BİLGİSAYAR BİLİMLERİ VE TEKNOLOJİLERİ DERGİSİ
JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGIES
http://dergipark.org.tr/tr/pub/bibted

Araştırma Makalesi

Positioning Security Cameras in The Central Transportation Networks of Barcelona With Minimum Cost via The Malatya Minimum Vertex Cover Algorithm

Cemalettin Sonakalan[*1], **Furkan Öztemiz**[1]
[1]*Inonu University, Engineering Faculty, Software Engineering Department, Malatya, Türkiye*

Keywords:
Minimum Vertex Cover
Security Cameras
Graph Theory
Malatya Centrality

ABSTRACT

The Minimum Vertex Cover issue (MVCP) is a significant NP-complete optimization issue in graph theory. Its objective is to find a set of nodes that covers all edges of a given graph and contains the minimum number of nodes. Many different approaches and algorithms have been tried for this issue. Nevertheless, as the MVCP problem is an optimization problem, solutions are usually non-heuristic and only work under certain constraints Moreover, the proposed methods do not achieve the expected effect and the solution sets may change with each iteration. Having a minimum number of nodes in a network with a minimum coverage area improves network efficiency, reduces energy consumption, and allows for more efficient resource utilization. This study aims to control all streets in a popular neighborhood in Barcelona with a minimum number of security cameras. The Malatya Vertex Cover method is used to locate the optimal number of security cameras around the area. For modeling, the area is transformed into a graph using Google Earth. Each intersection represents a node. The graph was modeled using R programming language. Then, with the Malatya Vertex Cover algorithm, the Malatya centrality values of the nodes of the graph will be calculated. This centrality value is obtained from the sum of the ratio of the degree of each node to the degree of its neighbors. For the MVCP solution, the node of the graph with the highest Malatya centrality value is selected and added to the solution set. Then, this node and its edge links are removed from the graph. When the edges are completely covered, the process is terminated. As a result of this analysis, a low-cost solution is achieved by using the minimum number of security cameras to cover the entire region.

Malatya Minimum Vertex Cover Algoritması ile Barselona'nın Merkezi Ulaşım Ağlarında Güvenlik Kameralarının Minimum Maliyetle Konumlandırılması

Anahtar Kelimeler:
Minimum Köşe Örtüsü
Güvenlik Kameraları
Çizge Teorisi
Malatya Merkezilik

ÖZ

Minimum Vertex Cover Problemi (MVCP), çizge teorisinde önemli bir NP-complete optimizasyon problemidir. Amacı, verilen bir grafın tüm kenarlarını kapsayan ve en az sayıda düğüm içeren bir düğüm kümesini bulmaktır. Bu problem için birçok farklı yaklaşım ve algoritma denenmiştir. Ancak MVCP problemi bir optimizasyon problemi olduğundan, çözümler genellikle sezgisel olmayıp belirli kısıtlamalar altında sonuç vermektedir. Bir ağda düğümlerin en az sayıda kapsanması ağın verimliliğini yükseltir, enerji tüketimini düşürür ve kaynakların daha verimli kullanılmasını sağlar. Bu çalışma Barcelona şehrinde popüler bir muhitteki cadde ve sokakların tümünü en az sayıda güvenlik kamerasıyla kontrol edilmesini hedefler. Bölge Google Earth kullanılarak çizgeye uygun modellenmiştir. Her bir kavşak bir düğümü temsil etmektedir. R programlama dili kullanılarak çizge oluşturulmuştur. Ardından Malatya Vertex Cover algoritmasıyla çizgenin düğümlerinin Malatya merkezilik değerleri hesaplanacaktır. Bu merkezilik değeri her bir düğümün derecesinin, komşularının derecesine oranının toplamından elde edilmektedir. MVCP çözümü için ise çizgenin en yüksek Malatya merkezilik değerine sahip olan düğümü seçilerek çözüm kümesine eklenir. Sonrasında bu düğüm ve kenar bağlantıları çizgeden çıkarılır. Kenarlar tamamen kapsandığında, işlem sonlandırılır. Bu analiz sonucunda tüm bölgeyi kapsayacak şekilde en az sayıda güvenlik kamerası kullanarak düşük maliyetli çözüm sağlanmıştır.

