focusing on the network layer, the objective is to provide stable, reliable, and scalable end-to-end connectivity.

Extensive work had been proposed in the literature to address the problem of network monitoring for fault tolerance in WSNs [11]–[14], [21]. Swain *et al.* [11] proposed a clustering framework for fault diagnosis to balance the load and minimize the energy consumption. The authors used a statistical mechanism for fault detection and neural networks for fault classification. Panda and Khilar [12] proposed a statistical-based, distributed fault diagnosis algorithm. Coordination between neighboring nodes was used in the fault diagnosis to minimize the communication overhead. Khan *et al.* [13], on the other hand, proposed a fault detection strategy for WSNs based on a fuzzy inference system.

Most of the proposed techniques are considered to be either active or passive monitoring [8], [22]. Active monitoring injects probes into the network to infer the network's performance from the measured parameters (such as delays and packet losses). Alternatively, passive monitoring observes the traffic that normally passes through the network and infers the network's performance from it. Examples of active and passive monitoring for WSNs are found in [14] and [21], respectively.

Given that the design of WSN, and LLNs is always constrained by life span concerns, prolonging the lifetime of the nodes has become a problem of paramount importance. The problem is called in the literature the maximum lifetime problem (MLP). Cardei *et al.* [23] proved that MLP is NP-complete. Since monitoring uses the energy of the nodes, this additional activity should be minimized to prolong the network lifetime as much as possible.

Lifetime maximization can be reached by using the high redundancy feature in node deployment, specifically, by collecting the nodes into subsets known as covers. Each target must be monitored by at least one sensor in the cover. Extending the network lifetime can be achieved by scheduling between these covers, where the monitoring role is alternated between different subsets. The sensors that are not currently monitoring switch to a sleep state, where the energy consumption is minimal. The total energy consumption of a node must be at most equal to its initial energy. Several propositions were formulated in the literature to address MLP and problem variations [5], [17], [24], [25].

Although the literature is rich with many researches on monitoring in WSNs [8], [11]–[14], [48], most of the researches proposed for WSNs are based on the assumptions that the network does not have a central controlling node, and that nodes are not associated with a unique global address [26]. On the other hand, IoT networks always include a central control device, known as the border router (BR), which connects IPv6 over LoWPANs (6LoWPAN) networks with the Internet. The BR is assumed to be always accessible [29], [30], therefore it is capable of performing a crucial role in monitoring. Moreover, nodes are globally identified by an IP address which facilitates active monitoring when checking the availability of networks' components, where communication has not been established for long times. The full integration with the Internet infrastructure and cloud services further enhances the computation capabilities of IoT networks, which could also be beneficial for monitoring.

Despite these favorable characteristics, ensuring a reliable, fault-tolerant communication for IoT networks remains a challenging task for the following reasons.

1) The IoT is enabled by resource-constrained and heterogeneous things that are connected to the insecure Internet.
2) The utilized protocols need to be optimized to minimize the overhead on the constrained things.
3) Things are globally accessible which imposes a huge security risk from the Internet.
4) The communication is unreliable as they are connected through lossy links [26].
5) Furthermore, there are challenges imposed due to node mobility and the dynamic and uncertain environmental conditions.

To enable the IoT, the Internet Engineering Task Force (IETF) working group has defined IPv6 over LoWPANs [29]. It was promised to provide end-to-end connectivity, ubiquity, and scalability. However, due to the resource-constrained nature of the nodes, crucial IP protocols that are usually incorporated for network monitoring cannot be supported by such nodes. Therefore, extra effort has to be spent to address optimized network monitoring for IoT networks. Several recent works were proposed to answer to the security requirements of the IoT [26]–[28]. For instance, Raza *et al.* [26] developed an intrusion detection system to catch routing attacks in the IoT. Giuliano *et al.* [27] proposed a key renewal algorithm for the secure access of devices.

Additionally, The IETF working group standardized the routing protocol for LLNs (RPL). It is considered to be the de facto routing protocol for IP-connected IoT [29], [30]. It was developed as a response to the routing requirements of LLNs [31]. Multipath routing as well as local and global routing structure repair mechanisms were proposed to handle unreliable connectivity in IP-connected IoT. Multipath routing utilizes the high density in node deployment to incorporate redundancy in routing paths. It contributes to fault tolerance as it facilitates the recovery of the routing structure in case of link failures which increases the chance of data delivery and provides load balancing [8]–[10].

RPL repair mechanisms are considered to be reactive to node or link failures. Consequently, delays in communications and nodes unreachability are expected for some time (possibly for several minutes) [32]. Such delays are unacceptable for critical, time-sensitive IoT applications. Thus, network monitoring is a requirement. It is imperative, and yet challenging, to develop monitoring systems compatible with these standards and adapted to the mentioned challenges and characteristics. This led to development of a monitor placement algorithm that works in tandem with RPL in [1]. The algorithm is modeled as a classical minimum VCP that works in tandem with IPv6 routing protocol for LLNs (RPL) The VCP works on the destination oriented directed acyclic graphs (DODAG) constructed by RPL. A polynomial-time algorithm that converts the DODAG into a nice-tree decomposition with unity treewidth was developed in [1]. This strategy yielded a significant reduction in the complexity of solving the originally NP-hard VCP on generic graphs to be only polynomial-time solvable on DODAGs. In this paper, an energy-efficient monitoring algorithm is proposed to maintain reliable network structure, while minimizing the overhead monitoring imposes on the network.

III. Problem Statement

The problem addressed is the efficient, full monitor coverage in RPL-based 6LoWPAN networks for a predetermined lifetime, considering minimum energy consumption and communication overhead. In RPL, the BR is responsible for the DODAG construction, which starts by broadcasting a DODAG information object (DIO) [30]. The DIO contains the configuration of the DODAG. The objective of this paper is to continuously test the availability and correct functioning of neighboring nodes by monitoring the status of the entire set of links present in the network.

Monitoring can be performed inside RPL by including a header in the DIO for metrics and constraints objects. This header is called DAG metric container object. Node energy and link reliability are examples of the metrics/constraints that

could be included in the DAG metric container object [33]. When the DIO traverses the DAG, each node augments a sub-object to the message, which expresses its value to the metric used. The monitored nodes will unicast their responses of the DIOs to their monitors. Monitors are the nodes responsible for gathering the monitoring data and reporting back to the BR, through multihop communication. The BR then sends this information to the networks operations center (NOC), where data analysis is performed.

It is assumed that a link can only be monitored by its extremities (i.e., endpoints). Also, each node has several activities independent from monitoring; therefore, the monitoring activity cannot consume more than a specified limited power (fraction of a node's battery). The intrinsic energy limitations of the resource-constrained things in IoT networks need to be taken into consideration. This is because a monitor cannot continuously monitor its neighbors without running out of the battery reserved for monitoring.

As mentioned in Section I, sleep scheduling is usually incorporated in WSNs and LLNs to reduce average power consumption and maximize longevity. This is often achieved by duty cycling between active and sleep states of the nodes. Hence, in order to achieve continuous monitoring, it is imperative to alternate the monitoring duty between several sets of nodes, while ensuring that each set is capable of covering the entire network. It should be noted that the active/sleep alternation addressed in this context is the turn on/off of the monitoring activity of the node, regardless of the other activities a node may perform (sensing, transmitting of sensed data, etc.). To realize this alternation, the monitoring function is periodical across the planning horizon. The planning horizon is the predetermined lifetime specified by the NOC in which all edges are being monitored. The NOC is responsible for planning the optimized periodic monitor scheduling. Therefore, the optimization is external from the resource-constrained things. At any time interval, the active node set is known from the DODAG.

Usually, the cost of monitoring a network includes a monitor deployment cost and an operational cost [34]. The deployment cost is the hardware cost of deploying monitoring nodes. For the problem in hand, the operational cost is the network overhead resulting from the communications between the monitoring devices and the BR. The monitors are ultimately transmitting the gathered information about the monitored nodes to the root of the DODAG. The transmission is done through multihop communication. Consequently, the longer the path to the root in number of hops, the higher is the cost. Moreover, the monitoring activity consumes a percentage from the battery reserved for monitoring. This paper aims to minimize the monitoring costs in terms of communication and energy costs.

