Selçuk-Teknik Dergisi *ISSN 1302-6178* *Journal of Selcuk-Technic*
Özel Sayı 2020 (ICAT'20) *Special Issue 2020 (ICAT'20)*

VERTEX COVER BASED LINK MONITORING TECHNIQUES FOR WIRELESS SENSOR NETWORKS

Zuleyha AKUSTA DAGDEVIREN[1,+]
[1]International Computer Institute Ege University, Izmir Turkey
zuleyhaakusta@gmail.com

Abstract

Wireless sensor networks (WSNs) are generally composed of numerous battery-powered tiny nodes that can sense from the environment and send this data through wireless communication. WSNs have wide range of application areas such as military surveillance, healthcare, miner safety, and outer space exploration. Inherent security weaknesses of wireless communication may prone WSNs to various attacks such as eavesdropping, jamming and spoofing. This situation attracts researchers to study countermeasures for detection and prevention of these attacks. Graph theory provides a very useful theoretical basis for solving WSN problems related to communication and security issues. One of the important graph theoretic structures is vertex cover (VC) in which a set of nodes are selected to cover the edges of the graph where each edge is incident to at least one node in VC set. Finding VC set having the minimum cardinality for a given graph is an NP-hard problem. In this paper, we describe VC algorithms aiming link monitoring where nodes in VC are configured as secure points. We investigate variants of VC problems such as weight and capacity constrained versions on different graph types to meet the energy-efficiency and load-balancing requirements of WSNs. Moreover, we present clustering and backbone formation operations as alternative applications of different VC infrastructures. For each VC sub-problem, we propose greedy heuristic based algorithms.

Keywords: Wireless Sensor Networks, Link Monitoring, Graph Theory, Vertex Cover, NP-Hard Problem.

[+] This paper has been presented at the ICAT'20 (9th International Conference on Advanced Technologies) held in Istanbul (Turkey), August 10-12, 2020.

KABLOSUZ SENSÖR AĞLARI İÇİN KÖŞE ÖRTME TABANLI BAĞLANTI İZLEME TEKNİKLERİ

Özet

Kablosuz sensor ağlar (KSAlar) genellikle ortamdan algılayabilen ve bu verileri kablosuz iletişim yoluyla gönderebilen pille çalışan çok sayıda küçük düğümden oluşur. KSAlar askeri gözetim, sağlık hizmetleri, madenci güvenliği ve uzay keşfi gibi çok çeşitli uygulama alanlarına sahiptir. Kablosuz iletişimin doğasında var olan güvenlik zayıflıkları, KSAları gizli dinleme, sinyal bozma ve sahtekarlık gibi çeşitli saldırılara eğilimli hale getirebilmektedir. Bu durum, araştırmacıları bu saldırıların tespiti ve önlenmesine yönelik karşı önlemleri incelemeye yöneltmektedir. Çizge teorisi, iletişim ve güvenlik sorunları ile ilgili KSA sorunlarını çözmek için çok yararlı bir teorik temel sağlar. Önemli çizge teorik yapılardan biri köşe örtmedir (KÖ), bu yapıda her bir kenarın KÖ kümesindeki en az bir düğüme bitişik olacak şekilde çizgenin tüm kenarlarını kapsayacak bir dizi düğüm seçilmektedir. Verilen bir çizge için en az elemana sahip KÖ kümesini bulmak NP-zor bir problemdir. Bu makalede, KÖdeki düğümlerin güvenli noktalar olarak yapılandırıldığı bağlantı izlemeyi amaçlayan KÖ algoritmaları açıklanmaktadır. KSAların enerji verimliliği ve yük dengeleme gereksinimlerini karşılamak için, farklı çizge yapılarında KÖ problemlerinin ağırlık ve kapasite kısıtlı versiyonları gibi çeşitli türleri çalışılmaktadır. Ayrıca kümeleme ve omurga oluşturma işlemlerini farklı KÖ altyapılarının alternatif uygulamaları olarak sunulmaktadır. Her KÖ alt problemi için, açgözlü sezgisel tabanlı algoritmalar önerilmektedir.

Anahtar Kelimeler: Kablosuz Sensör Ağları, Bağlantı İzleme, Çizge Teorisi, Kenar Örtme, NP-Zor Problem.

1. Introduction

Recent advances in processor, transceiver and sensor technologies have made wireless sensor networks (WSNs) possible to use in wide range of areas [1]-[9]. WSNs are composed of tiny sensor nodes which are able to achieve environmental sensing and wireless communication. Generally, sensor nodes have five main hardware components: a radio transceiver, a microcontroller, volatile/nonvolatile memories, a power supplier and various sensors to measure temperature, pressure, light, acceleration, chemical contaminant. These nodes execute in a distributed and autonomous manner to achieve predefined tasks. Healthcare, habitat monitoring, disaster relief, outer space exploration, military surveillance, miner safety, volcanic activity detection are some important applications of WSNs. Sink nodes provide a gateway mechanism between ordinary sensor nodes and internet to relay information to parties requesting this data. Since transmission range of ordinary sensor nodes are limited, multi-hop communication should be present in most real world scenarios. In this manner, the sensed data from the environment may be relayed by some intermediate sensor nodes to reach to the sink nodes. Although this mechanism is well-known and effective in terms of data communication, transmission is the dominant factor of energy consumption [10] so consuming energy wisely is very important to prolong the lifetime of WSN applications.