[*]**Cemalettin Sonakalan**

[*](yusufcs1234@gmail.com) ORCID ID 0009-0002-1391-3485
(furkan.oztemiz@inonu.edu.tr) ORCID ID 0000-0001-5425-3474

e-ISSN: 2717-8579

Geliş Tarihi: 08/09/2024; Kabul Tarihi: 31/12/2024

1. INTRODUCTION

A graph is a fundamental data structure consisting of nodes and edges(Thulasiraman et al., 2011). This form is commonly used in practical situations and the field of computer science(Hark and Karcı, 2022). The strategies and techniques applied to graphs facilitate the development of solutions and approaches in these domains. Nevertheless, graphs encompass numerous NP-complete problems(Thulasiraman et al, 2016). Among these, the node covering problem stands out as a key issue. This problem involves identifying which nodes should be selected to ensure that every edge in the graph is covered(Thulasiraman et al., 2011). When the objective is to cover all edges with the fewest possible nodes, this NP-complete problem is referred to as the minimum vertex covering problem (Khattab et al., 2022).

Minimum Vertex Cover Problem (MVCP) is frequently used in computer science to model today's problems (Angel, 2022) and (Wang, 2017). For this reason, the algorithms and solutions proposed for MVCP can be utilized in various applications (Dagdeviren, 2021). For example, the use of MVCP is important in various fields such as the detection of nodes for the protection of healthcare data (Angel, 2022), the generation of non-repetitive static genetic sequences (Hossain et al., 2020), the management of traffic, which is a common problem in today's world (Gusev, 2020), and the optimized use and positioning of hardware with limited facilities (Dagdeviren, 2021). In addition, security applications in wireless networks, routing and monitoring in wireless mobile networks (Yigit et al. 2022 and Yigit et al., 2021), link monitoring and formation in ad hoc networks (Dagdeviren, 2016), etc. are also areas where MVCP has been addressed. However, because the Minimum Vertex Cover is an NP-Complete optimization problem, it usually cannot be solved within an polinomial period. Therefore, exact solutions are presented for relatively simpler graphs or specified graphs. Various methods and approaches have been proposed to solve MVCP, but there is no method that produces optimal solutions to MVCP in polynomial time. Usually, after a certain number of iterations or processing steps, a set of possible solutions close to the optimal solution is identified. However, changing initial conditions and the characteristics of the method used may prevent the MVCP from determining the solution set. Real-world applications make solving the MVCP even more difficult due to the complexity and size.

In this work, Malatya Vertex Cover Algorithm(MVCA) is used to efficiently handle an NP-complete problem like MVCP. This algorithm calculates the Malatya centrality score for each node. When calculating this value, the node's own degree is taken into account as well as the degrees of its neighboring nodes. Then, Malatya centrality determines which nodes to select. The node with the highest MC value is selected and added to the solution set. Then, this node and its associated edges are removed from the graph. For the newly formed graph, the MVCA is applied again and new Malatya centralization scores are calculated. The operations on the graph continue until the vertex cover solution set is found. MVCA is a robust method with an efficient greedy approach. Thanks to the stable solution sets it produces, it can be used in real-time systems and large graphs.

1.1. Motivation To Work

Our contributions to the paper:

1. The main motivation of this study is to provide a high level of security in a city like Barcelona, which hosts millions of tourists every year, and to reduce the current cost and provide energy efficiency while providing this security.

2. As a result of this study, many touristic destinations in the world will be approached with a similar approach to this study and the concerns about the security problem will be eliminated. And the Barcelona region selected for analysis was prepared specifically for this study and a graph model was created. One of the most unique parts of the study is the Barcelona network developed specifically for this study.

3. In addition, Malatya Vertex Cover Algorithm was applied for the first time in the Barcelona region, which is a unique transportation network not previously available in the literature. Since the algorithm is an efficient method that provides optimum or near optimum solutions in polynomial time, successful results were obtained.

4. MVCA is applied to transportation networks for the first time in this study and the results are analyzed.

2. RELATED WORK

Since the MVCP is a popular graph theoretic problem, many methods with various approaches have been developed.

Khattab, Mahafzah, and Sharieh developed a hybrid algorithm using a combination of chemical reaction optimization and best-first search algorithms to solve the Minimum Vertex Cover problem, this approach provides remarkable performance improvements on large-scale problems (Khattab, 2022). Dinur and Safra analyzed the difficulty of approximate solutions to the Vertex Cover problem, showing that this problem is NP-hard and the theoretical limits of approximating the optimal solution, thus providing an important perspective on the effectiveness of algorithms(Dinur, 2005). Wang et al.vdeveloped a Polynomial Time Approximation Scheme (PTAS) for the Minimum Weighted Connected Vertex Cover problem in three-dimensional wireless sensor networks(Wang, 2017). Öztemiz and Karcı provide concrete data on how the Vertex Cover problem can be applied on real-world transportation networks by analyzing the centrality of intersections in the transportation network of Malatya province(Öztemiz and Karci, 2021). (Angel, 2022) used graph theory approaches to protect medical information systems against cyber-attacks and showed how the Vertex Cover problem can be applied to the development of security strategies. Again, the research by (Yigit and Dagdeviren, 2022), and Challenger made significant contributions in improving network security and efficiency by investigating Vertex Cover algorithms with self-stabilizing capacity in IoT-based wireless sensor networks (Yigit and Dagdeviren, 2022).

