Monitoring Internet of Things Networks

Basma Mostafa[1,2]

***Abstract*—To ensure robustness, functionality and Quality of Service in wireless networks, monitoring the network state and functioning of nodes and links is crucial; especially for critical applications. This PhD thesis targets robustness in Internet of Things (IoT) networks. Devices are resource-constrained and connected via lossy links; therefore, fault prevention and rapid repair mechanisms are crucial. Meanwhile, monitoring should minimize the resulting energy and traffic overhead; to leave the network unconstrained during its normal operation. To tackle this problem, several integrated optimization models and efficient algorithms were proposed during the course of PhD. Our activities and results cover the monitor placement and scheduling problems. The topology is represented by a graph, and several graph related optimization problems can be solved. We aim at realizing a polynomial-time solvable monitor placement algorithm. Furthermore, to minimize the monitoring overhead and maximize longevity, monitoring roles should be balanced and alternated amongst nodes; therefore, we target optimal monitor scheduling. We propose a Binary Integer Programming problem formulation. We present the exact solution as well as an efficient heuristic. Extensive experimentation was conducted using different network sizes and topologies. Results confirm effective monitoring with minimum energy consumption and network overhead while balancing the monitoring role between nodes.**

I. PROBLEM STATEMENT

Most of IoT networks are comprised of Low-power Wireless Personal Area Networks (LoWPAN) which adopt IPv6, creating what is known as 6LoWPANs [1]. Maintaining robustness in such networks is particularly challenging because devices are (1) connected via wireless unreliable, lossy channels, which makes disconnectivity, node unreachability and eavesdropping extremely common, (2) usually resource-constrained with low-power radio and limited and unpredictable bandwidth, (3) vulnerable to security risks from the Internet, (4) and unattended and possibly deployed in hostile, highly dynamic environments; which makes them susceptible to physical attacks [2].

Although a significant number of IoT applications are not time-sensitive, there is a whole class of real-time, mission-critical applications; where data must be processed and shared instantly and within strict reliability constraints. For instance, critical control, safety and health monitoring applications. To achieve fault tolerance in such sort of services, corrective actions must be taken within little delay. Network monitoring provides the appropriate tools for overseeing the network state, availability of nodes and links. Consequently, connectivity problems can be effectively detected and localized; in order to take fast, corrective measures.

[1,2] Basma Mostafa is with University of Montpellier, Laboratoire d'Informatique, de Robotique et de Microelectronique de Montpellier (LIRMM), 34090 Montpellier, France & with Faculty of Computers & Information, Cairo University, Cairo, 12411, Egypt. basma.mostafa@lirmm.fr

978-1-5386-4980-0/19/$31.00 ©2019 IEEE

II. RESEARCH GOALS AND METHODOLOGY

Routing protocols tend to favor route stability over fault tolerance. They are able to respond to some faults by running reactive route repair mechanisms. However, the existence of a *proactive approach*; where failures are captured and rapidly mitigated is extensively preferable for mission-critical IoT applications. In proactive monitoring, continual maintenance is enforced to alert network operators to the presence of faults; consequently, disconnectivity, node unreachibility and service failures are prevented from occurring in the first place. This could greatly improve robustness and Quality of Service (QoS); which will eventually increase the uptake of the technology by stakeholders.

IoT devices are usually resource-constrained and cannot adopt complex monitoring mechanisms. The network should be left unconstrained during its normal operation. Therefore, monitoring mechanisms should be *efficient*; such that the resulting energy and traffic overhead are minimized. Furthermore, to ensure QoS and reliable communication between elements, the system state monitoring should be a *periodic activity*.

Since it has been standardized, the Routing Protocol for Low Power and Lossy Networks (RPL) is considered to be the de-facto routing protocol over IP-connected IoT. Utilizing its characteristics and proposing monitoring models that work *in tandem with RPL* is highly preferable.

IoT networks are often dynamic; new elements may appear, and the topology and state of the system tend to change. To follow the changes by repeated resolutions of the problem is inefficient. Therefore, it is necessary to target *dynamic* monitoring algorithms; which include incremental methods that are able to adapt to real-time changes by reusing the existing, unaffected part of a previous solution.

To summarize the goals of our monitoring system, we aim at maintaining a highly reliable IoT network structure by:

- proactively and efficiently verifying the correct operation of nodes and links,
- collecting, aggregating and filtering real-time data from nodes,
- detecting and localizing (or even predicting) abnormal events or faults, and
- adapting to dynamic, real-time changes in the network state.

