Al-Furat Journal of Innovations in Electronics and Computer Engineering (FJIECE)
ISSN -2708-3985

Modified Algorithm for Shortening Path Kruskal Algorithm to Communicate Between the Patient and The Ambulance During COVID-19

Ghazwan Abdulnabi [1], Israa M. Hayder [2], Arkan A. Ghaib[3], Hussain A. Younis[4 *], Sani Salisu[5]

[1]Department of Computer Science (Educational Science), University of Basrah, 61004, Basrah, Iraq, ghazwan.alali@uobasrah.edu.iq

[2] Dept. Of Computer Systems Techniques, Qurna Technique Institute, STU, Iraq, israa.mh@stu.edu.iq

[3] Department of Information Technology, Management Technical College, Southern Technical University, Basrah, Iraq, arkan.ghaib@stu.edu.iq

[4] College of Education for Women, University of Basrah, 61004, Basrah, Iraq, hussain.younis@uobasrah.edu.iq

[5]Department of Information Tech. Federal University Dutse, Jigawa state, Nigeria, sani.salisu@fud.edu.ng

[*]Corresponding author E-mail: hussain.younis@uobasrah.edu.iq

https://doi.org/10.46649/fjiece.v4.1.7a.25.3.2025

Abstract. *The COVID-19 virus has highlighted the critical need for efficient and rapid communication between patients in distress and emergency medical services, especially ambulance dispatch systems. The following research paper presents a adapted algorithm intended to adjust the Kruskal method for shortening pathways in communication networks linking patients and ambulances through critical situations, confirming timely reply and effective resource distribution. The suggested method takes into reflection the unique challenges posed by COVID-19, as well as the essential to minimize patient-ambulance interaction to decrease the risk of viral transmission. This article shows a new technique, which effects real-time data on COVID-19 contagion rates, indications of patients, ambulance obtainability, and geographical info. In this case, we clarify how by adding these features to the Kruskal algorithm, we reach the finest paths for the best dispatch of ambulances with a perspective of minimizing the spread of the virus. The revised algorithm is designed to focus on routes with the fewest stops and contact points, enabling faster response times and reducing opportunities for disease spread. Furthermore, we present a case study demonstrating the effectiveness of our adapted algorithm in a simulated emergency reply scenario during the COVID-19 pandemic. Comparative analysis with traditional path-shortening algorithms showcases the superior concert of our approach in terms of reply time and safety. This shows delivers a valued effect to the field of emergency medical services, offering a practical solution to address the tests posed by COVID-19 while improving overall response efficiency. The modified algorithm can be seamlessly integrated into existing ambulance dispatch systems, helping healthcare providers and emergency responders make informed decisions that prioritize both patient well-being and public health safety during these unprecedented times.*

Keywords: *COVID-19, shortest path, Kruskal algorithm, linked lists, minimum spanning tree*

1. INTRODUCTION

The shortest route to reducing time and cost must be identified. There are multiple routes in the cocept, mentioned to as edges, which consist of the value or length of the path and are connected to the

Al-Furat Journal of Innovations in Electronics and Computer Engineering (FJIECE)
ISSN -2708-3985

edges of the edges that represent the cities. Its uses are found in various fields such as Google Maps, image processing, and communications. This is an example of the fact that we have a range of cities and a request to create a network at the lowest cost. In the beginning, we must select the straight route in order to attach all the cities without creating a closed circular track. This condition is similar to that of the seller's man, but it is less complex.

The shortest-spanning tree can be used for causal the optimal route. The Kruskal procedure is one of the algorithms used to control the shortest-spanning tree. It splits the tree into clusters, as well as each cluster containing two verses that are linked by a solo edge. The fundamental rule of Kruskal's algorithm is to select the edge with the minimum weight. A circular closed path is not possible due to the designated edges [1].

Emergency medical facilities faced their most extraordinary challenge through the COVID-19 pandemic. Ambulance routing systems designed for traditional purposes might prove inadequate when managing public health crises that display high infection rates through diseases. Healthcare patients must be transported fast to facilities as exposure reductions are equally vital while they move between locations.