As mentioned above, in sleep scheduling, a node assumes different states: active state, sleep state, or transient state (from active to sleep and vice versa). The energy consumption during the transient state can be high [35], and numerous state transitions for the nodes consume extra energy [Fig. 1(a)]. Consequently, the energy consumption of the monitors could further be minimized if the time periods, where a node is actively monitoring are merged together [Fig. 1(b)]. To that end, it is required to find the optimal sequencing between the sets of monitors across the time periods in a way that minimizes the state transitions of nodes. The modeling and mathematical formulation of the monitoring optimization are discussed in the following section.

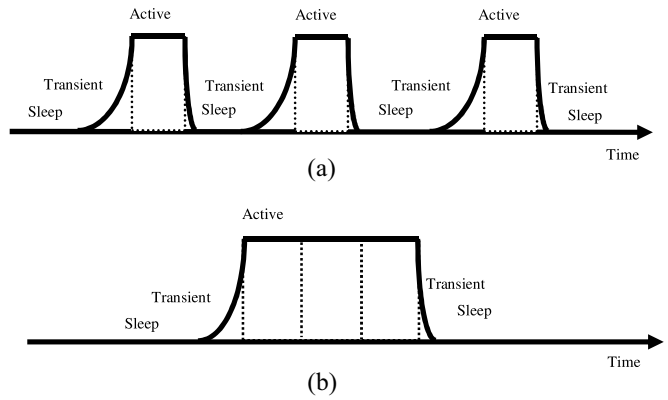

Fig. 1. Energy model. (a) Before active time periods merged and (b) After active time periods merged [10].

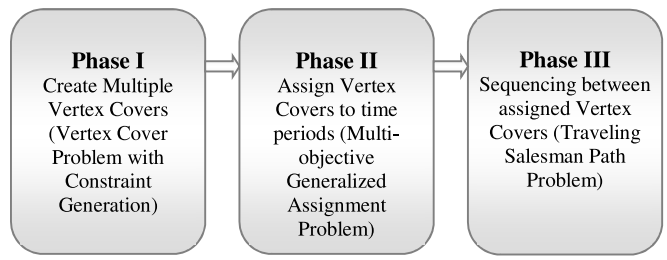

Fig. 2. Phases of the monitoring optimization.

IV. Modeling of Monitoring Optimization

The IoT network could be represented by its logical graph constructed by the routing protocol RPL, namely the DODAG [18], [19]. Consider an active DODAG $\boldsymbol{D} = (\boldsymbol{V}, \boldsymbol{E})$, where $\boldsymbol{V}$ represents all the vertices, $\boldsymbol{V} = \{\boldsymbol{v_k}, k = 1, 2, \ldots, q\}$ and $\boldsymbol{E}$ is the set of edges. For monitoring a time horizon $\boldsymbol{T} = \{T_j, j = 1, \ldots, n\}$, the duty is cycled between several sets of nodes; each node has a reserved battery for monitoring ($\boldsymbol{Reserved_battery_k}$). The monitoring activity for one period consumes energy ($\boldsymbol{E_m}$). During the lifetime of monitoring, communication and transition costs are incurred. The communication cost is the cost of transferring the monitoring data to the BR. Transition cost is the cost of alternating from one set of monitors to another. The objective is to minimize the overall monitoring costs while ensuring monitor coverage of the entire DODAG throughout the planning horizon.

Scheduling between several sets of monitors for minimum energy consumption is an NP-hard problem. This remains true even in the very special case, where, the $\boldsymbol{Reserved_battery_k}$ for monitoring is sufficiently large such that, one set of monitors can cover the entire planning horizon. Finding the minimum set of vertices to cover an entire graph is also NP-hard [36], [37].

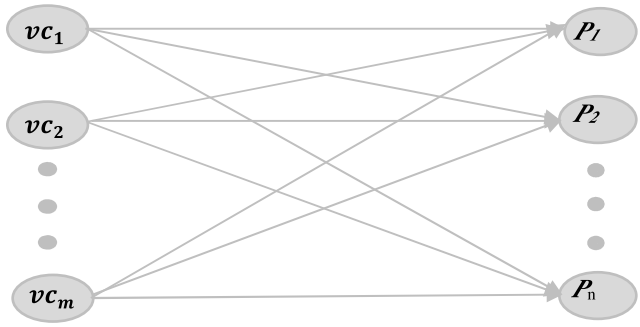

Fig. 3. Assignment of VCs i to periods j.

Model 1: Minimum VCP

Decision Variables

$$\textbf{Let } v_k = \begin{cases} \mathbf{1}, & \text{if } v_k \text{ is chosen in a Vertex Cover} \\ \mathbf{0}, & \text{if } v_k \text{ is not chosen in a Vertex Cover} \end{cases} \tag{2}$$

Model Equations

$$\textbf{Minimize } \sum_{k=1}^{q} v_k \tag{3}$$

$$\textbf{Subject to } v_i + v_j \geq 1 \ \forall \ (v_i, v_j) \in \mathbf{E} \tag{4}$$

$$v_k \in \{0, 1\} \ \forall \ v_k \in \mathbf{V} \tag{5}$$

To the best of our knowledge, the exact solution to the defined problem is not known. In this paper, a three-phase decomposition of the problem (Fig. 2) is proposed. The three proposed phases are modeled using well-known optimization problems in the literature. Although these problems are NP-hard in nature, approximation algorithms to solve them are available in the literature (see [36]–[39]). Moreover, decomposing the original problem allows a finer reduction of the search space and, hence, reduction in the solution's complexity.

The first phase (Phase I) of the proposed decomposition is responsible for creating multiple sets of monitoring nodes. Each set satisfies the coverage requirements of the network. Phase I can be modeled as a VCP [40]. The objective is to find the minimal sets of vertices that cover all the edges in the graph. The graph used in this paper, is the DODAG constructed by RPL [29], [30]. Only minimal VC sets are interesting for monitoring. If a node set is not minimal for the VCP then it contains a subset which is minimal and can cover all the edges with less cost.

As described in Section III, several monitoring sets (VCs) are required to achieve monitor scheduling while minimizing and balancing the energy consumption of the monitors. For this purpose, an algorithm (Algorithm 1) is developed to get multiple solutions of the same VCP, after incrementally adding new constraints to reduce the search space. Reducing the search space through incrementally adding new constraints is known as the constraint generation approach [41].

The output of Phase I is several sets of VCs. The working schedule of the VCs across the planning horizon is still required. Given a planning horizon (defined by the NOC), divided into several time periods, the purpose of Phase II is to optimally assign (a subset of) the VCs to each of the time periods with the following objectives.

1) The total energy consumed for monitoring is minimized.
2) The communication cost across all VCs is minimized.

Phase II is modeled as a GAP [42]. GAP is a special type of optimization problems, where agents are assigned to perform multiple tasks (Fig. 3). For the problem in hand, agents are the VC sets (or a subset of them) that should be assigned to time periods.

The last phase, Phase III, is sequencing the VCs with the objective of minimizing nodes' state transitions from one period to the next. The sequence generated determines the number of times a node's state is being changed from asleep to awake and vice versa. Phase III is modeled as a TSP problem [43]. We are interested in finding the minimum weighted Hamiltonian path (HP) (in terms of nodes state transitions) from an arbitrary starting point. A HP is a path that visits each vertex exactly once without the need to return to the starting vertex [47]. The vertices in this case are the selected VCs from Phase II. The starting vertex of the HP is completely arbitrary. The edges represent the transition costs. The result (the path) gives the optimal sequence (scheduling) of the selected VCs.

V. Mathematical Formulation of Monitoring Optimization

The three phases include the following sets.