Khattab, Sharieh, and Mahafzah improved the efficiency and accuracy of the algorithm by developing the Most Valuable Player (MVP) algorithm to solve the Minimum Vertex Cover problem (Khattab et al., 2019). Guo, Quan, and Chen achieved significant success in solving the Minimum Vertex Cover problem using the Membrane Evolutionary Algorithm (MEAMVC), which aims to achieve a solution by mimicking biological evolutionary processes(Guo, 2019). Xie et al. in their work called Test-cost-sensitive rough set based approach, addressed the Minimum Weight Vertex Cover problem and developed more effective solutions with this method, especially considering cost sensitivity (Xie, 2018). Jovanovic, Sanfilippo, and Voß used the Fixed Sets Search algorithm for the Multi-objective Minimum Weighted Vertex Cover problem and showed that this method is effective in multi-objective optimization problems (Jovanovic, 2022). Li et al. developed a new local search algorithm called NuMWVC to solve the Minimum Weighted Vertex Cover problem and proved that this method is especially effective for large graphs(Li et al., 2019).

These works provide innovative algorithms and theoretical analyses then developing solution for the Minimum Vertex Cover problem, providing significant advances in both academic research and practical applications. The analysis of various optimization techniques, security applications, and real-world networks makes it possible to address the Vertex Cover problem with a multifaceted approach. In particular, innovative methods such as chemical reaction optimization, evolutionary algorithms and local search techniques have made great progress in solving the Vertex Cover problem. In this context, the diversity and depth of research provides a broader perspective on the solution of the Vertex Cover problem.

It can be said that the methods developed for solving MVCP generally have exac, heuristic and greedy approaches. Exac methods aim for the optimum solution and usually all possibilities are tried. Therefore, it is not possible to produce solutions for large graphs. Heuristic methods run more than one iteration and can produce different results in each run. Greedy methods usually use a priority value for node selection and proceed by selecting the node that they think is the best in the current situation. Malatya Vertex Cover algorithm also produces deterministic results with an efficient greedy approach.

3. MATERIAL AND METHOD

There are two important aspects of the presented work. First, Malatya Vertex Cover algorithm is joined to transportation networks for the first time then analyzed. The second is the network selected for analysis. The region analyzed in the study is the center of the city of Barcelona. The reason why Barcelona was chosen is that it is a popular city known to many people and has a developed transportation network.In the following sections, detailed information about both the graph and the methodology is given. Figure 1 shows a visualization of the process of transforming the transportation network of the city of Barcelona into a graph. The selected region was specially designed for the study. The generated graph is unweighted and undirected

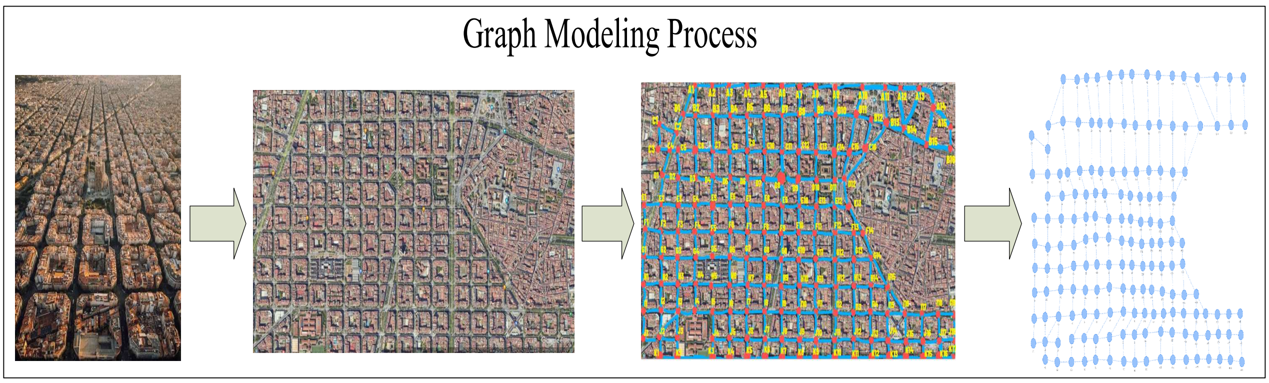

Figure 1. The Process of Barcelona Down Center Graph

4. METHOD

Barcelona is the region of interest in this study due to the fact that it hosts more than 10 million tourists per year on average and the increasing security concerns in such touristic areas. Thanks to the successful results of the study on a real transportation network model, it is thought that it can provide successful results on any transportation network. The region under study is shown in Figure2.