Inherent features of wireless communication introduce various vulnerabilities to WSNs. For example, since a sensor node's packets are physically transmitted to all neighbors in its transmission radio range, attacks such as eavesdropping, jamming and spoofing are possible if not precautions are taken. In this manner, before deploying the WSN applications, countermeasures for detection and prevention of these attacks should be studied in advance. One of countermeasures of these attacks is monitoring the network traffic of sensor nodes. When we place at least one secure point for each communication channel, whole network traffic can be monitored, logged and inspected to detect and prevent attacks. Although this strategy is very effective against attacks, configuring an ordinary node as a secure point will have various costs in terms of time and price. So, minimizing the number of secure points is of utmost importance for resource-efficient WSNs.

Graph theory is a collection of studies related to graphs and a graph is a structure corresponding to a set of objects where some pair of objects is in a sense related [11]-[13]. Graph theory provides a very useful theoretical basis to solve WSN problems related to communication and security issues [14]-[21]. In this manner, WSN is modeled as graph $G(V,E)$ where set V represents sensor nodes and E is the set of communication links. Vertex cover (VC) is one of the crucial graph theoretic structures which has various application areas such as deployment of road cameras for traffic control, dynamic detection of race condition in parallel programing and finding phylogenetic trees based on protein domain information. A VC is a set of nodes selected to cover the edges of the given graph where each edge is incident to at least one node in VC set. Finding VC set having the minimum cardinality for a given graph is an NP-hard problem in which designing an algorithm that guarantees the optimal solution in polynomial time is not possible.

In this paper, we describe VC algorithms for link monitoring in WSNs where nodes in VC are configured as secure points. We investigate variants of VC problems on different graph types to meet the energy-efficiency and load-balancing requirements of WSNs. For WSNs where energy distribution of nodes is uniform, basic VC structure is capable to handle link monitoring operations. When energy distribution is not uniform, selecting nodes with high energy for VC inclusion is very important to prolong network lifetime. This operation can be utilized by constructing weighted VC if weights are given inversely proportional to energies. Assigning links proportional to nodes' energies further provides energy consumption balance such that nodes with high energy will monitor more links than the other nodes. This strategy can be provided by implementing a capacitated VC structure where each node in VC is assigned a capacity value that is the maximum number of links it can monitor. To relay the data to the sink node, a virtual backbone should be constructed in advance. Connected VC structure can be built to operate as a virtual backbone in WSNs. In this paper, we provide related model and algorithmic foundations for aforementioned VC variants and propose greedy heuristics for each variant of VC problem.

The rest of this paper is organized as follows. Network model is given in Section 2. VC algorithms, weighted VC algorithms, capacitated VC algorithms and connected VC algorithms are studied in Section 3, Section 4, Section 5 and Section 6, respectively. Conclusions are drawn in Section 7.

2. Network Model

In this section, we describe the underlying network model, namely graph $G=(V,E)$, with corresponding definitions that we have used in this paper. Nodes have unique id that can be used for addressing them in our network. An example network model depicted as an undirected graph which is given in Figure 2.1 where $V=\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ and $E=\{(0,1), (0,2), (1,2), (1,3), (2,4), (2,5), (3,6), (3,7), (6,8), (7,9), (8,9)\}$. As aforementioned, generally one or more sink node is present in WSNs that is assigned to role of transferring data between the sensor nodes and the users. In the example graph in Figure 2.1, node 0 is the sink node that is filled with black. We use *n* and *m* literals for the total number of nodes in the graph and edge count of the graph, respectively. For example, the graph given in Figure 2.1, *n*=10 and *m*=11. n and m are very useful to formulate the time complexities of the algorithms. *N*(*u*) denotes the neighbors of node *u* that is the set of nodes that are in immediate transmission range of node *u*. In single-hop transmission mode, node 1 can only send message to nodes 0, 2 and 3 in Figure 2.1. Transmission range of node 1 is shown with a dotted circle. In undirected graph model, since edges are bidirectional, if node *u* can send a message to node *v*, the message can be sent in reverse direction from node *v* to node *u*.