Emergency medical facilities can use their conventional ambulance routing procedure under typical operations yet this method fails to account for pandemic-specific changes that occur in service demand. Route optimization becomes complicated because of isolation zones combined with high-risk zones while also requiring individual protective equipment (PPE). Real-time patient-ambulance communication stands essential because patient transportation needs to remain immediate [2]. The patient communicates through architectural design with the ambulance as depicted in Fig.1.

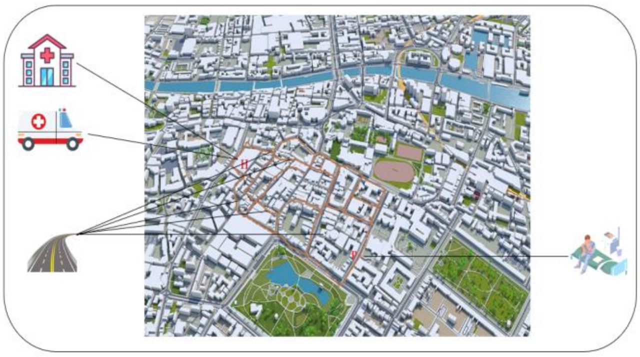


Fig. 1 Architectural communication between the patient and ambulance

1.1 The Role of the Modified Kruskal Algorithm

The Kruskal algorithm succeeds in identifying minimum spanning trees through efficient operations which when adapted can resolve communication issues in ambulance services during pandemic situations. This adapted algorithm enables the development of a safety-conscious scheme that optimizes ambulance routes operation through real-time patient need assessment[3].

A modified procedure integrates multiple criteria such as ambulance patient proximity and PPE availability along with risk assessment on different areas and emergency requirements. The system employs real-time communication between patient role and the ambulance service to create informed decisions about selecting routes.

Al-Furat Journal of Innovations in Electronics and Computer Engineering (FJIECE)
ISSN -2708-3985

1.2 Key Objectives of the Modified Algorithm

The modified Kruskal algorithm has several key purposes which are detailed below:

1. The Ambulance Route Optimization process establishes a path which shortens over time to lower the duration needed for ambulance transportation of a patient. The solution combines traffic data and patient conditions with road restriction information to create shortest possible routes.
2. Safety First: Prioritizing patient and ambulance personnel safety. The algorithm analyzes both infectious outbreak areas and disease risk zones and the availability of protective equipment to reduce the possibility of disease exposure.
3. Real-Time Communication functions as part of the method to keep patients and ambulance services connected in real time. The system can arrange emergency cases according to their urgency levels by tracking up-to-date patient information from patients.
4. Proper distribution of resources stands as essential for managing a pandemic situation. Through its algorithm the program directs ambulances to areas where their presence is essential first so it reduces unneeded journeys yet maximizes operational resources.

This modified Kruskal algorithm functions as a data-based method to configure ambulance routes within pandemic conditions. The system employs technological capabilities together with real-time communication to combine speed with safety measures. [4],[5], ensuring that patients receive prompt medical attention while minimizing the risk of infection transmission.

This article examines the detailed workings of the modified algorithm and implementation with focus on its impact on ambulance communication systems while COVID-19 persists. The paper is structured as following. Section II describes research regarding the Kruskal algorithm.

The shortest path is in Section III. Kruskal's algorithm in Section IV, Modified Kruskal's algorithm in Section V, and Section VI reveal the conclusion. Finally, the paper concludes with Section VII references.

2. RELATED WORK

The researchers presented a study on the use of Kruskal Algorithm Location in Work-flow Push System. An appropriate payment technology algorithm was designed. The work involved the study of the general structural characteristics of the work network on social networking sites, using the dual stack protocol, with a focus on the shortest-path location algorithm.

Drew a minimalist concept tree-spanning algorithm. It concluded with satisfactory results [2].

A study on improved delay management through mobile computing was presented by researchers. The study entailed the submission of two proposals. The first proposal suggested a new model of a three-layer network architecture. CENAM, an abbreviation for Cloud computing and edge computing, was combined.

In the measured edge layer, we propose a Calculator Mutual cooperation scheme among the edge devices and the Kruskal algorithm for calculating the minimum span weighted undirected graph tree composed of edge nodes. The results resolved the cloud computing delay issue [1].