Figure 2. Barcelona Downtown

A bird's eye view of the area subject to analysis is shown in Figure 3

Figure 3. Birdview of Related Area

In the process of designing the network, the popular area of the city of Barcelona was first marked with Google Earth. During this marking, a bird's eye view was obtained to model the intersections and streets more accurately. Each intersection in the image represents a node of the graph and the line between each node represents the edges of the graph. After the whole region was marked, each node was given a name. At this stage, the visual model of the graph was complete. See Figure 4.

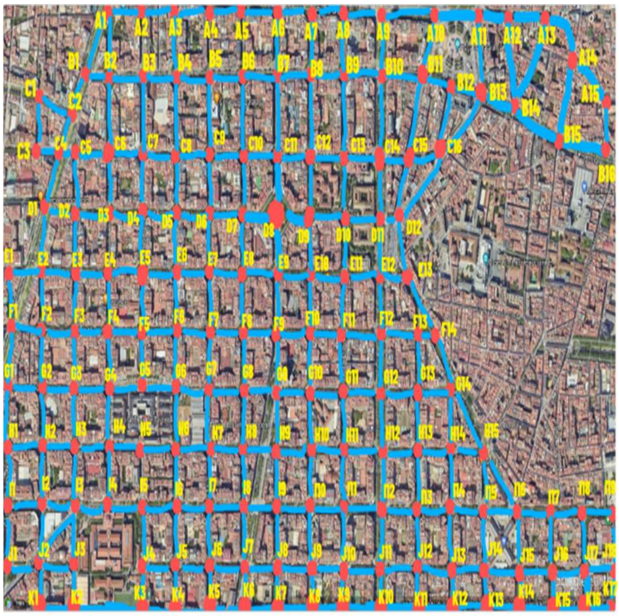

Figure 4. Identify nodes

After the visual model of the network was completed, the graph was modeled again in R programming language in order to apply the algorithm. Malatya Vertex Cover algorithm when applied to the graph, we get. The 169 nodes in the graph before applying the algorithm are shown in Figure 5.

The data set we obtained was simulated to the image on the real map using libraries such as igraph, visNetwork, shiny and an interactive image was provided.