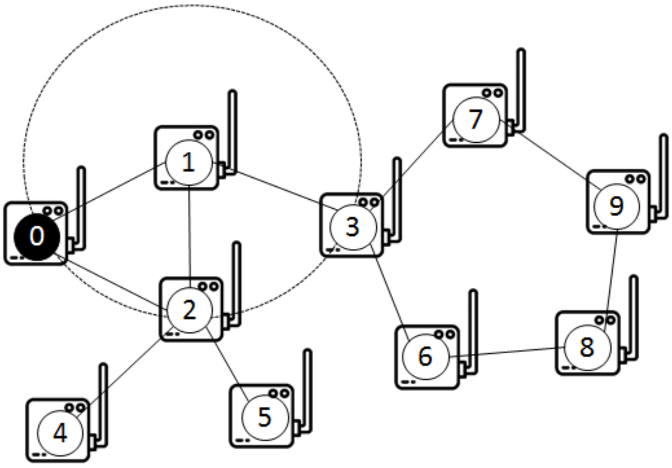

Figure 2.1 Network model

To show the degree of node *u*, the neighbor count of node *u* in other words, we use *d*(*u*). The maximum and minimum degrees of the network is denoted by Δ and δ, respectively. In the given example graph, *d*(1)=3, δ=1 and Δ=3. In weighted graphs, nodes and/or edges are assigned weights to represent various real world parameters. For example, a node's weight can be a function of its energy or its reliability. To denote the weight of node *u*, we use *w*(*u*). Diameter of the graph is the maximum distance between two nodes in the graph. In this paper, diameter of the graph is demonstrated with *D*. The diameter of the graph in Figure 2.1 is 5 where the distance between node 4 and node 9 is 5. In a similar manner, the distance between node 5 and node 9 is 5. To show the notations used throughout the paper, Table 2.1 presents a summary.

Table 2.1 Symbols and their meaning

Term	Meaning
G	Graph (network model)
V	Node set
E	Edge set
n	Node count
m	Edge count
$N(u)$	Neighbor set of node u
$d(u)$	Degree of node u
δ	Minimum degree of graph
Δ	Maximum degree of
$w(u)$	Weight of node u
D	Diameter of graph

3. Vertex Cover Algorithms

As aforementioned, VC provides a very suitable infrastructure for link monitoring in WSNs. An example VC structure is given in Figure 3.1. In this network nodes 1, 2, 3, 8 and 9 are in VC set. This set is minimal in terms of VC cardinality. Node u with a big $N(u)$ set, in other words having a high $d(u)$, can cover more links than other nodes. In this manner, for a given graph $G=(V,E)$, greedy VC algorithms generally include the node with Δ neighbors and exclude the node having δ neighbors to minimize VC set.

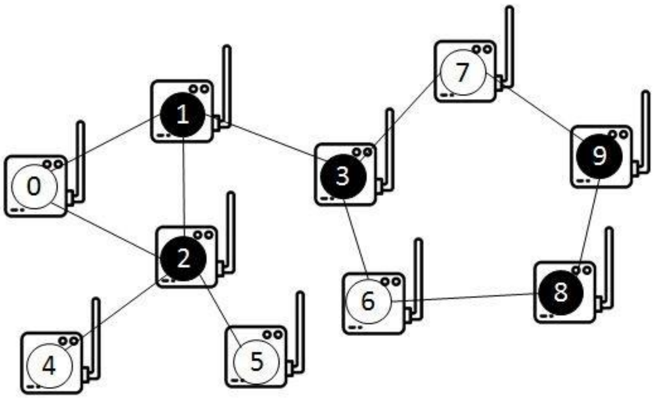

Figure 3.1 A VC on WSN

A heuristic algorithm named isolation algorithm has been proposed to find the minimum VC in Ugurlu's work [22]. The algorithm is tested on DIMACS benchmark graphs and BHOSLIB instances. The obtained measurements show that their algorithm can perform well in small size graph. Cai et al. introduce an algorithm that applies twostage exchange and edge weighting with forgetting strategies [23]. The first strategy selects two vertices to exchange and achieves this exchange in two stages. The latter strategy decreases weights for edges periodically. A local search algorithm is designed by the help of these strategies. The proposed approach is also implemented on DIMACS and BHOSLIB benchmarks. VC problem is studied for risk management and the main goal is formulated as a partial VC by Caskurlu et al. [24]. Bipartite graph model is used where VC problem in this model is in P complexity class. It is proved that partial version of VC problem is NP-hard.

A local search solver is proposed to utilize best-picking strategy with noise to remove vertices in [25]. The study claims that although best-picking strategy can have high complexity, it is very powerful for various problems. Authors conducted experiments on real-world massive graphs and show that their algorithm performs well. The VC problem with multiple coverage constraints for hypergraphs is considered by Hong et al. [26]. A primal-dual approximation algorithm is proposed in this manner. The algorithm is deterministic and pure combinatorial which means that no Ellipsoid solver is necessary. Another study concerning the VC problem on hypergraph structure is given by Cheung et al. [27].