The article discusses a study on industrial wireless sensor networks that are used by synthetic bee colony. The study is based on a tree-building algorithm. Researchers combined quantum computing with an artificial bee colony, as well as designing the construction of an extended tree for bee quantum calculations. In order to simulate the algorithm, the food source substitution strategy was improved. The proposed algorithm improved research performance and quality, according to the results [3].

Al-Furat Journal of Innovations in Electronics and Computer Engineering (FJIECE)
ISSN -2708-3985

The power distribution network acquisition of the minimum power flow path was explained through Kruskal's algorithm during 2019. A performance evaluation took place between Kruskal's Method and Binary Particle Swarm Optimisation (BPSO) in solving network reconfiguration problems. The current research team applies the proposed load restoration method to the 33-bus IEEE system. Observations from the obtained results demonstrate that Kruskal's algorithm runs faster than both Binary Particle Swarm Optimisation (BPSO) [4].

3. SHORTEST PATH

When searching the shortest path in an undirected graph (G) from vertex i to destination vertex j one needs to select the sequences of vertices that will become the second then subsequent vertices until reaching j. A path can be minimized through the selection of the smallest side. The following Fig. 2 displays an undirected graph referred to as a digraph. An examination of the minimum pathway exists between vertex 1 and vertex 3

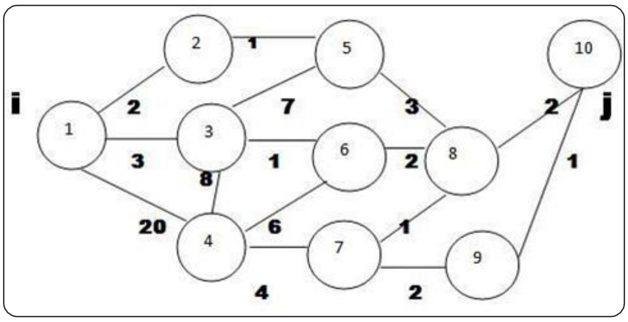


Fig. 2. Undirected graph G

4. KRUSKAL ALGORITHM

Kruskal's algorithm is one of the algorithms that finds the shortest spanning tree for a connected weighted graph [2]. The Minimum Spanning Tree (MST) is a sub-graph of the graph G=(V, U). It includes all the vertices with the least cost of undirected and weighted edges and has the minimum total edge weight [3-7]. Kruskal's algorithm is based on segmenting the graph into clusters and arranging them in ascending order. The weight of the edges determines the priority in the queue [8-9]. It can also be applied in the fields of education for choosing the shortest path for students and in communications for reducing costs in establishing conveyor roads [10-11]. A reviewed the number of studies on the Kruskal algorithm [12-15], Path-Planning Algorithms [16], and Prim's and Boruvka's algorithm [17]. Recently, studies have discussed the issue of communication between patients and ambulances in relation to the speed of receiving treatment, particularly during the COVID-19 pandemic. These studies include references [18-31].

4.1 *WORK OF KRUSKAL'S ALGORITHM*

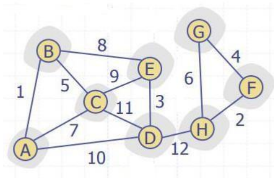
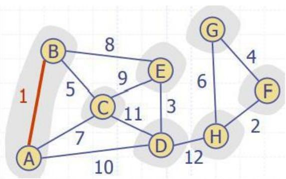
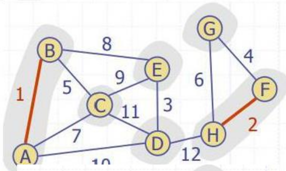
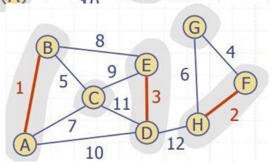
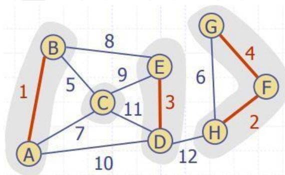
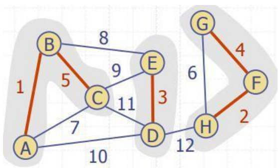
1) Divide the graph into clusters and sort them in ascending order.
2) choose the shortest edge in a graph.
3) Next, select the shortest edge.