1) *Set* $\boldsymbol{V}$*:* Represents all the vertices in the DODAG, where $\boldsymbol{V} = \{\boldsymbol{v_k}, k = 1, 2, \ldots, q\}$.
2) *Set* $\boldsymbol{S}$*:* Represents all the VCs $\boldsymbol{VC}$ obtain-ed from Phase I, where subset $\boldsymbol{VC} \subset \boldsymbol{V}$ represents the subset of monitoring nodes and $\boldsymbol{S} = \{\boldsymbol{vc_i}, i = 1, 2, \ldots, m\}$.
3) *Set* $\boldsymbol{T}$*:* Contains the disjoint time periods covering the entire planning horizon; which the VCs are assigned to monitor. $\boldsymbol{T} = \{T_j, j = 1, \ldots, n\}$.

A. Phase I: Generating Multiple Vertex Covers

Let $\boldsymbol{D} = (\boldsymbol{V}, \boldsymbol{E})$ be the DODAG, where $\boldsymbol{V}$ is the set of vertices and $\boldsymbol{E}$ is the set of edges. A subset $\boldsymbol{VC} \subset \boldsymbol{V}$ is a minimal VC of $\boldsymbol{D}$ if for every edge $(u, v) \in \boldsymbol{E}$, either $u \in \boldsymbol{VC}$ or $v \in \boldsymbol{VC}$ or both $u, v \in \boldsymbol{VC}$ and $\boldsymbol{VC}$ is irreducible, i.e., no vertex can be removed from the VC without losing the coverage property.

Phase I computes set $\boldsymbol{S}$ which includes the $\boldsymbol{m}$ VCs such that for each edge in the DODAG at least one of its endpoints belongs to $\boldsymbol{vc_i}$. Moreover, no VC in $\boldsymbol{S}$ should be a subset of another such that

$$\forall \ \boldsymbol{vc_i} \subset \boldsymbol{S}, \ \nexists \ \boldsymbol{vc_j} \subset \boldsymbol{S} | \boldsymbol{vc_i} \subset \boldsymbol{vc_j}. \tag{1}$$

Let the decision variables $\boldsymbol{v_k}$ express whether the vertex $\boldsymbol{v_k}$ is in the VC or not. The objective is to minimize the total number of vertices in the VC, subject to the constraint that at least one vertex of the edge $(\boldsymbol{v_i}, \boldsymbol{v_j})$ is a member of the $\boldsymbol{VC}$. The problem is the binary optimization problem represented by Model 1.

Algorithm 1: VCP With Constraint Generation

Input:
DODAG $\boldsymbol{D}$,
Initial solution $\boldsymbol{vc_i}$, initial matrix of constraints $\boldsymbol{A}$,
vector of RHS b, and the initial value of the
objective function z.
Output:
Matrix of solutions $\boldsymbol{M}$
Step 1 **While** (feasible minimal $\boldsymbol{VC}$ do exist) **do**
Step 1.1 $\boldsymbol{M} \leftarrow \boldsymbol{vc_i}$
Step 1.2 $\boldsymbol{A} \leftarrow$ A $+\boldsymbol{vc_i^T}$
Step 1.3 $b \leftarrow b + (z_i - 1)$
Step 1.4 i$\leftarrow$i +1
Step 1.5 $(\boldsymbol{vc_i}, z) =$ VCP ($\boldsymbol{A}$, b)
END While
Step 2 **Return** $\boldsymbol{M}$
END

The binary integer program in Model 1 is solved to get the minimum set of monitors, i.e., the minimum VC. In order to obtain multiple VCs, Algorithm 1 is used to solve the VCP iteratively. Each VC obtained using Algorithm 1 satisfies the constraints of Model 1.

Algorithm 1 works as follows: construct a matrix $\boldsymbol{M}$, where all solutions of the VCP are stored, and a z vector which contains the corresponding objective function value to each solution in $\boldsymbol{M}$. Initially, $\boldsymbol{M}$ contains the optimum solution obtained after solving Model 1 using BB algorithm, and z stores the minimum VC.

Then, the VCP is solved iteratively after adjusting the integer program by adding previous solutions to the set of constraints, thereby reducing the search space. The algorithm terminates when no other feasible minimal VCs can be found. In each iteration, a column representing the new solution, which has a cardinality greater than or equal to the previous solution, is appended to matrix $\boldsymbol{M}$ and the corresponding objective function value is added to the $\mathbf{z}$ vector, such that

$$\boldsymbol{vc_i^T} * \boldsymbol{M} < z. \tag{6}$$

To illustrate, let the initial solution be $\boldsymbol{vc}_1$, and the corresponding objective function value be z_1. To obtain $\boldsymbol{vc}_2$, solve the integer program $Ax \leq b$ after adding the following constraint to the matrix of constraints ($\boldsymbol{A}$) and the vector of RHS (b):

$$\boldsymbol{vc_2^T} * \boldsymbol{vc}_1 < z_1. \tag{7}$$

In this way, to solve for $\boldsymbol{vc}_3$ the following two constraints are appended:

$$\boldsymbol{vc_3^T} * \boldsymbol{vc}_1 < z_1 \tag{8}$$

$$\boldsymbol{vc_3^T} * \boldsymbol{vc}_2 < z_2. \tag{9}$$

To solve for $\boldsymbol{vc}_m$ the following ($\boldsymbol{m} - \mathbf{1}$) constraints are appended:

$$\boldsymbol{vc_m^T} * \boldsymbol{vc}_1 < z_1 \tag{10}$$

$$\vdots$$

$$\boldsymbol{vc_m^T} * \boldsymbol{vc}_{m-1} < z_{\boldsymbol{m}-1}. \tag{11}$$

B. Phase II: Assigning Time Periods to Vertex Covers

In Phase II, the planning horizon is divided into several periods. Then, a mathematical model is developed to optimally associate the VCs to periods throughout the planning horizon. The assignment does not define any ordering or sequencing of the time periods. Given a planning horizon denoted by $\boldsymbol{T} = \{\boldsymbol{T_j}, j = 1, \ldots, n\}$, define a binary decision variable $\boldsymbol{s_{ij}}$ that indicates whether a VC $\boldsymbol{vc_i}$ is assigned to monitor a period $\boldsymbol{T_j}$ [refer to (14)].

Assumptions:

1) Any node can be selected as monitor.
2) A node may be a member of several VCs.
3) Some VCs may monitor one period, more than one period, or none at all.
4) Each period is to be assigned exactly one VC.
5) The monitoring cost is defined in terms of the communication cost incurred by monitors and the energy loss due to monitoring.

Any node v_k may be a member of several VCs. The current $\boldsymbol{Consumed_energy_k}$ of node v_k is calculated using (15). Equation (15) states that: the current $\boldsymbol{Consumed_energy_k}$ for monitoring, depends on the number of times a VC set, including v_k, has been assigned to a period, multiplied by the energy loss ($\boldsymbol{E_m}$) per each monitoring period. The energy loss is the same for each active monitor and for each period. In (12), a binary variable $\boldsymbol{y}_{ki}$ is defined, which indicates whether a vertex $\boldsymbol{v_k}$ is a member of VC $\boldsymbol{vc_i}$ or not. The variable $\boldsymbol{y}_{ki}$ is the output of Phase I, and hence it is a parameter in Phase II. When $\boldsymbol{y}_{ki}$ is multiplied by $\boldsymbol{s_{ij}}$ and summed over the VCs and the periods, the result is the number of times a vertex $\boldsymbol{v_k}$ has been assigned to monitor a period [refer to (15)].

Energy is lost due to communication between the monitoring devices and the BR. This is because the monitors are ultimately transmitting the gathered information about the monitored nodes to the root of the DODAG. The energy lost for communicating the monitoring data is not part of the $\boldsymbol{Reserved_battery_k}$ for monitoring. On the other hand, this energy loss affects the rest of the battery that is not dedicated to monitoring. Accordingly, it is necessary to find the shortest path, in terms of the number of hops to the BR. The objectives in Phase II are twofold [refer to (16)–(18)]. At this stage, it is required to determine how all assignments should be made while minimizing the total number of hop counts traveled by all members of the VCs. This is done while minimizing the total energy spent for all nodes. Equation (12) denotes the number of hops traveled from each monitoring node $\boldsymbol{v_k}$ in $\boldsymbol{vc}_i$ to the root as $\boldsymbol{h_k}$, and (13) denotes the total number of hops traveled by all $\boldsymbol{v_k}$ in $\boldsymbol{vc}_i$ as $\boldsymbol{H_i}$. Constraint (19) indicates that each period must be monitored by one VC. Constraint (20) ensures that the energy consumed for monitoring never exceeds the ($\boldsymbol{Reserved_battery_k}$).

