Turkish Journal of Science & Technology
16(1), 23-30, 2021

4×4 Knight's Graph Analysis by Modularity: A Knight Graph Application

Serkan GÜLDAL[1*]
[1] Department of Physics, Faculty of Arts and Sciences, Adiyaman University, Adiyaman, Turkey
*sguldal@adiyaman.edu.tr

(Geliş/Received: 27/03/2020; Kabul/Accepted: 21/10/2021)

Abstract: Modularity is a well-known technic to analyze datasets that are in the form of graphs. The modularity divides the network into clusters concerning the connections between nodes. The clusters present the existence of shared properties between the nodes in the same clusters. In the present study, we analyze the 4×4 knight's graph by modularity to seek the relation with Knight Covering Problem solutions (for the specified problem a.k.a. 4-KCP). Our investigation is completed for resolutions from 0.2 to 3.2. The maximum modularity score is 0.417 which is found for the resolution = 0.6, 0.8, 1, and 1.4. Moreover, resolution 1.4 is the optimum resolution to find some solutions of 4-KCP. Also, the analyses show that resolution 0.2 is the best resolution to find all solutions of 4-KCP since it approximates the permutation algorithm.

Keywords: Knight's graph; modularity, 4-KCP

Modülerlik ile 4×4 At Graf Analizi: At Graf Uygulaması

Öz: Modülerlik graf biçimli verilerin analizinde iyi bilinen bir tekniktir. Modülerlik düğümler arasındaki bağları göz önünde bulundurarak anlamlı gruplara ayırır. Gruplar aynı gruptaki düğümler arasındaki ortak özelliklerin varlığını tanımlar. Bu çalışmada At Kaplama Problem çözümleri ile 4×4 at grafını (bu çalışmada tanımlan problem için kısaca 4-KCP) modülerlik analizi arasındaki ilişkiyi araştırıyoruz. Araştırmamız çözünürlüğün 0.2 ile 3.2 olduğu aralık için tamamlandı. Maksimum modülerlik puanı 0.417'dir ve bu modülerlik puanı 0.6, 0.8, 1, ve 1.4 çözünürlükleri için hesaplanmıştır. Buna ek olarak 1.4 4-AKP'nin bazı çözümlerini bulmak için ideal çözünürlüktür. Ayrıca, analizler 0.2 çözünürlüğünün permutasyon metoduna yakınsadığından dolayı 4-AKP'nin tüm çözümleri bulmak için en iyi çözünürlük olduğunu gösterdi.

Anahtar kelimeler: At grafı; modülerlik, 4-AKP

1. Introduction

The moves of a knight on the chess introduces surprise attacks against to opponent beyond the regular moves such as jumping over pieces. The generated problems by knights movements on the board have been subject to the many types of research, for instance, the knight's tour problem is investigated to obtain a solution [1-3]. The moves are used for encryption images [4-6]. Additionally, to position a certain number of knights on a board based on the movements which are the various versions of the Knight Covering Problem (KCP) [7-14]. There are methods are developed to solve by the means of the knight graph to find solutions for knight covering problems such as the independent set [15, 16] and the Girvan-Newman clustering algorithm [17].

The network analyses are intensively utilized to extract information in data networks such as social networks [18-20], computational networks [21-23], word networks [24-26], biological networks [27-29], infection networks [30, 31]. There are plenty of analysis methods such as statistical analysis, power graph analysis, visual analysis, and clustering. Clustering divides the whole network to clusters of the nodes. The clusters present the common properties of nodes in the same cluster. The clustering method is one of the first attempts to understand the considered graph since it generates intuitive results in a computationally efficient manner. There are various clustering algorithms based on the properties of the network. For example, the Girvan-Newman algorithm uses edge betweenness, Highly Connected Clusters uses graph connectivity, k-means clustering uses the mean value, and Modularity uses the strength of division of a network into modules (a.k.a. clusters). The modularity algorithm has flexibility based on the resolution for different graphs. Higher resolutions divide the network to greater clusters and lower resolutions are otherwise. The optimum resolution changes respect to the network and the intent. We intend to understand the response for various resolutions in the search of 4-KCP solutions. Thus, we have investigated a range of resolutions to obtain the resolution which is computationally most efficient and extracts all solutions of 4-KCP.