To achieve the stated goals, our research methodology is the following:

- extensive reviews to the state of the art of monitoring Wireless Sensor Networks (WSNs),

- creation of robust models and corresponding graph optimization problems,
- analysis of the proposed models from the point of view of complexity and resolvability,
- developing exact and approximated analytical solutions to the related graph problems,
- integrating the proposed models with active and/or passive network monitoring algorithms, and
- performing extensive simulations for performance evaluations; to verify the effectiveness and efficiency of the proposed models.

III. PROPOSED MODELS

One of the main challenges for network monitoring is determining where to embed (place) the monitoring nodes. These elements should have the possibility for actively/passively running monitoring probes and/or analyzing the monitoring results. The probes' placement must be optimized in order to minimize the energy cost and monitoring load. Moreover, the monitoring computational cost, battery and memory requirements should be minimal in order to satisfy the low cost and energy constraints of IoT devices. Network monitoring examples show that the related optimization is often NP-hard [3]. We begin by developing a model which aims at the optimum placement of monitors while ensuring network coverage and computational tractability.

A. Fixed Parameter Tractable Monitoring Placement Algorithm

The network topology and communication between elements can be modeled by a graph. Since the proposed models should work in tandem with RPL, the graph we use is the Destination Oriented Directed Acyclic Graph (DODAG) constructed by RPL. Optimal monitor placement is finding out the minimum number of monitoring nodes placed on the graph to keep track of all the links in the network. The problem can be modeled as the classic Vertex Cover Problem (VCP) [4]. VCP is NP-hard for general graphs. On the other hand, it is polynomial when solved on trees and Fixed Parameter Tractable (FPT) when solved on "tree-like" graphs, also called nice-tree decompositions [5], and the bounding parameter here is the treewidth. In light of this information, we proposed algorithms 1 and 2 in [6] to convert the DODAG representing the network topology into a nice-tree decomposition with unity treewidth (cf. Algorithm 1 for a brief description of the first algorithm). Bounding the treewidth to the value of one has the effect of reducing the complexity of solving the VCP on the DODAG to be polynomial-time, which is the main contribution of this part of the work.

Algorithm 1: Convert DODAG into Nice-Tree with Treewidth 1

Input: $D = (\{v_j : v_j \in V\}, E)$
Output: *Nice Tree Decomposition* $(\{X_i : i \in I\}, T)$ *with unity Treewidth*

Let $c_r \in C(v_j)$ **where** $C(v_j)$ *is the set of children of vertex* v_j *in DODAG* D;
Let $s_r \in$ $S(v_j)$ **where** $S(v_j)$ *is the set of siblings connected to vertex* v_j *in DODAG* D;
Non-Leaf nodes: $\{NL : NL \longleftarrow \cup\, v_j \in D$ **where** $\mid C(v_j) \mid + \mid S(v_j) \mid \ > 0\}$;
Initialization;
while $NL \neq 0$ **do**
  Select v_j from NL (its top vertex);
  Search T for bag $X_i \longleftarrow v_j$;
  Set $L \longleftarrow C(v_j) \cup S(v_j)$ **where** $s_r \in NL$;
  Let $no\_of\_required\_leaves \longleftarrow$|L| ;
  if $no\_of\_required\_leaves > 1$ **then**
    $t \longleftarrow$ ConstructBinaryTree(X_i, $no\_of\_required\_leaves$);
    At Xi augment T with t;
  endif
  while $l \leq no\_of\_required\_leaves$ **do**
    Make $leaf$ a forget bag via branching Xk, **where** $Xk \longleftarrow vj, vq$ **and** $v_q \in L$;
    Make Xk an "introduce bag" via branching vq;
    $L \longleftarrow L - v_q$;
    $l \longleftarrow l + 1$;
    $NL \longleftarrow NL - vj$;
  endWhile
endWhile

B. Three-Phase Heuristic for Monitoring Scheduling

Since WSNs and LoW-power Lossy Networks (LLNs) are mostly constrained by energy consumption, idle listening to the channel can quickly cause battery depletions. Duty cycling is commonly incorporated in such type of networks to maximize longevity. In duty cycling, a node frequently slips into a sleeping state and periodically wakes up to perform its sensing, receiving or transmission role. The same periodic activity can be applied to monitoring. We assume that the monitoring system reports the status of network components in prescheduled epochs. The frequency of epochs depends on the criticality of the application. The objective of this alternation is balancing the burden of monitoring across several subsets of nodes. Monitoring subsets wake up at prescheduled points in time to carry out their monitoring responsibility and go back to sleep-monitoring; so that other nodes continue the job. Unfortunately, there's an undesirable effect to duty cycling; multiple state transitions between active, sleep, and transient states consume extra energy [7]. Consequently, another goal for a resource-aware monitoring system is to optimally schedule the monitoring role between nodes while minimizing the number of monitoring state transitions.