C. Phase III: Sequencing Between Assigned Vertex Covers

The objective of Phase III is to minimize the nodes' state transitions from one VC to the next. If a node $\boldsymbol{v_k}$ belongs to more than one VC set $\boldsymbol{vc_i}$, the model associates consecutive periods to the VC sets containing the repeated node.

Model 2: Genaralized Assignment Problem

Parameters:

$$y_{ki} = \begin{cases} 1, & \text{if } v_k \in vc_i \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

h_k Number of hops travelled from each v_k in vc_i to the root of the DODAG

H_i Total number of hops travelled by all v_k in vc_i

$$= \sum (\mathbf{y_{ki}} * h_k) \ \forall \ \mathbf{k} \in \mathbf{V} \tag{13}$$

E_m Energy loss per each monitoring period assigned for v_k

$Reserved_battery_k$ Maximum battery allowed for monitoring

Decision Variables:

$$\textbf{Let } s_{ij} = \begin{cases} 1, & \text{if } vc_i \text{ is assigned to period j} \\ 0, & \text{otherwise} \end{cases} \tag{14}$$

Auxiliary Variables:

$$Consumed_energy_k = E_m * \left(\sum_{i=1}^{m} \sum_{j=1}^{n} y_{ki} * s_{ij} \right) \tag{15}$$

Objective Functions:

$\mathbf{F_1}$**: Total energy loss**

$$F_1 = \sum_{k \in V} Consumed_energy_k \tag{16}$$

$\mathbf{F_2}$ **:Total number of hops across all periods**

$$\mathbf{F}_2 = \sum_{i=1}^{m} \sum_{j=1}^{n} (H_i) * s_{ij} \tag{17}$$

Model Equations:

$$\textbf{Minimize } \mathbf{F}_1 \ \& \ \mathbf{F}_2 \tag{18}$$

Subject to:

$$\sum_{i=1}^{m} s_{ij} = 1, \ \forall \ j \in \mathbf{T} \tag{19}$$

$$\textbf{Consumed_energy}_\mathbf{k} \leq Reserved_battery_k \ \forall \ k \in \mathbf{V} \tag{20}$$

$$s_{ij} \in \{0, 1\} \ \forall \ i, j \tag{21}$$

Model 3: TSP Problem

Parameters

$$\textbf{Let } C_{ij} = \sum state\ Transitions_{ij} \tag{22}$$

Decision Variables

$$\textbf{Let} \quad x_{ij} = \begin{cases} 1, & \text{if } vc_j \text{ is chosen for monitoring after } vc_i \\ 0, & \text{otherwise} \end{cases} \tag{23}$$

Model Equations

$$\textbf{Minimize} \ \sum_{i=0}^{n} \sum_{j \neq i, j=1}^{n} C_{ij} x_{ij} \tag{24}$$

Subject to

$$\sum_{i=0, i \neq j}^{n} x_{ij} = 1 \ \forall \ j \tag{25}$$

$$\sum_{j=0, i \neq j}^{n} x_{ij} = 1 \ \forall \ i \tag{26}$$

$$u_i - u_j + (n-1) x_{ij} \leq \mathbf{n} - 2$$
$$\mathbf{i}, \mathbf{j} = 2, \ldots, n, \ \mathbf{i} \neq j \tag{27}$$

$$1 \leq u_i \leq n - 1 \quad \mathbf{i} = 2, \ldots, \mathbf{n} \tag{28}$$

Accordingly, the number of times a node needs to start up to perform its assigned monitoring is minimized. This Phase is modeled as TSP path. Details of the stocktickerTSP path are shown in Model 3, where the Miller, Tucker, and Zemlin mathematical formulation [38] is adopted. The decision variables and parameters are adjusted in Model 3 such as follows.

1) The cities here are the unique VC sets assigned to periods $\boldsymbol{T} = \{T_j, j = 1, \ldots, n\}$, i.e., the unique results from the previous assignment model (Model 2).
2) Define a binary decision variable x_{ij} denoting whether vc_j is selected in the path after vc_i (23).
3) A feasible solution is a path (HP) that passes through each set exactly once (25) and (26).
4) Number of state transitions of the members of the sets, from active to sleep and vice versa (22).
5) The cost (distance) C_{ij} of moving from one city (VC set) to the next, from vc_i to vc_j, is the total.
6) the objective is to find the least costly sequence of VCs over the planning horizon (24).
7) Extra variables u_i are required for subtour elimination, in the constraints expressed in (27) and (28).

VI. Implementation and Analysis

A. Problem Resolution and Implementation

Solving the problem starts with Phase I by generating a DODAG and outputs matrix $\boldsymbol{M}$. Multiple solutions of the VCP are stored in matrix $\boldsymbol{M}$. Algorithm 1 runs until no more feasible minimal VCs exist. However, for experimentation purposes, it has been shown to suffice to loop any number of times between the range [1.5|$\boldsymbol{T}$|– 3|$\boldsymbol{T}$|]. This range denotes the required number of VCs (θ). Analytical simulation is conducted for networks of 50, 100, 150, and 200 nodes with varying numbers of links. Table I shows the characteristics of the corresponding DODAGs, a summary of the results of the three phases for eight different DODAGs, as well as the metrics used for evaluation. DODAGs are constructed such that there is a root (BR) and each node has at least one path toward it. The edges on the DODAGs are constructed by randomly generating (x, y) positions of each node in a unit square (units do not matter in the graph). The distance between every

TABLE I
SUMMARY OF EXPERIMENTAL RESULTS $|T| = 10$, $E_m = 2\%$, RESERVED_BATTERY$_k = 50\%$, INITIAL_BATTERY$_k = 100$

Instance	Nodes	Links	Threshold	%monitors	Avg. residual battery (%)	Running time (sec)			%Reduction in nodes' state transitions
						Phase I	Phase II	Phase III	
1	50	123	0.125	60	86.0	0.83	3	0.001	66.6
2	50	104	0.09	52	89.8	0.46	2	0.001	67.0
3	100	234	0.09	55	88.8	2.02	2	0.001	70.0
4	100	347	0.125	66	86.6	9.71	600	0.001	80.8
5	150	380	0.09	62	87.5	48.9	591	0.001	70.2
6	150	407	0.125	57	87.5	21.5	523	0.001	52.9
7	200	383	0.06	54	98.0	22.98	534	0.001	80.0
8	200	576	0.08	63	87.3	30.06	300	0.001	0.00

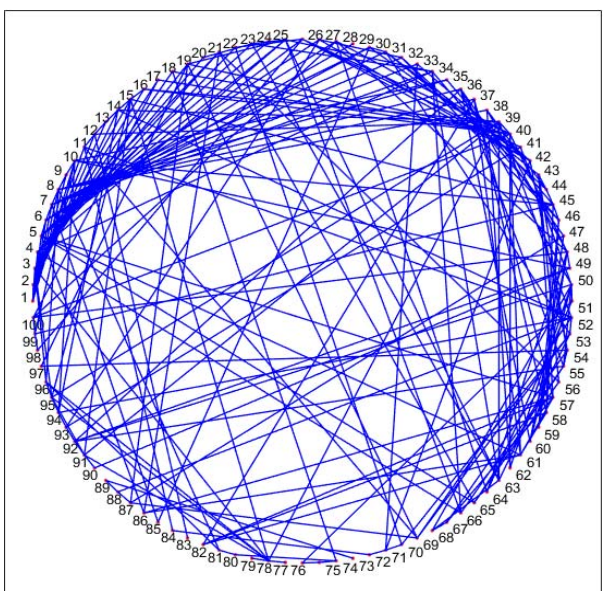

Fig. 4. DODAG of a network with 100 nodes and 234 links. Node number 1 is the root.