In the present study, the knight graph illustration of 4-KCP is investigated by the modularity of the graph. 4-KCP has 16 cells to be occupied by the knights (See **Figure 1**). The cells on the corners can attack 2 cells. The cells

* Corresponding author: sguldal@adiyaman.edu.tr. ORCID Number of authors: 1 0000-0002-4247-0786

on the edges can attack 3 cells, and the cells in the middle can attack 4 cells. Additional to cells which knight attacks, an extra cell is occupied by the particular knight.

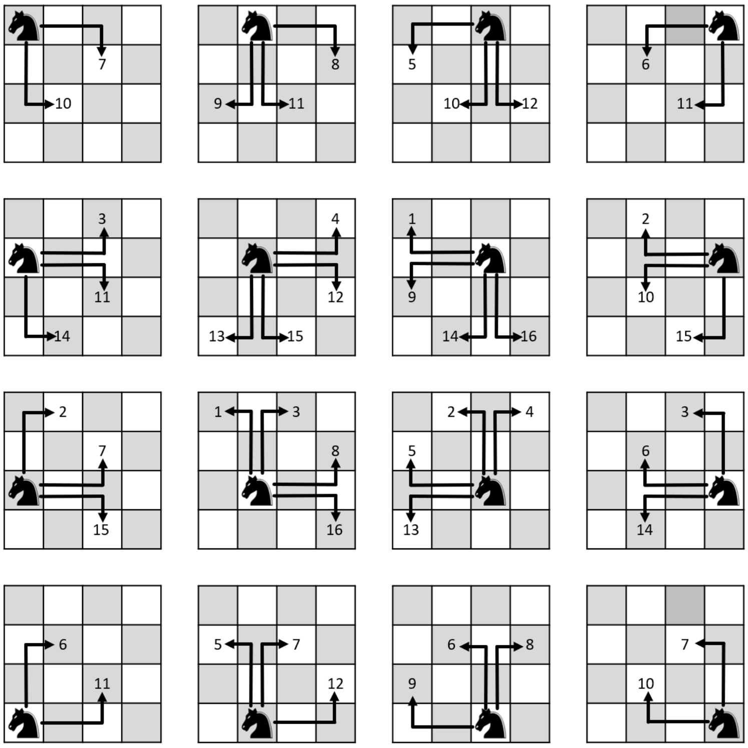
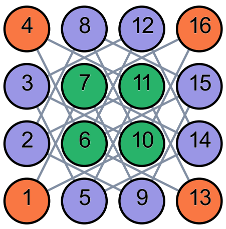


Figure 1. 4-KCP has 16 cells to place the knight

4-KCP board is indexed and presented color-coded respect to the degree of each node in Figure 2. 4-KCP is the graph which has 16 nodes and 24 edges. There are nodes with 2, 3 and 4 degrees. They are the portion of the whole graph respectively 25%, 50%, and 25%. The knights are on the corner cells (orange) can attack to 2 cells located in the center. The cells on the edges (purple) with 3 degrees can attack 2 cells on the edges and 1 cell on the center. The cells on the center can attack 4 cells which are 2 of them on the edges and 2 of them on the corners (see **Figure 1**).



Color code	Degree	Percentage in the graph (%)
● (orange)	2	25
● (purple)	3	50
● (green)	4	25

Figure 2. 4-KCP graph is shown by the degrees of nodes

Many algorithms are built on knight graphs to solve N-KCP [15]. The algorithms use the graph theory properties such as the independent set. In **Figure 3**, the solutions of 4-KCP are found by the independent set algorithm. In the presented solutions exactly 4 knights are placed to cover 4x4 board. The found solutions are rotationally symmetric. With similar utilization of graphs, in this study, we use modularity to analyze the 4-KCP graph to find solutions. The network analysis method divides the 4-KCP graph into densely connected clusters. The clustered nodes show a strong relationship between the knights. Thus, they highlight the knights which are less likely to be in the same solution. The details of the modularity and the algorithm is discussed in the following sections.