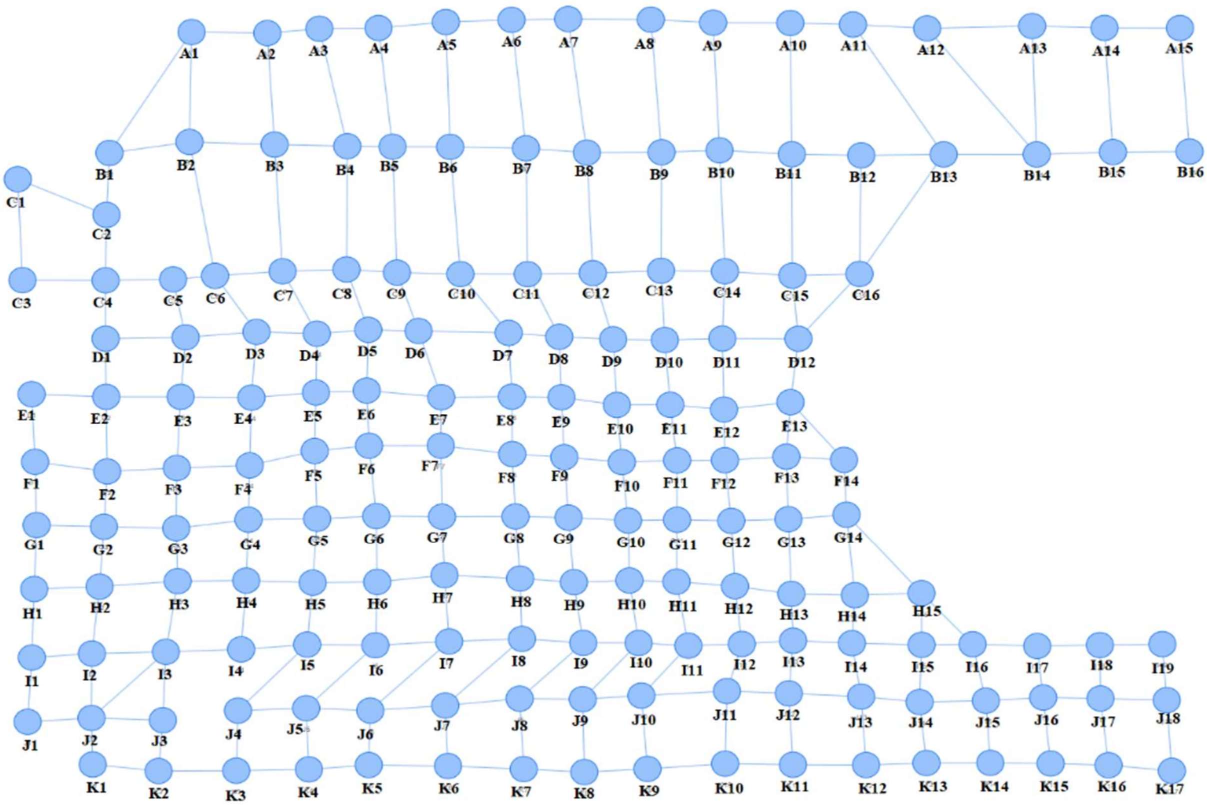

Figure 5. Graph model in R language

4.1. Malatya Vertex Cover Algorithm

The most critical parameter in solving the MVCP problem is the process of selecting the nodes. Malatya algorithm is used to guide this processThis algorithm computes a centrality value for each node and generates an efficient solution set for MVCP. Differently from previous optimization methods, the centrality value clearly defines the nodes. Operations start after Malatya centrality values are calculated. The nodes are ranked starting from the one with the highest centrality value and edge coverage is performed. A new graph is created by removing the selected node and the edges connected to it. These steps are repeated until all edges are covered and finally the minimum edge coverage set is obtained (Yakut et al., 2023). The extended code for this approach is shown in table 1.

Table 1. Proposed Algorithm Pseudo Code

Minimum Vertex-Cover (Karci et al, 2022)

```
calculateMalatyaCentrality <- function(graph) {
  centrality_values <- c()
  vertex_list <- V(graph)  # Get the list of vertices in the graph
 for (vertex in vertex_list) {
   vertex_degree <- degree(graph, v = vertex)

   neighbors_degrees <- degree(graph, v = neighbors(graph, v = vertex))
   # Centrality calculation
   centrality_value <- vertex_degree / neighbors_degrees
   centrality_values <- c(centrality_values, sum(centrality_value))
}
 return(centrality_values)   # Return the calculated centrality values}
# Function to find the vertex with the highest Malatya centrality
findMaximumCentralityVertex <- function(graph) {
   centrality_values <- calculateMalatyaCentrality(graph)
   centrality_data <- data.frame(centrality_values)
 return(order(centrality_data$centrality_values, decreasing = TRUE)[1]) }
# Function to find the vertex cover of the graph
findVertexCover <- function(graph) {
  vertex_cover <- c()  # Initialize an empty vertex cover
  while (ecount(graph) > 0) {   # Continue until all edges are covered
   max_vertex <- findMaximumCentralityVertex(graph)
   vertex_cover <- append(vertex_cover, V(graph)[max_vertex])
   # Remove the selected vertex and its edges from the graph
   graph <- delete_vertices(graph, max_vertex)}
 return(vertex_cover)  # Return the resulting vertex cover}
```

4.1.1. Calculation of Malatya Centrality Value

Centrality plays a critical role in graph theory and network analysis because it allows nodes to be weighted and assigned values based on their location. It is often used to find the most influential nodes in the network and several algorithms have been developed for this purpose. The approach, known as degree centrality, is determined by the links on the node and is the basis of popular algorithms such as PageRank.

For solving the Minimum Vertex Cover Problem (MVCP), calculating Malatya centrality values is an important step. Malatya algorithm, which is a pragmatic and efficient method, is used to calculate these values. Malatya algorithm calculates Malatya centrality values for each node separately. In this calculation, the node's own degree and the degrees of its neighboring nodes are used.

The obtained centrality values are used to determine the nodes to be selected for vertex cover. This approach allows the MVCP to be solved in polynomial time, making it possible to complete the solution in finite steps and in a predictable time.