Al-Furat Journal of Innovations in Electronics and Computer Engineering (FJIECE)
ISSN -2708-3985

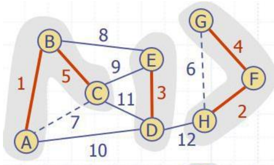
Which one does not make a cycle? Repeat step 2 until all vertices have been connected. Choose an edge that has the lowest cost and does not create a cycle when included in the set of edges that have already been chosen. Note: Input the edge weight into the algorithm, and the output is the shortest spanning tree [2]. The pseudo code is shown below, and Table 1.

```
MST-Kruskal(G,w)
A ← ∅
for each vertex v ∈ V[G] do
   Make-Set(v)
sort the edges of E by nondecreasing weight w
for each (u,v)∈E, in nondecreasing of weight do
  if Find-Set(u) ≠ Find-Set(v) then
     A ← A ∪ {(u,v)}
     Union(Set(u),Set(v))
return A
```

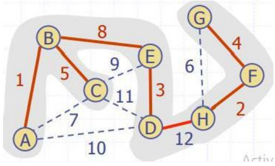
Table 1. Work of kruskal's algorithm

Image	Edge case description
	Example to Kruskal's algorithm
	AB is the shortest edge with weight 1.
	HF is the shortest edge with weight 2.
	DE is the shortest edge with weight 3.
	GF is the shortest edge with weight 4.
	BC is the shortest edge with weight 5.

Al-Furat Journal of Innovations in Electronics and Computer Engineering (FJIECE)
ISSN -2708-3985



They not selected each(GH, AC) because they form a cycle.



BE is the shortest edge with weight 8.
And They not selected each(EC, CD) because they form a cycle.
And then, select BE is the shortes edge with weight 12.

4.2 THE RUNTIME FOR THE ALGORITHM KRUSKAL'S

Kruskal's algorithm consists of two stage:
The first step is to initialise A that takes time O (1).
The second step is to sort the clusters and takes time ElogE [5]. Performing: loop takes time |E|-1.
|V|: MakeSet ().
|E|: FindSet ().
V|-1: Union ().
The total running time is ElogE.

5. MODIFICATION OF THE KRUSKAL'S ALGORITHM

The idea of modification was to reduce the period complexity of this algorithm, which depends on its sorted state. It is known that the time taken for sorting is ElogE. Currently, there was no research algorithm that is less than this time in paper. We reduced the E-1 iteration time by finding the last edge without reaching an end. The technique we used was Linked List.

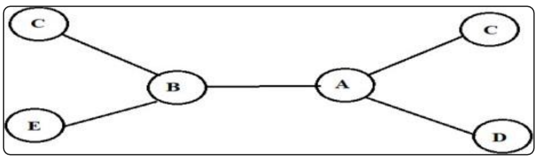


Fig. 3. Nodes network

We modified the MakeSet() function to return the amount of vertices and the grade of each vertex. And the contents of each vertex and a discover variable that specifies whether or not the node was detected. Fig. 3 and Fig. 4 displays the variables the node.

A	degree	Discovered

Fig. 4. The components of the node

Al-Furat Journal of Innovations in Electronics and Computer Engineering (FJIECE)
ISSN -2708-3985