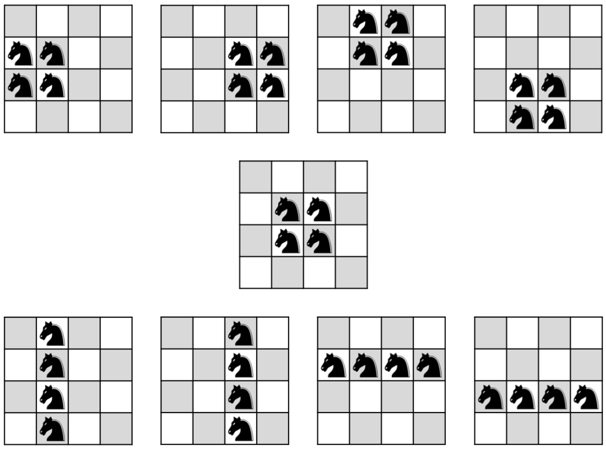


Figure 3. 4-KCP solutions by 4 knights only

2. Modularity

In the present study, we used the modularity to identify the close connected knights for the 4-KCP problem. The modularity score is formulated by various formulas. The formula which is utilized is as follows [32]:

$$Q = \frac{1}{2\text{m}} \sum_{i,j} \left(A_{ij} - \gamma \frac{k_i k_j}{2m} \right) \delta(c_i, c_j) \qquad (1)$$

where δ-function is 1 if $c_i = c_j$; in other words, node i and j are in the same cluster. m stands for the number of edges in the graph. γ is the resolution. k_i is the degree of node i and k_j is the degree of node j. A_{ij} represents the weights of the edge between nodes i and j. It is the same for all since the effect of all knights is equal.

Our analysis is completed by the Gephi [33-35]. We applied the resolution from 0.2 to 3.2. The implementation and analysis will be given in the Results and Discussion section.

3. Results and Discussion

In the present study, we have investigated the relational information of the 4-KCP graph by modularity score. There are extracted communities from 1 to 8 respect to resolution.

The modularity phenomena built on the strong relationships between nodes. On the other hand, the N-KCP solutions are to place the knight which has weak/no relations. Thus, the extracted clusters show the list of positions that are the least likely to be in the same solution.

In **Figure 4**, modularity results on 4-KCP (for resolution = 0.2 - 3.2) graphs are presented. The resolution 3.2 is extracted 1 cluster in **Figure 4**.p. Thus, in the generated solutions shows that no two nodes could be included simultaneously in a solution. Also, there is no solution by one knight, so 17 permutations which include no knight are not solutions for 4-KCP. In **Figure 4**.o, 2 clusters are generated which are (1, 2, 7, 8, 9, 10, 15, 16) and (3, 4, 5, 6, 11, 12, 13, 14). Similarly, 2 nodes / 2 knights solutions do not exist for 4-KCP. 80 combinations of the two clusters are not the solutions. Likewise, modularity application for resolutions = 2.8, 2.6, 2.4, 2.2, 2.0, 1.8, and 1.6 cannot generate 4-KCP solutions. For the resolution = 1.4 in **Figure 4**.g, the 4-KCP graph is divided into 4 clusters. 8 out of 9 solutions with the length of 4 are generated by the permutation 499 combinations (See **Table 1**).

Table 1. The solution generated by modularity application for the resolution 1.4. 8 out of 9 solutions for length 4 are generated.

(2, 3, 6, 7)	(5, 6, 7, 8)	(7, 8, 11, 12)
(2, 6, 10, 14)	(5, 6, 9, 10)	(9, 10, 11, 12)
(3, 7, 11, 15)	(6, 7, 10, 11)	(10, 11, 14, 15)

3 clusters are introduced by the resolution 1.2, and there is no such solution with length 3. For the resolutions 1.0, 0.8 and 0.6, 4 clusters are generated. This modularity is similar to resolution 1.4, so the generated solutions. The modularity is divided into 6 clusters for the resolution 0.4. Clusters make 2159 combinations length varies from 1 to 6. This generates 13 solutions of 4-KCP.