The Malatya algorithm uses the following formula to calculate the Malatya centrality value $\psi(v_i)$ of each node (Yakut et al., 2023)

$$\psi(v_i) = \sum_{\forall v_j \in N(v_i)} \frac{d(v_i)}{d(v_j)} \quad (1)$$

Here, the Malatya centrality value of each node is the sum of the node's own degree divided by the degree of each neighboring node. This calculation shows that the number and degree of neighboring nodes are as important as the degree of the node itself.

4.1.2. Working and Spacetime Complexities

The Malatya centrality algorithm has been applied to calculate the centrality values for all nodes in the graph, exhibiting a time complexity of $O(n^2)$, which depends on the number of nodes and edges. Additionally, the algorithm requires a maximum of two adjacency matrices, resulting in a space complexity of $|V^2|$(Yakut et al., 2023).

5. EXPERIMENTAL RESULTS

Table 2 shows the nodes selected according to the centrality values taken as a criterion when applying the Malatya Vertex Cover algorithm. 89 nodes were obtained from the entire graph.

Table 2. Nodes selected by the algorithm in the example graph

A1	A3	A5	A7	A9	A11	A12	A14
B1	B3	B5	B7	B9	B11	B13	B14
C1	C4	C6	C8	C10	C12	C14	C16
D2	D4	D6	D8	D10	D12	E2	E4
E6	E8	E10	E12	E13	F1	F3	F5
F7	F9	F11	F13	G2	G4	G6	G8
G10	G12	G14	H1	H3	H5	H7	H9
H11	H13	H15	I1	I3	I4	I6	I8
I10	I12	I14	I16	I18	J2	J4	J6
J8	J10	J12	J14	J16	J18	K2	K4
K6	K8	K10	K12	K14	K16		

The initial 169 nodes were reduced to 88 nodes after the algorithm was applied. The selected nodes are colored in different yellow colors (Figure 6).

These 88 selected nodes cover all edges of the graph. Considering that security cameras are positioned only on the selected yellow nodes, an energy and cost saving of almost 50% is achieved.

The GMin algorithm with a similar greedy approach was applied on the same transportation network and the result was compared with MVCA. GMin, developed by M. Goldberg and colleagues, uses the degree centrality value to select the vertex with the lowest centrality at each iteration as a member of the independent set. The selected vertex and its neighbors are then removed from the graph (Goldberg et al. 2005). Using the mathematical relation between independent set and vertex cover, Maximum Independent Set Members + Minimum Vertex Cover Members = Total number of Vertexes (Alipour and Salari, 2022), the VC locations in the transportation network of Barcelona city were determined.

After applying Gmin to the transportation network, the Vertex Cover value was determined as 92. When the results obtained are analyzed, it is seen that MVCA produces better results than Gmin, which is popularly used in the literature with a value of 88. In other words, with MVCA, all roads can be monitored by installing security cameras at 88 locations detected in the transportation network, while this value is 92 with the Gmin algorithm. MVCA provides a significant cost advantage by using 4 fewer cameras. These results show that MVCA produces results that are successful enough to be included in the literature for solving real world problems.