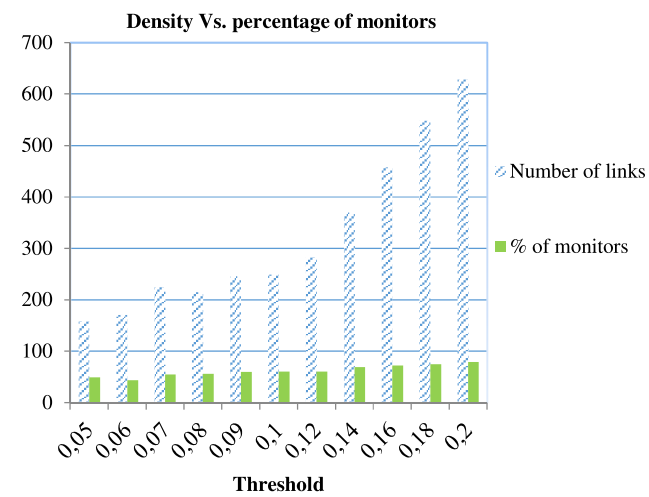

Fig. 5. Effect of varying the density of the DODAG on the percentage of monitors.

two nodes is measured; if it is less than a certain threshold parameter then the two nodes are connected.

By varying the threshold parameter, in the range [0 – 1], the DODAG gets sparser or denser. Fig. 4 shows an example of a DODAG with 100 nodes and a threshold of 0.09, which gives 234 links. Varying the density of the graph affects the number of nodes required to monitor the entire DODAG. Fig. 5 depicts the effect of varying the density of a DODAG, with 100 nodes, on the percentage of monitors. It is evident that the more the number of communication links between the nodes, the more monitors are required.

Throughout the entire set of experiments, it is assumed that the planning horizon $\boldsymbol{T}$ is divided into ten time periods, and that the energy loss $\boldsymbol{E_m}$ per monitoring period assigned for $\boldsymbol{v_k}$ is 2% of its total energy. Running Phase I with DODAG number 3 in Table I, for example, (DODAG shown in Fig. 4), outputs matrix $\boldsymbol{M}$ which includes 15 different VCs ($\boldsymbol{VC}_1, \ldots, \boldsymbol{VC}_{15}$). $\boldsymbol{M}$ is the input of the multiobjective GAP in Phase II.

As mentioned in Section III, the model in Phase II a multiobjective mathematical programming (MMP) problem. There are several approaches to solving MMP problems in the literature. This research adopts the ε-constraint method. This is due to its several advantages over its rivals [44]. In the ε-constraint approach, only one objective function is optimized, whereas the others are added to the constraints. Pareto-efficient solutions are achieved by varying the right hand side of the constrained objective functions.

Although it is widely used, the ε-constraint method has its disadvantages. First of all, it is required to calculate the range of every objective function used as a constraint. The usual way is to build a payoff table. It includes the optimal solution for each objective function optimized individually. There is no guarantee that these optimal solutions are indeed nondominated solutions [45]. Another weak point is that the optimal solution is not guaranteed to be an efficient solution if there are alternative optima. To overcome some of the limitations of the ε-constraint approach, [augmented ε-constraint method (AUGMECON)] [45] was developed.

AUGMECON guarantees the Pareto-efficiency of the solutions by using lexicographic optimization[1] of the objective functions. Therefore, the AUGMECON method is implemented in this paper. Through using AUGMECON method for Phase II, the output is the scheduled assignment of the VCs over the periods, which is represented by a binary decision variable s_{ij}.

Input to Phase III is the unique VCs assigned to monitoring in Phase II. Output of Phase III is the sequence that minimizes the total number of state transitions of the nodes. The sequence is generated using a dynamic programming implementation of the TSP problem. Fig. 6 shows a comparison between the sequencing of the VCs assigned in Phase II [Fig. 6(a)] on DODAG number 3, and the sequence after solving the TSP Path of Phase III [Fig. 6(b)]. The new sequence reduces the total number of state transitions by 70%.

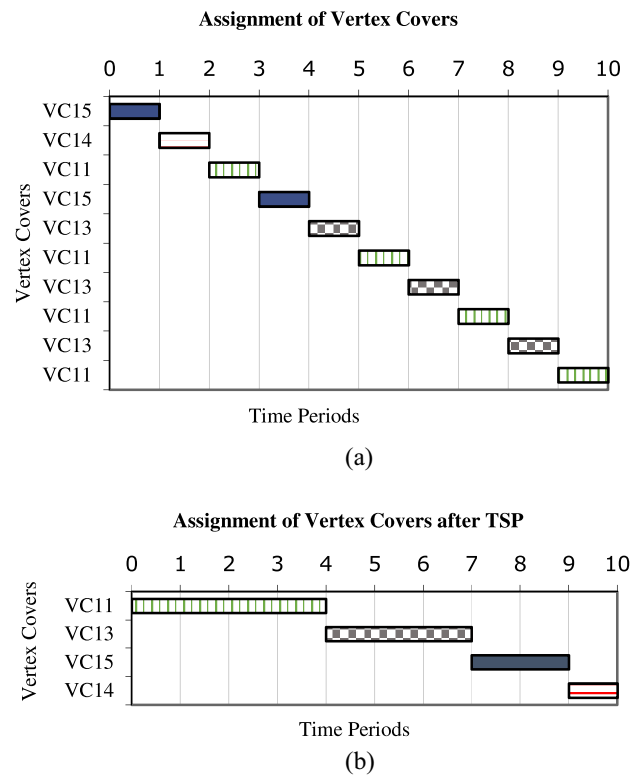

Fig. 6. Assignment of VCs. (a) Before TSP and (b) after TSP.

B. Performance Evaluation

The experiments were performed on a personal computer with 8 gigabite RAM and an Intel Core i7 processor @2.20 gigahertz. Considering the problem formulations presented in Section V, the metrics used for evaluation (shown in Table I) are as follows.

1) Percentage of nodes selected as monitoring nodes.
2) Average residual energy among monitors.
3) Percentage of reduction in the nodes' state transitions.

The originality of this paper is mainly in the proper modeling of the defined problem. Modeling the monitor selection problem as VCP in Phase I guarantees optimal monitor placement to cover the entire set of links. Simultaneously, the percentage of deployed monitors is relatively small, (52%–66%, depending on the density of the network). Consequently, it is possible to identify fine-grained performance link metrics, where monitors send passive probes in an end-to-end approach. This leads to reducing the need for active probes; thus less monitoring overhead. In addition, active probes can be used when needed, to check the availability of network parts, where communication has not been established for long times. Moreover, the proposition in Phase II is able to optimally assign monitors to periods with minimum energy consumption, depending on the number of periods ($|T|$), the energy loss per period ($\boldsymbol{E_m}$), and the reserved battery per node ($\boldsymbol{Reserved_battery_k}$). The values of the parameters used for the test instances have been chosen arbitrarily. This is because the main overriding objective of this paper is to emphasize that energy-efficient solution methods for this difficult problem do exist. That being said, it can be seen that, after setting the parameters to the values: ($|T| = 10, \boldsymbol{E_m} = 2\%, \boldsymbol{Reserved_battery_k} = 50\%$ and $\boldsymbol{Initial_battery_k} = 100\%$), the average residual battery throughout the eight instances depicted in Table I is in the range [86%–98%].

It is interesting to emphasize that when the battery dedicated for monitoring is sufficiently large, fewer VCs are assigned to periods and less monitor scheduling is required.

On the other hand, when the $\boldsymbol{Reserved_battery_k}$ is relatively small, more VCs are required to monitor the same number of periods and scheduling for minimal energy consumption is critical. Fig. 7 shows the minimum, average, and maximum remaining (residual) battery after monitoring for a network of 50, 100, 150, and 200 nodes. For $E_m = 2\%$, the minimum residual battery does not fall below 80% and the average residual battery is between the range [83%–90%]. Also, when the model was tested without running Phase I, all the nodes were assigned to monitoring.