A game-based memetic algorithm is given by Wu et al. [28] in which the global search and local search are achieved by an asynchronous updating snowdrift game and evolutionary algorithm, respectively. Degree-based initialization method is used in their algorithm. The local refinement is done by each vertex that is modeled as an intelligent agent, by playing the snowdrift game. The strict nash equilibrium of the snowdrift game is always a vertex cover of the given graph. Besides these reviewed algorithms, there are other studies given in [29], [30]. To solve the vertex cover problem for link monitoring, we propose a greedy algorithm that is given in Algorithm 3.1 where the *link_coverage* of node *i* is the number of uncovered edges incident to node *i*. The algorithm starts with the selection of the monitor node with the greatest *link_coverage*. Then, the incident uncovered edges (links) to monitor node are assigned as covered. These steps repeat until all links in graph are covered and at the end of the algorithm, the selected monitor nodes construct VC set.

Algorithm 3.1 Proposed Greedy VC Based Link Monitoring Algorithm

---

link_coverage: # of uncovered incident edges

```
Until all links are covered
    Select the node having the greatest link_coverage as a monitor
    Cover new monitor node's uncovered links
End
```

4. Weighted Vertex Cover Algorithms

The nodes in VC are assigned to link monitoring, so their energies are subjected to depletion. In order to avoid this problem, node energies are considered during weighted VC construction. An example weighted VC is given in Figure 4.1 where weight (1/energy) of each node is written near to it. Nodes 0, 2, 3, 6 and 9 are included in weighted VC set and the total weight of VC set is 150. Greedy weighted VC algorithms generally choose the nodes having low weights so node *u* with a low *w(u)* is preferable than the other nodes.

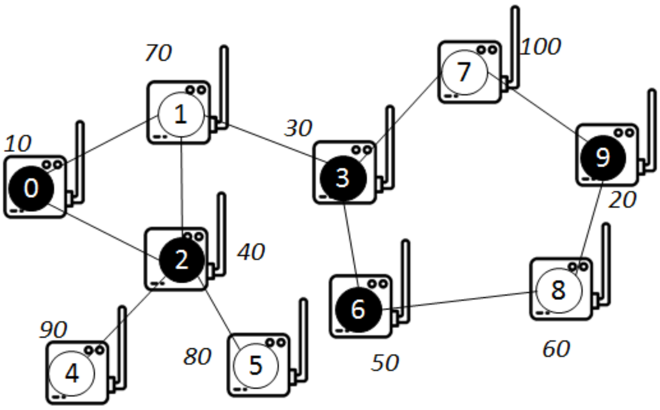

Figure 4.1 A weighted VC on WSN

In Xu et al.'s work [31], the authors propose a solver for weighted VC problem that uses primal-dual approximation algorithm. They show that their satisfiability testing based solver performs well. An algorithm based on list-heuristic for finding minimum weighted VC is given by Shimizu et al. [32]. The paper given in [33] studies how the dual formulation of weighted VC can be used. The authors show that their randomized local search process can obtain a 2-approximation. In Cai et al.'s work [34], the authors propose two strategies that are used to improve local search process for weighted VC problem. The authors evaluate their algorithms along with the others for map labeling problem. They also use massive graphs in their performance evaluations and obtain well results.

k-VC problem is to find a VC whose size does not exceed *k* nodes and *k*-weighted VC problem is to a find a VC whose total weight does not exceed *k*. The study given in [35] generalizes the Buss reduction to the *k*-weighted VC problem. They study its features on real-world datasets. In Islam et al.'s study [36], both vertex weights and edge weights are considered as the generalized vertex cover problem. The authors propose a chemical reaction optimization based algorithm by adding repair function and redesigning the other four operators. The proposed algorithm is compared with local search with tabu strategy and perturbation mechanism, genetic algorithm, hybrid metaheuristic algorithm and genetic algorithm. A parameterized algorithm for weighted VC on graphs having maximum degree 3 is given in Tsur's work [37].

The authors propose a local search that is based on three ideas in [38]. Firstly, in the initial phase, four reduction rules are designed. Second, for reducing cycling during search operation, configuration checking with aspiration strategy is proposed. To decrease the consumed time for search process, a self-adaptive vertex removing strategy is designed. The algorithm is test on standard benchmarks, real-world instances and massive graphs.

The study given by Pourhassan et al. [39] extends the analysis given by Kratsch's work [40] to the weighted VC problem in which the weights of the vertices are integers. The aim of the paper is finding a weighted VC that has the minimum total weight. The authors propose a fixed parameter evolutionary algorithm which uses an alternative mutation operator given in [40]. Besides, to keep the population size in polynomial order and to find a 2-approximation, a multiobjective evolutinary algorithm is presented. For a study related minimum weight *k*-path cover problem, that is a generalization of weighted VC, please refer to [41].

Algorithm 4.1 Proposed Greedy Weighted VC Based Link Monitoring Algorithm

```
Until all links are covered
    Select the node having the minimum weight as a monitor
    Cover new monitor node's uncovered links
End
```

For an energy-efficient link monitoring solution, we propose a greedy heuristic based algorithm given in Algorithm 4.1. The node with the minimum weight in graph is selected as monitor node initially. Then, uncovered links incident to the monitor node are signed as covered. The selection continues among the remaining nodes. These processes are executed until all links are covered.