To address the optimized scheduling of the monitoring role, we proposed in [8] a three-phase decomposition of the problem. In the first phase, multiple minimal sets of monitors are generated by iteratively solving the VCP using a Constraint Generation algorithm. The algorithm enumerates possible minimal Vertex Covers (VCs) and runs until no more feasible VCs exist.

The optimized scheduling of VCs is handled in Phases II and III. Assuming periodical functioning, VCs are optimally assigned to time periods in PhaseII. The assignment is modeled as a multiobjective Generalized Assignment Problem (GAP) (cf. mathematical model in Table I). The objectives are to minimize the energy and communication costs incurred while monitoring the network. Input to Phase III are the unique VCs assigned to monitoring in Phase II; while the output is the sequence that minimizes the total number of state transitions (from active-monitoring to sleep and vice versa). The sequence is generated using a dynamic programming implementation of the Traveling Salesman Path Problem (TSP).

Table I
PHASE II, MULTIOBJECTIVE GAP

Variables	Description	
y_{ki}	$= \begin{cases} 1, & \text{if } v_k \in vc_i \\ 0, & \text{otherwise} \end{cases}$	
h_k	Number of hops traveled from each v_k in vc_i to the root of the DODAG	
H_i	Total number of hops traveled by all v_k in vc_i	$=\sum y_{ki} \times h_k \forall k \in V$
E_m	Energy loss per each monitoring period assigned for v_k	
$reserved\_battery_k$	Maximum battery allowed for monitoring	
Let $s_{i,j}$	$= \begin{cases} 1, & \text{if } vc_i \text{ is assigned to period } j \\ 0, & \text{otherwise} \end{cases}$	

Model Equations

$$\min F_1 = \sum_{k \in V} E_m \times (\sum_{i=1}^{m} \sum_{j=1}^{n} y_{ki} \times s_{i,j}) \tag{1}$$

$$\min F_2 = \sum_{i=1}^{m} \sum_{j=1}^{n} H_i \times s_{i,j} \tag{2}$$

$$s.t. \quad \sum_{i=1}^{m} s_{i,j} = 1 \forall j \in T \tag{3}$$

$$E_m \times (\sum_{i=1}^{m} \sum_{j=1}^{n} y_{ki} \times s_{i,j}) \leq reserved\_battery_k \forall k \in V \tag{4}$$

$$s_{i,j} \in 0, 1 \forall i, j \tag{5}$$

C. Exact Solutions for Passive Monitoring

The overriding objective of the thesis all along is providing an energy-efficient monitoring mechanism for IoT networks. We choose to develop a passive monitoring approach; thus, minimizing the overhead on the network. Nonetheless, active probes can be used only when required; to accurately localize faults or check the availability of network parts where communication has not been established for long times.

The sniffer technology for WSNs is one of the distinguished passive, real-time monitoring tools. A sniffer (or passive monitoring node) will (1) eavesdrop on the communicated messages between its neighbor set within radio coverage, (2) assemble data reflecting the current status of neighboring nodes and links, and (3) route the monitoring data through an optimal path towards a central node, usually the Border Router (BR), which is responsible for taking corrective measures.

We propose exploiting the RPL for our monitoring purposes. By using RPL's multiple instances feature, routes in the DODAG are built for the sole purpose of node/link monitoring; to take the burden off regular nodes. Moreover, optional headers for metrics and constraints objects, called DAG Metric Containers, are added to the DODAG Information Objects (DIO) used for topology construction. These sub-objects provide information on the status of nodes and links; for example, estimated remaining power-level and link reliability [9].

Although the three-phase decomposition was a good heuristic, the exact solution of the IoT monitoring problem has been missing from the literature. It will serve as a benchmark for performance comparisons. We proposed a Binary Integer Programming (BIP) formulation for the exact model. We implement it using Julia; a high-performance dynamic programming language for numerical computing and solve it using Gurobi solver. Extended computational experiments were also conducted using different network topologies; in order to verify and validate the proposed models.

IV. EXPERIMENTAL EVALUATION

The experiments are performed on a personal computer with 16 Gigabytes of RAM and 2.20 Gigahertz Intel Core i7 processor. The proposed models are tested using instances with variable number of nodes (|V|), links (|E|), and graph densities (p). For the three-phase decomposition, the instances ranged from 50 to 200 nodes, and 123 to 576 links. For the exact solution, network sizes ranged from 25 to 4941 nodes, and from 150 to 11535 links. Table II presents a brief summary of experimental results for the exact solution.