The modularity by resolution 0.2 divides the particular graph to 8 clusters. The number of permutations is 6560 length varies from 1 to 8. This permutation includes all 31 solutions of 4-KCP.

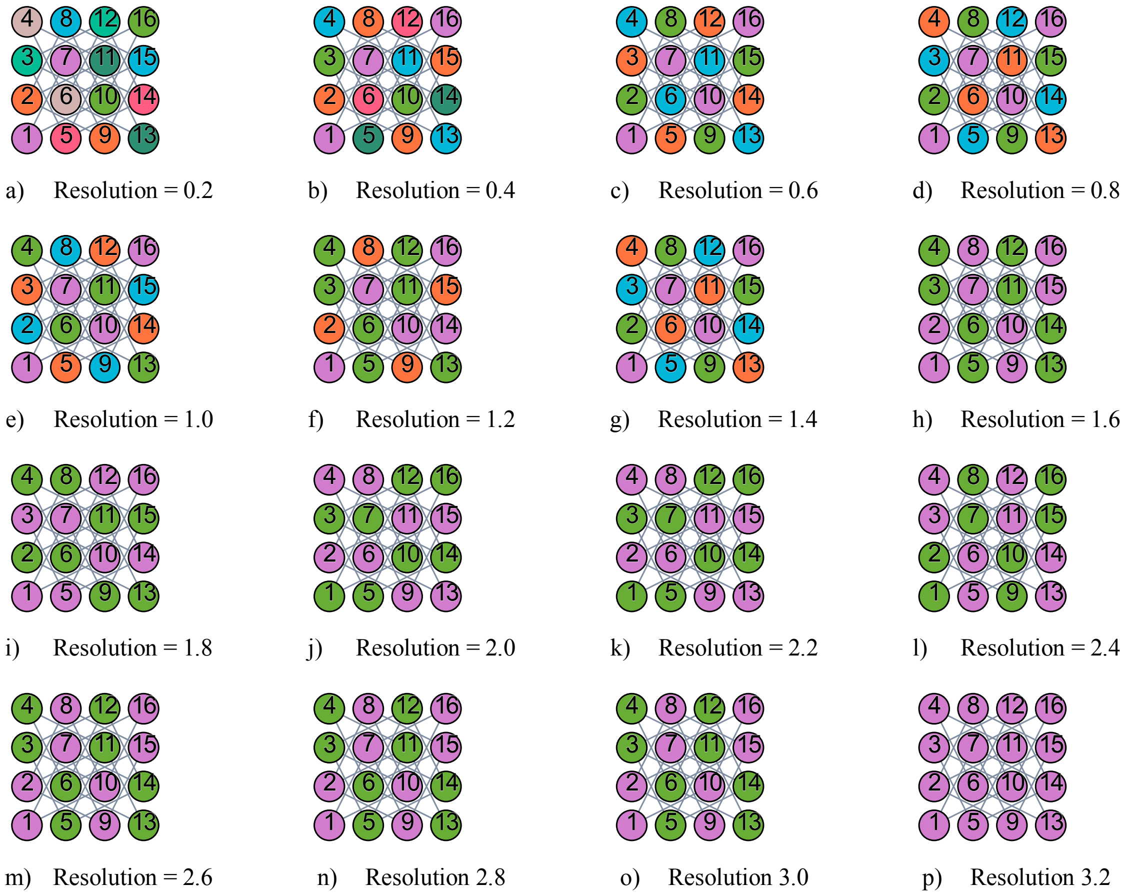


Figure 4. Modularity is applied to 4-KCP graphs for various resolutions from 0.2 to 3.2