Consequently, the nodes' battery level dropped to the minimum level. Fig. 8, emphasizes the model's capability of monitoring a dense network of 200 nodes with increasing number of links, without depleting the nodes' reserved battery for monitoring. The figure shows that the density of the network grows up to 2463 links and yet the average residual battery is in the range [82%–98%].

The results after solving the TSP path, for the optimal sequencing of the VCs across the time periods, are very promising. The model is both effective and efficient in reducing the state transitions of nodes up to 80%. It is worth mentioning that in some instances, like DODAG number 8 in Table I, the solution from Phase II is already optimum with

[1]Lexicographic optimization is the sequential optimization of multiple objective functions; by optimizing one function and from the pool of alternative optima sequentially optimize the next.

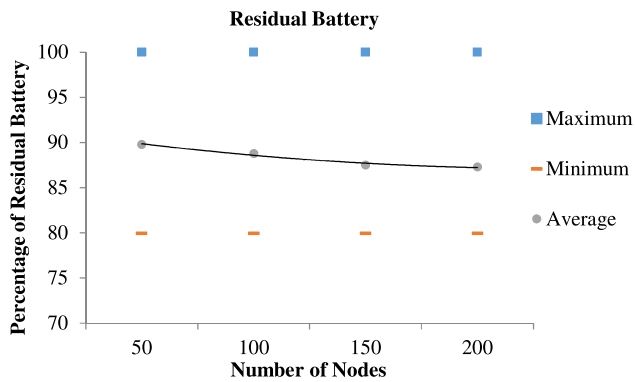

Fig. 7. Residual battery after monitoring for different-sized networks.

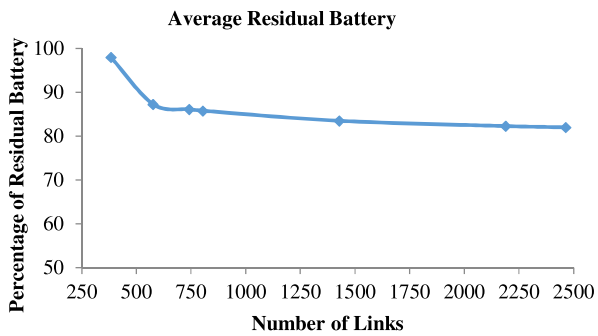

Fig. 8. Average residual battery after monitoring for a network of 200 nodes and varying number of links.

respect to the number of state transitions. Therefore, the percentage reduction of state transitions after the TSP path is sometimes zero.

VII. Complexity Analysis

The proposed model's time complexity depends on several factors.

1) $|\boldsymbol{V}|$: Size of the DODAG, in terms of the number of nodes.
2) $|\boldsymbol{M}|$: Number of VCs obtained from Phase I.
3) $|\boldsymbol{T}|$: Number of periods in the planning horizon.
4) u: Number of unique VCs assigned to monitor the periods (solution of Phase II).

Lemma 1: Phase I has a time complexity of $\mathbf{O}(2^{|\boldsymbol{V}|}.|\boldsymbol{V}|^2)$.

Proof: The integer programming solution for the VCP includes a nested loop that runs in exactly $(1/2)(|\boldsymbol{V}|-1).|\boldsymbol{V}|$. This nested loop is analogous to the constraint in (3) in Model 1. The constraint generation algorithm (Algorithm 1) has a constant running time of θ, where θ represents the required number of generated VCs. As mentioned in Section VI, it is noticed from the preliminary experimentation that it is sufficient to set θ in the range $[1.5|\boldsymbol{T}|– 3|\boldsymbol{T}|]$. The BB algorithm is used to solve the VCP. The complexity of BB is lower bounded by the total number of nodes, which is proportional to $2^{|\boldsymbol{V}|}$ [24]. Hence, Phase I has a total running time of $((1/2)|\boldsymbol{V}|-1).(|\boldsymbol{V}|+\theta).2^{|\boldsymbol{V}|}$, i.e., $\mathbf{O}(2^{|\boldsymbol{V}|}.|\boldsymbol{V}|^2)$.

As VCP is NP-hard, a lot of research has been conducted to produce efficient approximation algorithms (see [36], [37]). The work in [1] develops a polynomial-time algorithm that converts the DODAG into a nice-tree decomposition with unity treewidth. Using the algorithm proposed in [1] yields a significant reduction in the complexity of solving VCP on DODAGs. The problem becomes polynomial-time solvable, even though it is run iteratively using a constraint generation algorithm.

It can be seen from Table I that the majority of computations is relatively centered in the multiobjective GAP in Phase II. For example, DODAG number 4 with 100 nodes and 347 links requires around 10 s to reach the optimal solution for the VCP, and less than 1 s for the TSP Path, and 600 s to reach the optimal solution for the multiobjective GAP. The problem of large running time exists for instances having more than 200 nodes and 350 links. The optimum solution is still reachable, albeit slowly. Fortunately, there exist several approximation algorithms for GAP. For instance, [39] presented a polynomial-time 2-approximation algorithm for GAP.

Lemma 2: Phase II has a total time complexity of $\mathbf{O}(2^{(|\boldsymbol{M}|.|\boldsymbol{T}|)}.\ (|\boldsymbol{V}|+|\boldsymbol{M}|+|\boldsymbol{T}|))$.

Proof: To prepare the coefficients for the communication objective function F_2 as represented in (17), the solution loops $|\boldsymbol{M}|$ times. Moreover, a loop of $|\boldsymbol{T}|$ times is required to enforce a period to be assigned once, which is represented in (19). Also, to constraint the energy used for monitoring of each node to be less than or equal to the $\boldsymbol{Reserved_battery_k}$ the solution loops $|\boldsymbol{V}|$ times [represented in (20)]. The number of variables in the GAP is $|\boldsymbol{M}|.|\boldsymbol{T}|$. Using the BB algorithm for solving the GAP, which is bounded by the total number of variables, gives an overall time complexity for Phase II of $\mathbf{O}(2^{(|\boldsymbol{M}|.|\boldsymbol{T}|)}.\ (|\boldsymbol{V}|+|\boldsymbol{M}|+|\boldsymbol{T}|))$.

Lemma 3: Phase III has time complexity of $\mathbf{O}(\mathbf{2^u}.\boldsymbol{u}^2)$.

Proof: Although the TSP path is also NP-hard and there are approximation algorithms proposed in the literature (see [6]), its running time in the proposed model is relatively small. This is due to the fact that the input of Phase III is only the unique VCs assigned to monitor the periods (u) from Phase II. TSP path is solved using Held–Karp dynamic programming algorithm [17]. There are at most $(2^u.u)$ subproblems, each of which takes linear time (u) to solve. Therefore, the time complexity of Phase III is $\mathbf{O}(\mathbf{2^u}.\boldsymbol{u}^2)$.

VIII. Conclusion

In this paper, a three-phase mathematical model was proposed to address the optimized scheduling of the monitoring role in IoT networks. In the first phase, multiple subsets of the nodes that cover the entire graph are generated by solving VCP iteratively; while incorporating a constraint generation algorithm. The optimized scheduling of the VCs is handled in Phases II and III. The multiple VCs are input to Phase II which is modeled as a multiobjective GAP. The objectives are to minimize the energy and communication costs incurred while monitoring the network. Input to Phase III is the unique VCs assigned to monitoring in Phase II. Phase III is a TSP problem with the objective of further minimizing the energy consumption by reducing nodes' state transitions (from active to sleep and vice versa).

The proposed solution is a proof of concept to emphasize that energy-efficient solutions for monitoring IoT networks do exist. The proposed model and solution method were tested on a number of test instances of different sizes, ranging from

50 to 200 nodes, and from 123 to 2463 links. Experimental results show that, for the sizes of the tested instances, the proposed model was indeed effective and scalable in achieving the monitoring objective, while providing load balancing between monitors and minimizing the cost of monitoring in terms of energy and communication costs, and the number of nodes' state transitions.

Future work will be the development of heuristics and approximation algorithms to reduce the complexity of Phase II, since it was noticed to be the most time consuming phase. Often in IoT, the underlying network can change dynamically (nodes, links, topology, etc.), in order to handle these changes, it will be interesting to develop incremental methods and dynamic algorithms for the system control (including the monitoring). To test the energy consumed for transmission of monitoring probes by the nodes, a network simulator should be used. Therefore, testing the proposed model on Cooja network simulator is another possible future work.