5. Capacitated Vertex Cover Algorithms

Node energies are considered during weighted VC construction whereas assignment of links to VC set is not considered. In capacitated VC, links are assigned to the nodes in VC set where the maximum link count that can be assigned to a monitor node is bounded by a predefined capacity value. An example capacitated VC structure having capacity value as 3 is depicted in Figure 5.1 where link (0,2) is assigned to node 0, links (0,1) and (1,3) are assigned to node 1, links (1,2), (2,4) and (2,5) are assigned to node 2, links (3,6) and (3,7) are assigned to node 3, link (6,8) is assigned to node 8, links (7,9) and (8,9) are assigned to node 9.

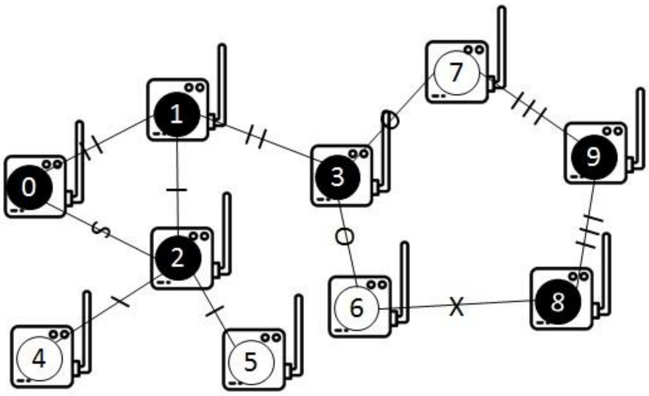

Figure 5.1 A capacitated network model

Capacitated VC algorithms for multigraphs and hypergraphs assuming hard capacities are given in Wong's study [42]. The approaches are based on applying iterative rounding to a LP relaxation. A soft capacitated VC algorithm in a dynamic setting is given in Wei et al.'s work [43]. In this setting, vertex weights can change as well as edge insertions and deletions are allowed. The studies in [44], [45] is extended to obtain a deterministic primal-dual algorithm having constant factor approximation for minimum capacitated VC problem. The authors show that the proposed algorithm can be extended to a more general model where edges are given with unsplittable and non-uniform demand. In Rooij's et al.'s study [46], NP-completeness proofs of capacitated VC problem on special cases of general graphs such as modular graphs, tree-convex graphs and planar bipartite graphs having maximum degree three is given.

Algorithm 5.1 Proposed Greedy Capacitated VC Based Link Monitoring Algorithm

```
link_coverage: # of uncovered incident edges
k: the maximum link count can be assigned to a monitor node

1:  Until all links are covered
2:      Select the node having the greatest link_coverage as a monitor
3:      Cover new monitor node's up to k uncovered links randomly
4:  End
```

The capacitated version of VC problem for link monitoring can be solved by our proposed greedy approach based algorithm given in Algorithm 5.1 where *k* is the capacity value. The algorithm is very similar to Algorithm 3.1. The only difference is that instead of all uncovered links, up to *k* links can be covered and these links are selected randomly.

6. Connected Vertex Cover Algorithms

If the vertex induced subgraph of nodes in VC set is connected, then we call the structure as Connected VC. In another words, for a connected VC, each node pair in VC set has a connecting path only consisting of nodes in VC. In this VC type, nodes in VC set can form a virtual backbone since monitor nodes are connected. In this manner, connected VC is very important structure since it provides link monitoring and backbone formation at the same time. Figure 6.1 shows an example connected VC where nodes 1, 2, 3, 6, 7 and 9 are in this set. Besides other graph parameters, *D* can be an important factor effecting the connected VC size.