The applied modularity for various resolutions divided the 4-KCP graph to the various number of clusters. In **Figure 5**, pie charts are shown which are greater to highlight the direction of the increasing number of clusters. The decreasing resolutions divide the network into smaller clusters. The increasing number of clusters more likely to generate 4-KCP solutions. In **Figure 6**, the number of permutations is compared with the number of solutions for the resolution from 0.2 to 3.2. There are no solutions obtained for the resolutions 1.6-3.2. Similarly, resolution 1.2 divides the 4-KCP graph to insufficient clusters. The resolutions 1.4 (number of permutations = 499) and 0.6-1 (624) generates 8 solutions. Resolution 1.4 generates the same number of solutions by less number of permutations. Thus, resolution 1.4 extracts relatively more meaningful clusters. All the solutions are obtained by the maximum number of permutations. The maximum number of permutations are generated for resolution 0.2. The 4-KCP graph is divided into 8 clusters with two nodes for each. Thus, it approximates the permutation solution algorithm [15]. It is computationally more efficient than permutation solutions since reduces the number of permutations. Thus, we conclude the existence of an explicit correlation between generated permutations and related solutions.

Based on the modest comparison of the number of generated permutations with the number of found solutions, the computational efficiency of clustering is obtained with respect to the resolution in **Figure 7**. The computational efficiency is calculated as:

$$Efficiency\ of\ the\ cluster = \frac{Number\ of\ found\ solutions * 100}{Number\ of\ permutations} \quad (2)$$

Resolution 0.2 introduces the best clustering which obtains all solutions, but it is not computationally efficient (0.47256) because the highest number of permutations are generated. The most computationally efficient resolution is 1.4 by 1.60321. It finds 8 solutions by 499 permutations. All clustering could not identify a solution. Between the clusters which find a solution, resolution 0.4 is the least efficient (0.60213) since it finds 13 solutions in 2159 permutations.

Figure 5. Increase resolution causes to decrease to the number of communities for 4-KCP graph

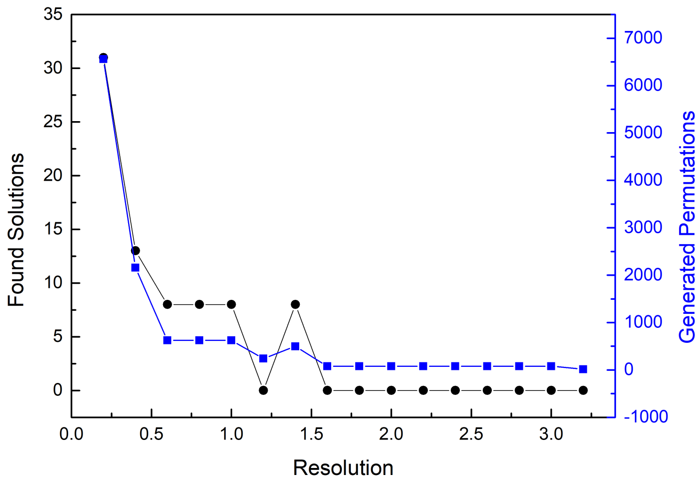


Figure 6. (Color online) Generated permutations have correlations with the found solutions

The modularity score which identifies the quality of clustering has no explicit correlation with the number of found solutions for the specified resolutions (**Figure 6** and **Figure 8**). The highest modularity score is 0.417 (for resolution = 0.6, 0.8, 1, and 1.4). Thus, clustering presents the best sub-communities.

Resolution 1.4 identifies computationally as the most efficient clustering to find solutions with fewer permutation. However, all solutions are found for resolution 0.2 which is the computationally least efficient between the clusters that can lead to some solution.