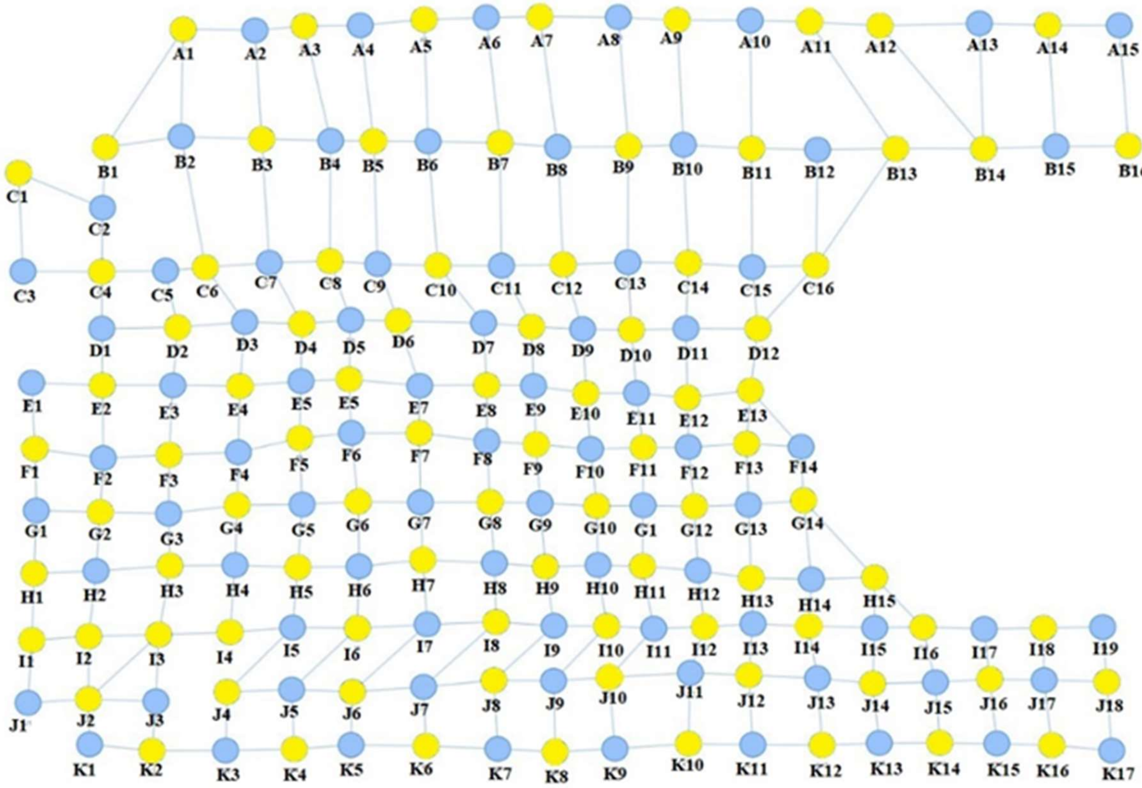

Figure 6. Nodes obtained as a result of the algorithm application those collored yellow

6. RESULTS

In popular touristic areas, it is necessary for each region to install security cameras for tourists to travel more safely and to monitor possible thefts, etc. It would be quite costly to install these cameras at all intersections in the entire region and monitor the area. However, this issue has been meticulously addressed in this study. Thanks to the Malatya Vertex Cover algorithm, 169 intersection points were initially identified for the region whose boundaries were determined, but this number was reduced to 88 by the application of the algorithm. In this way, the region can be monitored with far fewer cameras than usual. Both cost and energy savings were achieved. The transportation network used belongs to the city of Barcelona and has not been studied before in the literature. In addition, this is the first time such an analysis has been done in this region using the Malatya Vertex Cover algorithm. Considering that there are many such touristic regions in the world, this study has paved the way for other regions and the study can be extended for other purposes with different sectors and stakeholders.

REFERENCES

Thulasiraman, K., Swamy, M, NS. (2011). Graphs: theory and algorithms. Montreal: John Wiley & Sons.

Hark, C., Karcı, A. (2022). A new multi-document summarisation approach using saplings growing-up optimisation algorithms : Simultaneously optimised coverage and diversity. https://doi.org/10.1177/01655515221101841.

Thulasiraman, K., Arumugam, S., Brandstädt, A., and Nishizeki, T. (2016). Handbook of Graph Theory, Combinatorial Optimization, and Algorithms, Boca Raton: Chapman & Hall/CRC.

Khattab, H., Mahafzah, B, A., and Sharieh, A. (2022). A hybrid algorithm based on modified chemical reaction optimization and best-first search algorithm for solving minimum vertex cover problem. Neural Comput. Appl., vol. 34(18), pp. 15513–15541, https://doi.org/10.1007/s00521-022-07262-w.

Dinur, I., and Safra, S. (2005). On the hardness of approximating vertex cover. Ann. Math. 162(1), 439–485, https://doi.org/10.4007/annals.2005.162.439.

Angel, D. (2022). Protection of Medical Information Systems Against Cyber Attacks: A Graph Theoretical Approach. Wirel. Pers. Commun., 126(4), 3455–3464, https://doi.org/10.1007/s11277-022-09873-x

Wang, L., Du, W., Zhang, Z., and Zhang, X. (2017). A PTAS for minimum weighted connected vertex cover P_3 problem in 3-dimensional wireless sensor networks. J. Comb. Optim., 33(1), 106–122. https://doi.org/10.1007/s10878-015-9937-z

Hossain, A. (2020). Automated design of thousands of nonrepetitive parts for engineering stable genetic systems. Nat. Biotechnol., 38(12), 1466–1475. https://doi.org/10.1038/s41587-020-0584-2

Gusev, V, V. (2020). The vertex cover game: Application to transport networks. Omega, 97, 102102. https://doi.org/10.1016/j.omega.2019.08.009

Dagdeviren, Z., A. (2021). Weighted Connected Vertex Cover Based Energy-Efficient Link Monitoring for Wireless Sensor Networks Towards Secure Internet of Things. IEEE Access, 9, 10107–10119. https://doi.org/10.1109/ACCESS.2021.3050930