Table II
EXPERIMENTAL RESULTS OF EXACT MONITORING SCHEDULING

$\mid V \mid$	$\mid E \mid$	p	% Monitors	% Residual battery	Execution time (sec)
34	114	0.101	41	92.62	66.90
62	207	0.054	58	85.15	104.95
115	613	0.047	82	77.90	78.55
399	950	0.005	29	93.71	15.00
500	249500	0.990	99	78.96	276.60
600	179700	0.500	99	74.72	466.90
1589	4331	0.002	57	90.81	2.82
4941	11535	0.0005	47	94.00	111.24

Regarding the three-phase heuristic, analysis of experimental results reveals that modeling the monitoring placement as VCP guarantees full monitoring coverage. Our algorithm 1 in [6] is beneficial for reducing the computational time and realizing Fixed Parameter Tractability of VCP. The proposition in Phase II (cf. Table I) is able to optimally assign monitors to planning periods with minimum monitoring and communication energy consumption; depending on the problem's parameters (number of periods (|T|), energy loss per period (E_m), and $reserved\_battery_k$). The percentage of residual battery after monitoring, relaying the monitoring data to the BR and state transitions is more than 86% in all tested instances. Therefore, it is concluded that the BIP formulation was effective in minimizing the energy consumption. Results after solving the TSP Path (PhaseIII), for optimal sequencing of VCs across time periods, confirm both effectiveness and efficiency in reducing the state transitions; in some cases up to 80%.

It is interesting to emphasize that when the $reserved\_battery_k$ is relatively small, more VCs are required to monitor the same number of periods and scheduling for minimal energy consumption is critical. All of the mentioned conclusions are important for the adoption of network monitoring; particularly into mission-critical IoT applications.

The same conclusions are drawn for the exact model. The optimization ensured full network coverage and minimal energy consumption. The tested instances for this model were larger in size and density. Yet, the residual battery never fell below 74% in all instances. Looking at the execution time, it is concluded that the exact solution is efficiently computed for small-medium instances, while it can be time consuming for large-sized or dense networks. Nevertheless, it serves as a benchmark.

V. Future Research Directions

Since the IoT network topology is often dynamic, it is necessary to target heuristics and dynamic monitoring algorithms in our future work. Furthermore, it is necessary to test the effectiveness and efficiency of the proposed models against RPL's repair mechanisms. Implementing network monitoring by using RPL's DAGMC objects requires the help of a network simulator. We plan on using COOJA; the Contiki network simulator that targets constrained IoT networks.

VI. Acknowledgment

The author would like to thank her supervisors Miklos Molnar (University of Montpellier), Abderrahim Benslimane (Avignon University), Mohamed Saleh and Sally Kassem (Cairo University) for the continuous support. Also, she would like to thank the L'Oreal foundation and UNESCO for their award in the 2017 L'Oreal-UNESCO for Women in Science Levant and Egypt Fellowship. Last but not least, she thanks the French Ministry of Foreign Affairs, the Embassy of France in Egypt, and the French Institute in Egypt for their scholarship.

References

[1] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, "Transmission of ipv6 packets over ieee 802.15. 4 networks," Tech. Rep., 2007.

[2] Z. Sheng, S. Yang, Y. Yu, A. Vasilakos, J. Mccann, and K. Leung, "A survey on the ietf protocol suite for the internet of things: Standards, challenges, and opportunities," *IEEE Wireless Communications*, vol. 20, no. 6, pp. 91–98, 2013.

[3] S. Agrawal, K. Naidu, and R. Rastogi, "Diagnosing link-level anomalies using passive probes," in *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE.* IEEE, 2007, pp. 1757–1765.

[4] S. Balaji, V. Swaminathan, and K. Kannan, "Optimization of unweighted minimum vertex cover," *World Academy of Science, Engineering and Technology*, vol. 43, pp. 716–729, 2010.

[5] H. Moser, "Exact algorithms for generalizations of vertex cover," *Master's thesis, Fakultät für Mathematik und Informatik, Friedrich-Schiller-Universität Jena*, 2005.

[6] B. Mostafa, A. Benslimane, E. Boureau, M. Molnar, and M. Saleh, "Distributed monitoring in 6lowpan based internet of things," in *2016 International Conference on Selected Topics in Mobile Wireless Networking (MoWNeT)*, April 2016, pp. 1–7.

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

[8] B. Mostafa, A. Benslimane, M. Saleh, S. Kassem, and M. Molnar, "An energy-efficient multiobjective scheduling model for monitoring in internet of things," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1727–1738, June 2018.

[9] J.-P. Vasseur, M. Kim, K. Pister, N. Dejean, and D. Barthel, "Routing metrics used for path calculation in low-power and lossy networks," Tech. Rep., 2012.