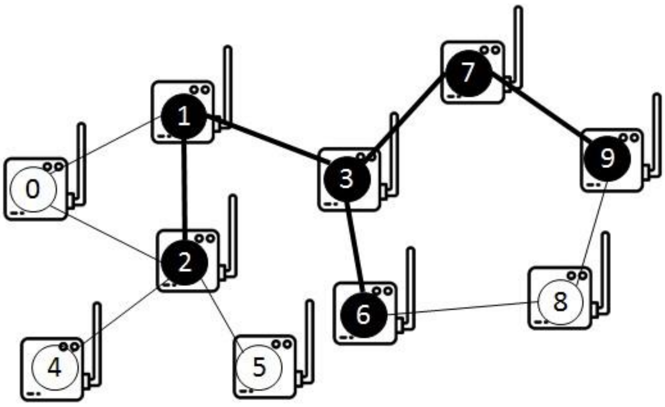

Figure 6.1 A connected VC on WSN

Li et al. study on the connected VC problem for 4-regular graphs which proves that the problem is still NP-hard in this case [47] and they propose two approximation algorithms. In Krithika et al.'s work [48], parameterized complexity statuses of connected VC are investigated. Other studies for connected VC problem for special cases of graphs can be found in [49]. Zhang et al. propose an algorithm which has two phases: construction phase and local search phase [50]. A greedy function and a restricted candidate list are designed to construct a good initial solution in the construction phase. To decrease the cycling problem, the configuration checking strategy is adopted in the latter phase. Li et al. [51] investigate the connected VC problem for *k*-regular graphs for any constant $k \geq 4$. They first prove that the problem in this case is NP-hard, then they propose an approximation algorithm to solve the problem. To construct a connected VC structure that can be used for link monitoring or backbone formation, we propose a greedy heuristic based algorithm given in Algorithm 6.1 where *S* represents the set of nodes not in VC and having at least one neighbor in VC. The first step of the algorithm is the selection of the node with the greatest *link_coverage* as monitor node. The uncovered links incident to the monitor are assigned as covered. The algorithm provides connectivity by selecting the subsequent monitor nodes from the neighbors of the selected ones.

Algorithm 6.1 Proposed Greedy Connected VC Based Link Monitoring and Backbone Formation Algorithm

```
link_coverage: # of uncovered incident edges
S: set of nodes not in VC and having at least one neighbor in VC
---
1: Select the node having the greatest link_coverage as a monitor
2: Cover monitor node's uncovered links
3: Until all links are covered
4:     Select the node having the greatest link_coverage from S as a monitor
5:     Cover new monitor node's uncovered links
6:   End
```

7. Conclusions

As a result of calculations and analyses performed, it is concluded as follows: Recent advancements in wireless communication and processor technologies lead to utilize WSNs for wide range of critical application areas such as military surveillance, miner safety and disaster relief. Sensor nodes can sense events from the environment and generally relay this data through the sink node in a multi-hop fashion. Since communication is the most important factor in energy consumption, the algorithms should be designed considering this fact. WSNs are vulnerable to attacks such as eavesdropping, jamming and spoofing since wireless communication has inherent security flaws. So that, designing countermeasures to detect and prevent these attacks is of paramount importance. Monitoring links by deploying secure points can provide to defend these attacks where whole network traffic can be logged and inspected. Although this strategy solves the problem, since secure points have their own cost, the secure point count should be minimized for total cost reduction in terms time and price.

Graph theory plays a critical role as a theoretical basis to solve various WSN communication and security problems. VC is one of the fundamental graph theoretic structures that has various applications in bioinformatics and computer science where it is a set of nodes to cover the edges of the graph. Finding minimum VC is an NP-hard problem, so designing polynomial time efficient algorithms is an important research thread. In this paper, we describe VC algorithms for link monitoring in WSNs where nodes in VC are defined as secure points. Standard VC structure without any extra constraint is able to meet the needs of link monitoring operations when the energies of sensor nodes are nearly equal. We show that weighted VC can be used when energy distribution of sensor nodes is not uniform and capacitated VC can further provide energy-efficiency by assigning links to nodes considering their energies. We also present that connected VC structure can provide both link monitoring and backbone formation at the same time. Moreover, we propose simple greedy algorithm based link monitoring algorithms to solve each variant of vertex cover problem.

References

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks,* vol. 38, no. 4, pp. 393 – 422, 2002.

[2] M. Tubaishat and S. Madria, "Sensor networks: an overview," Potentials, IEEE, vol. 22, no. 2, pp. 20–23, 2003.

[3] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292 – 2330, 2008.

[4] V. Potdar, A. Sharif, and E. Chang, "Wireless sensor networks: A survey," in *2009 International Conference on Advanced Information Networking and Applications Workshops*, 2009, pp. 636–641.

[5] A. Hadjidj, M. Souil, A. Bouabdallah, Y. Challal, and H. Owen, "Wireless sensor networks for rehabilitation applications: Challenges and opportunities," *Journal of Network and Computer Applications,* vol. 36, no. 1, pp. 1 – 15, 2013.

[6] P. Rawat, K. D. Singh, H. Chaouchi, and J. M. Bonnin, "Wireless sensor networks: A survey on recent developments and potential synergies," *J. Supercomput.*, vol. 68, no. 1, p. 1–48, Apr. 2014.

[7] B. Rashid and M. H. Rehmani, "Applications of wireless sensor networks for urban areas: A survey," *Journal of Network and Computer Applications*, vol. 60, pp. 192 – 219, 2016.

[8] S. R. Jino Ramson and D. J. Moni, "Applications of wireless sensor networks — a survey," in *2017 International Conference on Innovations in Electrical, Electronics,*

Instrumentation and Media Technology (ICEEIMT), 2017, pp. 325–329.

[9] R. E. Mohamed, A. I. Saleh, M. Abdelrazzak, and A. S. Samra, "Survey on wireless sensor network applications and energy efficient routing protocols," *Wirel. Pers. Commun.*, vol. 101, no. 2, p. 1019–1055, Jul. 2018.

[10] H. Karl and A. Willig, *Protocols and Architectures for Wireless Sensor Networks*. Hoboken, NJ, USA: John Wiley Sons, Inc., 2005.