Yigit, Y., Dagdeviren, O., and Challenger, M. (2022). Self-Stabilizing Capacitated Vertex Cover Algorithms for Internet-of-Things-Enabled Wireless Sensor Networks. Sensors, 22(10), 3774. https://doi.org/10.3390/s22103774

Yigit Y., Dagdeviren, Z. A., Dagdeviren, O., and Challenger, M. (2021). Performance Evaluation of Capacitated Vertex Cover Algorithms for Security Applications in Wireless Sensor Networks. in 7th International Conference on Electrical, Electronics and Information Engineering: Technological Breakthrough for Greater New Life, ICEEIE. https://doi.org/10.1109/ICEEIE52663.2021.9616719

Yigit, Y., Akram, V. K., and Dagdeviren, O. (2021) Breadth-first search tree integrated vertex cover algorithms for link monitoring and routing in wireless sensor networks. Comput. Networks, 194, 108144. https://doi.org/10.1016/j.comnet.2021.108144

Dagdeviren, Z. A. (2022) A Metaheuristic algorithm for vertex cover based link monitoring and backbone formation in wireless Ad hoc Networks. Expert Syst. Appl., 213(PA), 118919. https://doi.org/10.1016/j.eswa.2022.118919.

Öztemiz, F., Karci, A. (2021). Malatya İli ulaşım ağı kavşak noktalarının Merkezlilik Analizi. Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi 37(1),511-528. https://doi.org/10.17341/gazimmfd.834255

Yigit, Y., Dagdeviren, O., and Challenger, M., (2022). Self-Stabilizing Capacitated Vertex Cover Algorithms for Internet-of-Things-Enabled Wireless Sensor Networks. Sensors, 22(10), 3774. https://doi.org/10.3390/s22103774

Hebatulla,h K., Sharieh, A. (2019). Most Valuable Player Algorithm for Solving Minimum Vertex Cover Problem. Basel A. January 2019International Journal of Advanced Computer Science and Applications 10(8) https://doi.org/10.14569/IJACSA.2019.0100821

Guo, Quan, and Chen (2019). achieved significant success in solving the Minimum Vertex Cover problem using the Membrane Evolutionary Algorithm (MEAMVC)

Xiaojun, X., Xiaolin, Q., Chunqiang, Y., Xingye, X. (2018). Test-cost-sensitive rough set based approach for minimum weight vertex cover problem. Applied Soft Computing, 64, 423-435. https://doi.org/10.1016/j.asoc.2017.12.023

Jovanovic. R., Sanfilippo, AP., & Voß, S. (2022). Fixed set search applied to the multi-objective minimum weighted vertex cover problem. Journal of Heuristics, 28(4), 481–508. https://doi.org/10.1007/s10732-022-09499-z

Li, R., Hu, S., Cai, S., Gao, J., Wang, Y., & Yin, M. (2019). NuMWVC: A novel local search for minimum weighted vertex cover problem. Journal of the Operational Research Society, 71(9), 1498–1509. https://doi.org/10.1080/01605682.2019.1621218

Yakut, S., Öztemiz, F., & Karci A. (2023). A new robust approach to solve minimum vertex cover problem: Malatya vertex-cover algorithm. J Supercomput 79, 19746–19769 https://doi.org/10.1007/s11227-023-05397-8

Karci, A., Yakut, S., & Öztemiz, F. (2022). A New Approach Based on Centrality Value in Solving the Minimum Vertex Cover Problem: Malatya Centrality Algorithm. Computer Science, 7(2), 81-88. https://doi.org/10.53070/bbd.1195501

Yakut, S., Öztemiz, F., & Karci, A. (2023). A New Approach Based on Centrality Value in Solving the Maximum Independent Set Problem: Malatya Centrality Algorithm. Computer Science, 8(1), 16-23. https://doi.org/10.53070/bbd.122452

Goldberg, M., Hollinger, D., Magdon-Ismail, M. (2005). Experimental Evaluation of the Greedy and Random Algorithms for Finding Independent Sets in Random Graphs. In: Nikoletseas, S.E. (eds) Experimental and Efficient Algorithms. WEA 2005. Lecture Notes in Computer Science, 3503. Springer, Berlin, Heidelberg. https://doi.org/10.1007/11427186_44

Alipour, S., & Salari, M. (2022). Brief announcement: Distributed algorithms for minimum dominating set problem and beyond, a new approach. In 36th International Symposium on Distributed Computing (DISC 2022), 246, 40:1-40:3. https://doi.org/10.4230/LIPIcs.DISC.2022.40