Acknowledgment

Author B. Mostafa would like to thank the L'Oréal foundation and UNESCO for their support by granting her the 2017 L'Oréal-UNESCO for Women in Science Levant and Egypt Fellowship.

References

[1] B. Mostafa, A. Benslimane, E. Boureau, M. Molnar, and M. Saleh, "Distributed monitoring in 6LoWPAN based Internet of Things," in *Proc. MoWNeT*, Cairo, Egypt, 2016, pp. 1–7, doi: 10.1109/MoWNet.2016.7496626.

[2] J.-P. Vasseur and A. Dunkels, *Interconnecting Smart Objects With IP: The Next Internet.* Burlington, MA, USA: Morgan Kaufmann, 2010.

[3] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.

[4] J. Lin *et al.*, "A survey on Internet of Things: Architecture, enabling technologies, security and privacy, and applications," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1125–1142, Oct. 2017.

[5] S. He, J. Chen, and Y. Sun, "Coverage and connectivity in duty-cycled wireless sensor networks for event monitoring," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 3, pp. 475–482, Mar. 2012.

[6] A. J. Jara, L. Ladid, and A. Skarmeta, "The Internet of everything through IPv6: An analysis of challenges, solutions and opportunities," *J. Wireless Mobile Netw. Ubiquitous Comput. Depend. Appl.*, vol. 4, no. 3, pp. 97–118, 2013.

[7] J. A. Stankovic, "Research directions for the Internet of Things," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 3–9, Feb. 2014.

[8] L. Paradis and Q. Han, "A survey of fault management in wireless sensor networks," *J. Netw. Syst. Manag.*, vol. 15, no. 2, pp. 171–190, 2007.

[9] F. Ye, G. Zhong, S. Lu, and L. Zhang, "GRAdient broadcast: A robust data delivery protocol for large scale sensor networks," *Wireless Netw.*, vol. 11, no. 3, pp. 285–298, 2005.

[10] D. Ganesan, R. Govindan, S. Shenker, and D. Estrin, "Highly-resilient, energy-efficient multipath routing in wireless sensor networks," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 5, no. 4, pp. 11–25, 2001.

[11] R. R. Swain, P. M. Khilar, and S. K. Bhoi, "Heterogeneous fault diagnosis for wireless sensor networks," *Ad Hoc Netw.*, vol. 69, pp. 15–37, Feb. 2018.

[12] M. Panda and P. M. Khilar, "Distributed self fault diagnosis algorithm for large scale wireless sensor networks using modified three sigma edit test," *Ad Hoc Netw.*, vol. 25, pp. 170–184, Feb. 2015.

[13] S. A. Khan, B. Daachi, and K. Djouani, "Application of fuzzy inference systems to detection of faults in wireless sensor networks," *Neurocomputing*, vol. 94, pp. 111–120, Oct. 2012.

[14] Y. J. Zhao, R. Govindan, and D. Estrin, "Residual energy scan for monitoring sensor networks," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Orlando, FL, USA, 2002, pp. 356–362.

[15] S. K. Devi, "Survey on sleep scheduling methods in wireless sensor networks," *Int. J. Adv. Res. Comput. Eng. Technol.*, vol. 3, no. 1, pp. 81–85, 2014.

[16] A. Keshavarzian, H. Lee, and L. Venkatraman, "Wakeup scheduling in wireless sensor networks," in *Proc. MobiHoc*, Florence, Italy, 2006, pp. 322–333.

[17] M. Molnár, G. Simon, and L. Gönczy, "Quasi-optimal scheduling algorithm for area coverage in multi-functional sensor networks," *Int. J. Ad Hoc Ubiquitous Comput.*, vol. 14, no. 2, pp. 109–122, 2008, doi: 10.1504/IJAHUC.2013.056418.

[18] C. Razafimandimby, V. Loscrì, A. M. Vegni, and A. Neri, "Efficient Bayesian communication approach for smart agriculture applications," in *Proc. IEEE 86th Veh. Technol. Conf. (VTC-Fall)*, Toronto, ON, Canada, Sep. 2017.

[19] S. Fang *et al.*, "An integrated system for regional environmental monitoring and management based on Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1596–1605, May 2014.

[20] A. J. Jara, M. A. Zamora-Izquierdo, and A. F. Skarmeta, "Interconnection framework for mHealth and remote monitoring based on the Internet of Things," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 9, pp. 47–65, Sep. 2013.

[21] A. Woo, T. Tong, and D. Culler, "Taming the underlying challenges of reliable multihop routing in sensor networks," in *Proc. 1st Int. Conf. Embedded Netw. Sensor Syst. (SenSys)*, Los Angeles, CA, USA, 2003, pp. 14–27.

[22] S. Andreozzi *et al.*, "On the integration of passive and active network monitoring in grid systems," in *Integrated Research in GRID Computing*. Boston, MA, USA: Springer, 2007, pp. 147–161.

[23] M. Cardei, M. T. Thai, Y. Li, and W. Wu, "Energy-efficient target coverage in wireless sensor networks," in *Proc. 24th Annu. Joint Conf. IEEE Comput. Commun. Soc. (INFOCOM)*, Miami, FL, USA, 2005, pp. 1976–1984, doi: 10.1109/INFCOM.2005.1498475.

[24] F. El-Moukaddem, E. Torng, and G. Xing, "Maximizing network topology lifetime using mobile node rotation," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 7, pp. 1958–1970, Jul. 2015. doi: 10.1109/TPDS.2014.2329851.

[25] Y.-M. Cho, S.-J. Lee, C. Kim, and S. Kim, "An energy scheduling algorithm for ensuring the pre-determined lifetime in sensor network," in *Proc. Int. Conf. Embedded Ubiquitous Comput.*, Taipei, Taiwan, 2007, pp. 467–473.

[26] S. Raza, L. Wallgren, and T. Voigt, "SVELTE: Real-time intrusion detection in the Internet of Things," *Ad Hoc Netw.*, vol. 11, no. 8, pp. 2661–2674, 2013, doi: 10.1016/j.adhoc.2013.04.014.

[27] R. Giuliano, F. Mazzenga, A. Neri, and A. M. Vegni, "Security access protocols in IoT capillary networks," *IEEE Internet Things J.*, vol. 4, no. 3, pp. 645–657, Jun. 2017.

[28] H. Sedjelmaci, S. Senouci, and T. Taleb, "An accurate security game for low-resource IoT devices," *IEEE Trans. Veh. Technol.*, vol. 66, no. 10, pp. 9381–9393, Oct. 2017.

[29] N. Kushalnagar, G. Montenegro, and C. Schumacher. (2007). *IPv6 Over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals*. Accessed: Jul. 19, 2017. [Online]. Available: https://tools.ietf.org/ html/rfc4919

[30] Z. Sheng *et al.*, "A survey on The IETF protocol suite for the Internet of Things: Standards, challenges, and opportunities," *IEEE Wireless Commun.*, vol. 20, no. 6, pp. 91–98, Dec. 2013, doi: 10.1109/MWC.2013.6704479.

[31] J. Martocci, W. Vermeylen, N. Riou, and P. D. Mil. (2010). *Building Automation Routing Requirements in Low Power and Lossy Networks*. Accessed: Nov. 10, 2017. [Online]. Available: https://tools.ietf.org/html/rfc5867.html

[32] K. D. Korte, A. Sehgal, and J. Schönwälder, "A study of the RPL repair process using ContikiRPL," in *Proc. 6th IFIP WG 6.6 Int. Conf. Auton. Infrastruct. Manag. Security Depend. Netw. Services (AIMS)*, 2012, pp. 50–61.