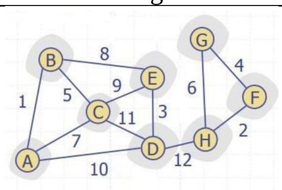
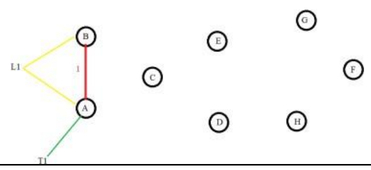
5.1 Pseudocode of Modification Kruskal's Algorithm

```
Algorithm KruskalMST(G):
  Input: A simple connected weighted graph G with n vertices and m edges
  Output: A minimum spanning tree T for G
  for each vertex v in G do
      Define an elementary cluster C(v) ← {v}.
  Let Q be a priority queue storing the edges in G, using edge weights as keys
  T ← ∅        // T will ultimately contain the edges of the MST
  while T has fewer than n − 1 edges do
      (u, v) ← Q.removeMin()
      Let C(v) be the cluster containing v
      Let C(u) be the cluster containing u
      if C(v) ≠ C(u) then
          Add edge (v, u) to T

          if v or u not Discovered then

            v= discover or u= discover

            Link =children  v and u
          vertex :=vertex +1

          Merge C(v) and C(u) into one cluster, that is, union C(v) and C(u)
          if T=2 and all vertices is discovered then
            break

  T=min(degree(Link1)degree(Link2))

  edge=SearchEdge(T)   //search connect edge between T1&T2 from T
  T=Merge(T1,T2)   will merge T1 and T2 with last edge
  Rtuern T
```

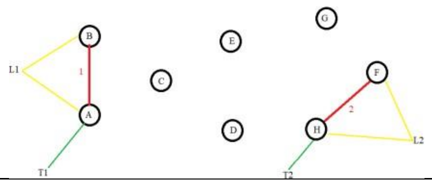
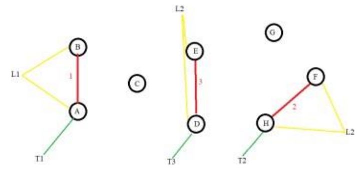
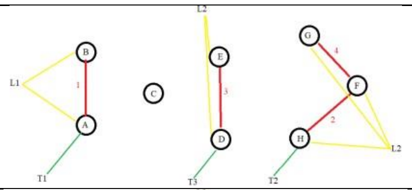
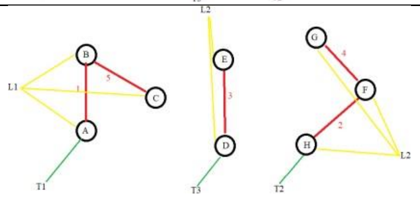
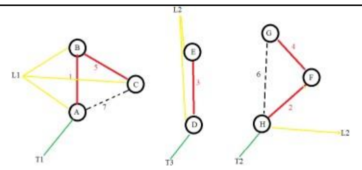
5.2 Work of our_ Kruskal's Algorithm

Initially, the modified algorithm behaves exactly like the classic algorithm. Each cluster has two references until we get two clusters; the first reference refers to the top of the cluster to know the circle when adding. The other reference is to the end of the vertices that contain children, Kruskal's algorithm stopped functioning at that step because there were two trees that contained all the nodes in Fig. 5. The stop of Kruskal's is shown below.

Table 2. Work of our_ kruskal's algorithm

Image	Edge case description
	example to Kruskal's algorithm.
	AB is the shortest edge with weight 1. And L1 T1 references.

Al-Furat Journal of Innovations in Electronics and Computer Engineering (FJIECE)
ISSN -2708-3985

	HF is the shortest edg with weight 2. And L2 T2 references.
	DE is the shortest edg with weight 3. And L3 T3 references.
	GF is the shortest edg with weight 4. And L3 T3 references
	BC is the shortest edg with weight 5. And L1 T1 references.
	They not selected each(GH, AC) becaus they form a cycle. And reference L2 no refers to the vertices G F because not have children (degree=0).

The references T1 and T2 pertain to all vertices within the same tree. L1 and L2 refer to all vertices that contain edges and are not detected as belonging to the same tree (i.e., they have degrees that are not equal to zero).

The reference L1 was compared with L2, whichever contained the lowest degree was searched. In Table 2, L2 had the smallest degree. On L2, the last edge connecting the trees T1 and T2 was searched.