[11] J. L. Gross and J. Yellen, *Graph Theory and Its Applications, Second Edition (Discrete Mathematics and Its Applications)*. Chapman Hall/CRC, 2005.

[12] J. Bondy and U. Murty, *Graph Theory*, 1st ed. Springer Publishing Company, Incorporated, 2008.

[13] D. B. West, *Introduction to graph theory*, 2nd ed. Prentice Hall Of India Pvt.Ltd., 2011.

[14] Jingbo Dong, Qing Chen, and Zhisheng Niu, "Random graph theory based connectivity analysis in wireless sensor networks with rayleigh fading channels," in *2007 Asia-Pacific Conference on Communications*, 2007, pp. 123–126.

[15] L. Ding and Z.-H. Guan, "Modeling wireless sensor networks using random graph theory," *Physica A: Statistical Mechanics and its Applications*, vol. 387, no. 12, pp. 3008 – 3016, 2008.

[16] N. Nasser and L. Arboleda, *Clustering in Wireless Sensor Networks: A Graph Theory Perspective*, 03 2008, pp. 161 – 193.

[17] H. Kenniche and V. Ravelomananana, "Random geometric graphs as model of wireless sensor networks," in *2010 The 2nd International Conference on Computer and Automation Engineering (ICCAE)*, vol. 4, 2010, pp. 103–107.

[18] D. Sarioz, "Geometric graph theory and wireless sensor networks," Ph.D. dissertation, USA, 2012.

[19] K. Gao, P. Suganthan, Q.-K. Pan, M. Tasgetiren, and A. Sadollah, "Artificial bee colony algorithm for scheduling and rescheduling fuzzy flexible job shop problem with new job insertion," *Knowledge-Based Systems*, vol. 109, 06 2016.

[20] H. Rhim, T. Karim, R. Abassi, D. Sauveron, and S. Guemara, "A multi-hop graph-based approach for an energy- efficient routing protocol in wireless sensor networks," *Human-centric Computing and Information Sciences*, vol. 8, 12 2018.

[21] M. Z. Masoud, Y. Jaradat, I. Jannoud, and M. A. A. Sibahee, "A hybrid clustering

routing protocol based on machine learning and graph theory for energy conservation and hole detection in wireless sensor network," *International Journal of Distributed Sensor Networks*, vol. 15, no. 6, p. 1550147719858231, 2019.

[22] U. O., "New heuristic algorithm for unweighted minimum vertex cover," *Problems of Cybernetics and Informatics*, pp. 1–4, 2012.

[23] S. Cai, K. Su, C. Luo, and A. Sattar, "Numvc: An efficient local search algorithm for minimum vertex cover," *J. Artif. Int. Res.*, vol. 46, no. 1, p. 687–716, Jan. 2013.

[24] B. Caskurlu, V. Mkrtchyan, O. Parekh, and K. Subramani, "On partial vertex cover and budgeted maximum coverage problems in bipartite graphs." in *IFIP TCS*, ser. Lecture Notes in Computer Science, J. Díaz, I. Lanese, and D. Sangiorgi, Eds., vol. 8705. Springer, 2014, pp. 13–26.

[25] Z. Ma, Y. Fan, K. Su, C. Li, and A. Sattar, "Local search with noisy strategy for minimum vertex cover in massive graphs," in *Proceedings of the 14th Pacific Rim International Conference on Trends in Artificial Intelligence*, ser. PRICAI'16. Springer, 2016, p. 283–294.

[26] E. Hong and M.-J. Kao, "Approximation Algorithm for Vertex Cover with Multiple Covering Constraints," in *29th International Symposium on Algorithms and Computation (ISAAC 2018)*, ser. Leibniz International Proceedings in Informatics (LIPIcs), W.-L. Hsu, D.-T. Lee, and C.-S. Liao, Eds., vol. 123. Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2018, pp. 43:1–43:11.

[27] C. W. Cheung, X. M. Goemans, and C.-w. S. Wong, "Improved algorithms for vertex cover with hard capacities on multigraphs and hypergraphs," *SODA*, pp. 1714–1726, 2014.

[28] J. Wu, X. Shen, and K. Jiao, "Game-based memetic algorithm to the vertex cover of networks," *IEEE Transactions on Cybernetics*, vol. 49, no. 3, pp. 974–988, 2019.

[29] C. Witt, "Analysis of an iterated local search algorithm for vertex cover in sparse random graphs," *Theoretical Computer Science*, vol. 425, pp. 117 – 125, 2012, theoretical Foundations of Evolutionary Computation.

[30] A. Bazzi, S. Fiorini, S. Pokutta, and O. Svensson, "No small linear program approximates vertex cover within a factor 2 – e," in *Proceedings of the 2015 IEEE 56th Annual Symposium on Foundations of Computer Science (FOCS)*, ser. FOCS '15. USA: IEEE Computer Society, 2015, p. 1123–1142.