[33] J. P. Vasseur, M. Kim, K. Pister, N. Dejean, and D. Barthel. (2011). *Routing Metrics Used for Path Calculation in Low Power and Lossy Networks*. Accessed: Sep. 1, 2017. [Online]. Available: https://tools.ietf.org/html/draft-ietf-roll-routing-metrics-19

[34] E. Salhi *et al.*, "Anomaly detection joint optimization of monitor location and network anomaly detection," in *Proc. IEEE LCN*, 2010.

[35] J. Ma, W. Lou, Y. Wu, X.-Y. Li, and G. Chen, "Energy efficient TDMA sleep scheduling in wireless sensor networks," in *Proc. IEEE INFOCOM*, Rio de Janeiro, Brazil, 2009, pp. 630–638.

[36] N. Bourgeois, B. Escoffier, and V. T. Paschos, "Approximation of max independent set, min vertex cover and related problems by moderately exponential algorithms," *Discr. Appl. Math.*, vol. 159, no. 17, pp. 1954–1970, 2011, doi: 10.1016/j.dam.2011.07.009.

[37] M. Åstrand and J. Suomela, "Fast distributed approximation algorithms for vertex cover and set cover in anonymous networks," in *Proc. 22nd ACM Symp. Parallelism Algorithms Archit.*, 2010, pp. 294–302, doi: 10.1145/1810479.1810533.

[38] G. Laporte, "The traveling salesman problem: An overview of exact and approximate algorithms," *Eur. J. Oper. Res.*, vol. 59, no. 2, pp. 231–247, 1992.

[39] D. B. Shmoys and É. Tardos, "An approximation algorithm for the generalized assignment problem," *Math. Program.*, vol. 62, nos. 1–3, pp. 461–474, 1993.

[40] S. Balaji, V. Swaminathan, and K. Kannan, "Optimization of unweighted minimum vertex cover," *Int. J. Math. Comput. Phys. Elect. Comput. Eng.*, vol. 4, no. 7, 2010.
[41] B. Boots, G. J. Gordon, and S. M. Siddiqi, "A constraint generation approach to learning stable linear dynamical systems," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2008, pp. 1329–1336.
[42] D. W. Pentico, "Assignment problems: A golden anniversary survey," *Eur. J. Oper. Res.*, vol. 176, no. 2, pp. 774–793, 2007, doi: 10.1016/j.ejor.2005.09.014.
[43] F. Lam and A. Newman, "Traveling salesman path problems," *Math. Program.*, vol. 113, no. 1, pp. 39–59, 2008.
[44] K. Khalili-Damghani, M. Tavana, and S. Sadi-Nezhad, "An integrated multi-objective framework for solving multi-period project selection problems," *Appl. Math. Comput.*, vol. 219, no. 6, pp. 3122–3138, 2012, doi: 10.1016/j.amc.2012.09.043.
[45] G. Mavrotas, "Effective implementation of the ϵ-constraint method in multi-objective mathematical programming problems," *Appl. Math. Comput. J.*, vol. 213, no. 2, pp. 455–465, Jul. 2009.
[46] R. Bellman, "Dynamic programming treatment of the travelling salesman problem," *J. ACM*, vol. 9, no. 1, pp. 61–63, 1961.
[47] D. Roy and G. V. Rao, *Stochastic Dynamics, Filtering and Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2017.
[48] X. Liu, Y. Gao, W. Wu, W. Dong, and J. Bu, "Robust monitor assignment with minimum cost for sensor network tomography," *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 8, pp. 1–6, 2015.

Basma Mostafa received the B.S. and M.S. degrees in operations research from the Faculty of Computers and Information, Cairo University, Cairo, Egypt, in 2008 and 2013, respectively. She is currently pursuing the dual Ph.D. degree in operations research at the University of Montpellier, Montpellier, France, and Cairo University.

She has been an Assistant Lecturer with the Faculty of Computers and Information, Cairo University, since 2014. Her current research interests include development of efficient and optimized models for monitoring IoT networks.

Ms. Mostafa was a recipient of the 2017 L'Oréal–UNESCO Award for Women in Science Levant and Egypt Fellowship.

Abderrahim Benslimane (M'99–SM'08) received the B.S. degree in computer science from the University of Nancy, Nancy, France, in 1987, and the DEA (M.S.) and Ph.D. degrees in computer science from the Franche-Comte University, Besançon, France, in 1989 and 1993, respectively.

He has been a Full Professor of computer-science with Avignon University, Avignon, France, since 2001. He has been recently a Technical International Expert with the French Ministry of Foreign and European Affairs, from 2012 to 2016. He served as a Coordinator with the Faculty of Engineering, French University in Egypt, El Shorouk, Egypt. He has been an Associate Professor with the University of Technology of Belfort–Montbéliard, Belfort, France, since 1994. He obtained the title to supervise research (HDR 2000) from the University of Cergy–Pontoise, Cergy, France. He has been involved in many national and international projects, such as ANR, PHC, and H2020. His current research interests include development of secure communication protocols for vehicular networks and Internet of Things. He has over 140 refereed international publications, including book, journals, and conferences in the above areas.

Dr. Benslimane was a recipient of the French Award of Scientific Excellency from 2011 to 2014. He is an Area Editor of *Wiley Security and Privacy Journal* and an Editorial Board member of *IEEE Wireless Communication Magazine* and *Elsevier Ad Hoc*. He has been serving as the General-Chair of IEEE WiMob since 2008; he was launched and has been serving as the General-Chair of iCOST and MoWNet since 2011. He served as the Symposium Co-Chair/Leader in many IEEE international conferences, such as ICC, Globecom, AINA, and VTC. He was a Guest Editor of many special issues. He participates in the Steering and the Program Committee of many IEEE international conferences. He was a Board Committee member, the Vice-Chair of student activities of the IEEE France Section/Region 8. He was the Publication Vice-Chair, the Conference Vice-Chair, and is currently the Chair of the IEEE Communication Society TC of Communication and Information Security. He participates on the Steering and the Program Committees of many IEEE international conferences. He was a member of the French Council of Universities (CNU Section 27) from 2003 to 2007.

Mohamed Saleh received the master's degree from Bergen University, Bergen, Norway, the M.B.A. degree from the Maastricht School of Management, Maastricht, The Netherlands, and the Ph.D. degree in system dynamics from the University of Bergen, Bergen.

He is a Professor and the former Head with the Faculty of Computers and Information, Department of Operations Research and Decision Support, Cairo University, Cairo, Egypt. He is also an Adjunct Professor with the System Dynamics Group, University of Bergen. He has authored or co-authored numerous papers in several international journals and conferences. He is currently the Manager of the Virtual Center of Excellence for Data Mining and Computer Modeling, Cairo University. His current research interests include system dynamics, simulation, futures studies, revenue management optimization, and management science.

Dr. Saleh was a recipient of the IBM Faculty Award.

Sally Kassem received the graduation degree in 1998 and M.Sc. degree in industrial engineering from the Faculty of Engineering, Mechanical Design and Production Department, Cairo University, Cairo, Egypt, and the Ph.D. degree in industrial engineering from Concordia University, Montreal, QC, Canada, in 2011.

She has been an Assistant Professor with the Faculty of Computers and Information, Department of Operations Research and Decision Support, Cairo University since 2012. She is also an Assistant Professor with the School of Engineering and Applied Science, Industrial and Service Engineering Program, Nile University, Giza, Egypt. Her current research interests include mathematical modeling and optimization, linear and integer programming, operations research methodologies and tools, supply chain, logistics, and modeling and simulation.

Miklos Molnar received the graduation degree from the Faculty of Electrical Engineering, University BME, Budapest, Hungary, in 1976, the Ph.D. degree in computer science from the University of Rennes 1, Rennes, France, in 1992, and the French HDR degree in 2008.

He has been with the University Montpellier, Montpellier, France, since 2010. He is a Full Professor with the Department of Computer Science, IUT. His research activities are in combinatorial optimization, network design and optimization algorithms and tools in the laboratory LIRMM of Montpellier. He conducted several studies to find dependable routes for sensible communications, efficient multicast routes, energy aware k-coverage and routing protocols, and different optimizations in ad hoc wireless networks.

Dr. Molnar participates as a PC member in the organization of conferences and on the Editorial Board of several journals.

View publication stats