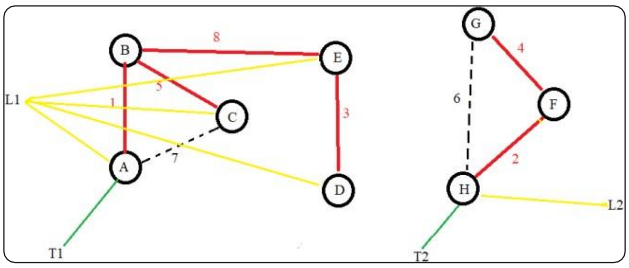


Fig. 5. Steps of Kruskal's

5.2 The runtime for the algorithm Our_ Kruskal's

Our Kruskal's algorithm consists three stage.
The first step is to initialise A that takes time O (1).
The second step is to sort the clusters and takes time ElogE.
performing:
|V|: MakeSet() .
|E|: FindSet().
|V|-1: Union().
We have two trees L1, L2
L=minimum(deg(L1),deg(L2)) take time O(1). Search(L) take time O(deg(L)).
Loop takes time |E|-deg(L1) , such that deg(L1) is from 1 to E. / 2.
The total running time is ElogE.

The interpretation of the results (Basic concept of the classical and modified algorithms and Comparison of the classical and modified algorithms) is shown in Table 3 and Table 4 respectively.

Table 3. Basic concept of the classical and modified algorithms

No.	Classical Kruskal Algorithm	Modified Kruskal Algorithm
1	The classical Kruskal algorithm is used for finding the minimum spanning tree in a connected weighted graph.	The modified Kruskal algorithm aims to reduce the time complexity of the classical algorithm by avoiding unnecessary edges.
2	It operates by selecting the edges with the smallest weights while ensuring that no closed circular paths are formed.	It uses linked lists to manipulate the edges and neglect those that do not contribute to the minimum spanning tree.
3	The algorithm's time complexity is O (E log E), where E is the number of edges.	The time complexity of the modified algorithm is still O(E log E) but may be more efficient in practice due to edge elimination

Table 4. Comparison of the classical and modified algorithms

No.	Function	Classical Kruskal Algorithm	Modified Kruskal Algorithm
1	Efficiency	The time complexity of O(E log E)	Reducing the number of edge calculations
2	Edge Elimination	Edge is present	Edge is not present
3	Application	It finds Minimum Spanning Trees, benefiting network design, clustering, circuit design, transportation, biology, image processing, and more.	It considers factors like proximity to patients, availability of personal protective equipment (PPE), quarantine zones. real-time communication.
4	Flexibility	Can not change	It is adaptable to changing conditions. evolving data
5	Safety	high	high

6. CONCLUSION

In conclusion, the modified Kruskal algorithm presented in this study represents a significant advancement in optimizing communication pathways between patients and ambulances during the COVID-19 pandemic. The algorithm's integration of real-time COVID-19 data, patient symptoms, ambulance availability, and geographical information offers a comprehensive and adaptable solution for emergency response systems. This study demonstrates the practical benefits of this algorithm through a compelling case study, which showcases its superior performance in terms of reducing response times and minimizing the risk of viral transmission. These findings underscore the algorithm's potential to revolutionize the way emergency services operate during public health crises. By prioritizing paths that minimize stops and contact points, our algorithm ensures a faster and safer response to patients in need, which is crucial during a pandemic where every moment counts. Furthermore, its seamless integration into existing ambulance dispatch systems allows for efficient adoption and implementation. Thus, the modified Kruskal algorithm for shortening communication pathways between patients and ambulances during COVID-19 represents a promising innovation in the field of emergency medical services. It empowers healthcare providers and emergency responders to make data-driven decisions that prioritize patient well-being and public health safety, ultimately saving lives and reducing the spread of infectious diseases in the face of unprecedented challenges.

REFERENCES

[1] B. Zhang, J. Fang, and Z. Xiong, "Work-flow Push System Based on Kruskal Algorithm Location," J. Phys. Conf. Ser., vol. 1087, no. 2, 2018, doi: 10.1088/1742-6596/1087/2/022023.

[2] Erniyati and P. Citra, "The implementation of the Kruskal algorithm for the search for the shortest path to the location of a building store in the city of Bogor," IOP Conf. Ser. Mater. Sci. Eng., vol. 621, no. 1, 2019, doi: 10.1088/1757-899X/621/1/012010.

[3] Y. Li, Y. Zhao, and Y. Zhang, "A spanning tree construction algorithm for industrial wireless sensor networks based on quantum artificial bee colony," Eurasip J. Wirel. Commun. Netw., vol. 2019, no. 1, 2019, doi: 10.1186/s13638-019-1496-z.