[31] H. Xu, T. K. S. Kumar, and S. Koenig, "A new solver for the minimum weighted vertex cover problem," in *Proceedings of the 13th International Conference on Integration of Artificial Intelligence and Operations Research Techniques in Constraint Programming (CPAIOR)*, 2016, pp. 392–405.

[32] S. Shimizu, K. Yamaguchi, T. Saitoh, and S. Masuda, "A fast heuristic for the minimum weight vertex cover problem," *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, pp. 1–5, 2016.

[33] M. Pourhassan, T. Friedrich, and F. Neumann, "On the use of the dual formulation for minimum weighted vertex cover in evolutionary algorithms," in *Proceedings of the 14th ACM/SIGEVO Conference on Foundations of Genetic Algorithms*, ser. FOGA '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 37–44.

[34] S. Cai, W. Hou, J. Lin, and Y. Li, "Improving local search for minimum weight vertex cover by dynamic strategies," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*. International Joint Conferences on Artificial Intelligence Organization, 7 2018, pp. 1412–1418.

[35] H. Xu, X.-Z. Wu, C. Cheng, S. Koenig, and T. K. S. Kumar, "The buss reduction for the k-weighted vertex cover problem," in *ISAIM*, 2018.

[36] M. R. Islam, I. Arif, and R. Shuvo, "Generalized vertex cover using chemical reaction optimization," *Applied Intelligence*, vol. 49, 01 2019.

[37] D. Tsur, "Weighted vertex cover on graphs with maximum degree 3," *CoRR*, vol. abs/1810.12982, 2018.

[38] R. Li, S. hu, S. Cai, J. Gao, Y. Wang, and M. Yin, "Numwvc: A novel local search for minimum weighted vertex cover problem," *Journal of the Operational Research Society*, pp. 1–12, 06 2019.

[39] M. Pourhassan, F. Shi, and F. Neumann, "Parameterized analysis of multiobjective evolutionary algorithms and the weighted vertex cover problem," *Evolutionary Computation*, vol. 27, no. 4, pp. 559–575, 2019.

[40] S. Kratsch and F. Neumann, "Fixed-parameter evolutionary algorithms and the vertex cover problem," in *Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation*, ser. GECCO '09. New York, NY, USA: Association for Computing Machinery, 2009, p. 293–300.

[41] Y. Zhang, Y. Shi, and Z. Zhang, "Approximation algorithm for the minimum

weight connected k-subgraph cover problem," *Theoretical Computer Science*, vol. 535, pp. 54 – 58, 2014.

[42] S. C.-w. Wong, "Tight algorithms for vertex cover with hard capacities on multigraphs and hypergraphs," in *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA '17. USA: Society for Industrial and Applied Mathematics, 2017, p. 2626–2637.

[43] H.-T. Wei, W.-K. Hon, P. Horn, C.-S. Liao, and K. Sadakane, "An O(1)-Approximation Algorithm for Dynamic Weighted Vertex Cover with Soft Capacity," in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2018)*, ser. Leibniz International Proceedings in Informatics (LIPIcs), E. Blais, K. Jansen, J. D. P. Rolim, and D. Steurer, Eds., vol. 116. Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2018, pp. 27:1–27:14.

[44] S. Bhattacharya, M. Henzinger, and G. F. Italiano, "Deterministic fully dynamic data structures for vertex cover and matching," in *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA '15. USA: Society for Industrial and Applied Mathematics, 2015, p. 785–804.

[45] S. Bhattacharya, M. Henzinger, and G. F. Italiano, "Design of dynamic algorithms via primal-dual method," *CoRR*, vol. abs/1604.05337, 2016.

[46] S. B. van Rooij and J. M. M. van Rooij, "Algorithms and complexity results for the capacitated vertex cover problem," in *SOFSEM 2019: Theory and Practice of Computer Science*, B. Catania, R. Královič, J. Nawrocki, and G. Pighizzini, Eds. Cham: Springer International Publishing, 2019, pp. 473–489.

[47] Y. Li, Z. Yang, and W. Wang, "Complexity and algorithms for the connected vertex cover problem in 4-regular graphs," *Applied Mathematics and Computation*, vol. 301, pp. 107 – 114, 2017.

[48] R. Krithika, D. Majumdar, and V. Raman, "Revisiting connected vertex cover: FPT algorithms and lossy kernels," *CoRR*, vol. abs/1711.07872, 2017.

[49] M. Johnson, G. Paesani, and D. Paulusma, *Connected Vertex Cover for (sP1+P5)-Free Graphs*, 09 2018, pp. 279–291.

[50] Y. Zhang, J. Wu, L. Zhang, P. Zhao, J. Zhou, and M. Yin, "An efficient heuristic algorithm for solving connected vertex cover problem," *Mathematical Problems in*

Engineering, vol. 2018, pp. 1–10, 09 2018.

[51] Y. Li, W. Wang, and Z. Yang, "The connected vertex cover problem in k-regular graphs," *Journal of Combinatorial Optimization*, vol. 38, pp. 635–645, 03 2019.