[4] Aslam, S., Ayub, N., Farooq, U., Alvi, M. J., Albogamy, F. R., Rukh, G., & Bukhsh, R. (2021). Towards electric price and load forecasting using cnn-based ensembler in smart grid. *Sustainability*, *13*(22), 12653.

[5] Razavi, S. A. (2023). *Circulation management in hospitals during COVID-19* (Doctoral dissertation, École de technologie supérieure).

[6] Mensah, I.K.,Zeng, G.and Luo, C.,"E-Government Services Adoption: An Extension of the Unified Model of Electronic Government Adoption", Volume 10, Issue 2, 2020.

[7] Omar Al-Hujran, Mutaz M. Al-Debei b, Akemi Chatfield , Mahmoud Migdadi ,The imperative of influencing citizen attitude toward e- government adoption and use,2018:53;189-203.

[8] Hussain A. Y., Ahmed S. A. & Israa M. H., 2018." Supposed techniques to overcome institute corruption that result from bad use of database management system, ". 7(4): 4980-4983.

[9] Lemuria C.,2005. Electronic government, or e-government, increases the convenience and accessibility of government services and information to citizens. Information Systems Journal, 15(1) :5–25.

[10] H. A. Younis, A. S. A. Mohamed, R. Jamaludin, and M. N. Ab Wahab, "Survey of robotics in education, taxonomy, applications, and platforms during COVID-19," Computers, Materials and Continua, vol. 67, no. 1. pp. 687–707, 2021, doi: 10.32604/cmc.2021.013746..

[11] Katsigiannis, A., Anastopoulos, N., Nikas, K. & Koziris, N. An approach to parallelize Kruskal's algorithm using helper threads. Proc. 2012 IEEE 26th Int. Parallel Distrib. Process. Symp. Work. IPDPSW 2012 1601–1610 (2012) doi:10.1109/IPDPSW.2012.201.

[12] A. A. Wirabudi, R. Munadi, A. Rusdinar, D. Rohdiana, and D. H. Lee, "Design Autonomous Drone Control for Monitoring Tea Plantation Using Dynamic Programming and Kruskal Algorithm," Proc. - 2019 IEEE Int. Conf. Signals Syst. ICSigSys 2019, pp. 103– 109, 2019, doi: 10.1109/ICSIGSYS.2019.8811074.

[13] S. Bistarelli, F. Rossi, and F. Santini, "Kruskal with embedded C- semirings to solve MST problems with partially-ordered costs," Inf. Process. Lett., vol. 169, p. 106107, 2021, doi: 10.1016/j.ipl.2021.106107.

[14] C. Programming, N. R. August, and R. December, "AN alternative for the implementation of kruskal's minimal spanning tree algorithm," vol. 3, 1983.

[15] J. Pak, J. Kim, Y. Park, H. I. L. Son, and S. Member, "Fi eld Evaluation of Path-Planning Algorithms for Autonomous Mobile Robot in Smart Farms," IEEE Access, vol. 10, pp. 60253–60266, 2022, doi: 10.1109/ACCESS.2022.3181131.

[16] D. Rachmawati, Herriyance, and F. Y. P. Pakpahan, "Comparative Analysis of the Kruskal and Boruvka Algorithms in Solving Minimum Spanning Tree on Complete Graph," 2020 Int. Conf. Data Sci. Artif. Intell. Bus. Anal. DATABIA 2020 - Proc., pp. 55–62, 2020, doi: 10.1109/DATABIA50434.2020.9190504.

[17] A. Oesterling, S. Bott, and E. Davis, “Outcomes From Long-Distance Interfacility Transport of Adult Patients on Acute Mechanical Circulatory Support Modes of Transport for MCS PaƟents,” Air Med. J., vol. 41, no. 2, pp. 233–236, 2022, doi: 10.1016/j.amj.2021.09.011.

[18] M. Chetty et al., “Risk factors associated with hospital transfer among mild or asymptomatic COVID-19 patients in isolation facilities in Tokyo : a case-control study,” Australas. Emerg. Care, vol. 25, no. 2, pp. 13–18, 2022, doi: 10.1016/j.nephro.2021.08.008.

[19] M. Chetty, K. P. Govender, and S. Sobuwa, “African Journal of Emergency Medicine Occupational blood and body fluid exposure among emergency medical service providers in the eThekwini metropole of South Africa,” African J. Emerg. Med., vol. 12, no. 2, pp. 97–101, 2022, doi: 10.1016/j.afjem.2022.01.001.

[20] Younis, H.A.; Ruhaiyem, N.I.R.; Ghaban, W.; Gazem, N.A.; Nasser, M. A Systematic Literature Review on the Applications of Robots and Natural Language Processing inEducation. Electronics 2023, 12,2864.https://doi.org/10.3390/electronics12132864

[21] G. Maher, “Australasian Emergency Care Effect of COVID-19 on management of patients with low back pain in the emergency department,” Australas. Emerg. Care, vol. 25, no. 2, pp. 154–160, 2022, doi: 10.1016/j.auec.2021.07.001.

[22] R. Sedgman, E. Aldridge, J. Miller, D. Fleming, and P. Buntine,“department : A retrospective video audit,” Australas. Emerg. Care, vol. 25, no. 2, pp. 126–131, 2022, doi: 10.1016/j.auec.2021.11.002.

[23] Younis, H.A.; Jamaludin, R.; Wahab, M.N.A.; Mohamed, A.S.A. The review of NAO robotics in educational 2014-2020 in COVID-19 Virus (pandemic ara): Technologies, type of application, advantage, disadvantage and motivation. IOP Conf. Ser. Mater. Sci.Eng. 2020, 928, 032014.

[24] A. S. Al-dosary et al., “Epilepsy & Behavior Public awareness of first-aid management of seizures in Saudi Arabia,” Epilepsy Behav., vol. 129, p. 108634, 2022, doi: 10.1016/j.yebeh.2022.108634.

[25] Al-Hamodi, A. A., & Lu, S. F. (2016, June). MapReduce Frequent Itemsets for Mining Association Rules. In 2016 International Conference on Information System and Artificial Intelligence (ISAI) (pp. 281-284). IEEE.

[26] N. K. Glober, “American Journal of Emergency Medicine Increased weight in patients with time-sensitive diagnosis is associated with longer prehospital on-scene times,” Am. J. Emerg. Med., vol. 53, pp. 236–239, 2022, doi: 10.1016/j.ajem.2022.01.022.

[27] L. P. Dawson et al., “Annals of Epidemiology Incidence , diagnoses and outcomes of ambulance attendances for chest pain : a population- based cohort study,” Ann. Epidemiol., vol. 72, pp. 32–39, 2022, doi: 10.1016/j.annepidem.2022.04.010.

[28] P. Brunet, “ScienceDirect ´ s en re ´ gion PACA , et estimation des e ´ conomies re ´ alisables par dialyse l ’ Assurance maladie Evaluation of ambulance transport relevance of dialysis patients in the PACA region (France), and estimation of savings by the Health I,” vol. 18, pp. 35–44, 2022, doi: 10.1016/j.nephro.2021.08.008.

[29] M. Rahbar, Y. Sharifi, S. Mohammad, and K. Aghamir, “Case Report A case of COVID-19 infection in a kidney transplant recipient after receiving the single dose of COVID-19 vaccination,” Ann. Med. Surg., vol. 77, no. March, p. 103517, 2022, doi: 10.1016/j.amsu.2022.103517.

[30] Hashim, M. M., Aleiwi, J. J., Ibrahim, R. K., Mohammed, W. R., & Nahi, A. A. (2024, February). Concealing Secret Data in Medical Images Based on Even/Odd Pixels and PSO Algorithm for Improve Steganography System. In *2024 IEEE International Conference on Artificial Intelligence and Mechatronics Systems (AIMS)* (pp. 1-5). IEEE.

[31] M. Gharib, S. Mohammad, T. Fatemi, and F. Jolai, “Transportmetrica A : Transport Science A multi-objective stochastic programming model for post-disaster management post-disaster management,” 2021, doi: 10.1080/23249935.2021.1928790.