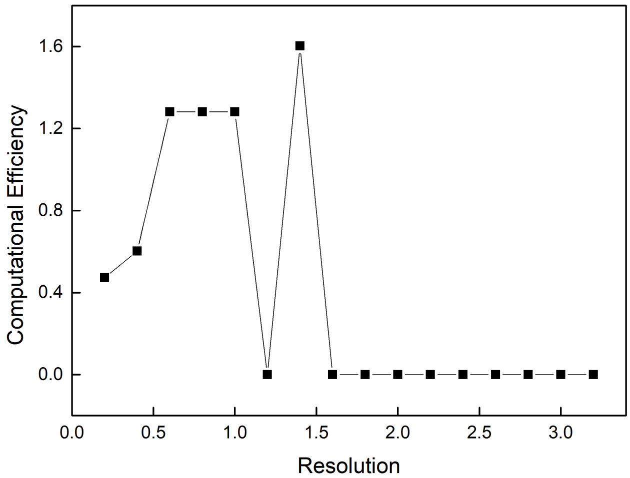


Figure 7. Computational efficient clustering is obtained by resolution 1.4. However, all solutions are obtained by the resolution 0.2 which is the computationally least efficient resolution between the solutions found

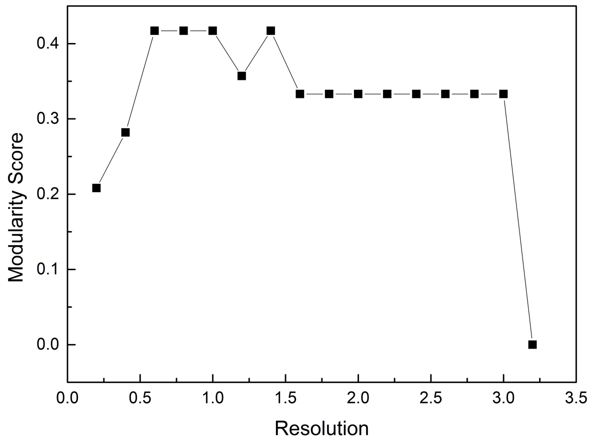


Figure 8. Modularity score versus Resolution

4. Conclusion

In this study, we have applied the Modularity 4×4 knight graph. Thus, an analysis is applied to solve the Knight Covering Problem (N-KCP). Our analysis is limited to the 4-KCP graph. The analyses show resolution 1.4 is the computationally optimal resolution to find some solutions of 4-KCP. Moreover, the analysis shows resolution 0.2 is the best resolution to find all solutions of 4-KCP.

Based on our analysis, the modularity is a promising method to solve N-KCP. Thus, in future studies, the analyses will be extended around the N-KCP with modularity.

5. Acknowledgment

The authors declare no conflict of interest.

References

[1] S. Bai, G. Zhu, and J. Huang, "An Intelligent Algorithm for the (1,2,2)-Generalized Knight's Tour Problem," in *2013 Ninth International Conference on Computational Intelligence and Security*, 2013, pp. 583-588.

[2] H. Jian and B. Sen, "An Efficient Algorithm for the Generalized (1,k)-Knight's Tours Problem," in *2009 First International Workshop on Education Technology and Computer Science*, 2009, vol. 1, pp. 697-701.

[3] S. Bai, X. Liao, X. Qu, and Y. Liu, "Generalized Knight's Tour Problem and Its Solutions Algorithm," in *2006 International Conference on Computational Intelligence and Security*, 2006, vol. 1, pp. 570-573.
[4] A. Philip, "A Generalized Pseudo-Knight?s Tour Algorithm for Encryption of an Image," *IEEE Potentials,* vol. 32, no. 6, pp. 10-16, 2013.
[5] J. Kumar and S. Nirmala, "Securing the contents of document images using knight moves and genetic approach," in *2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2015, pp. 1091-1095.
[6] J. Delei, B. Sen, and D. Wenming, "An Image Encryption Algorithm Based on Knight's Tour and Slip Encryption-Filter," in *2008 International Conference on Computer Science and Software Engineering*, 2008, vol. 1, pp. 251-255.
[7] D. C. Fisher, "On the n x n Knight Cover Problem," *Ars Comb.,* vol. 69, / 2003.
[8] F. Rubin, "Improved Knight Coverings," *Ars Comb.,* vol. 69, / 2003.
[9] F. Rubin, "Knight Covers for the 50x50 Chessboard," presented at the Mathfest 2004, Providence RI, 2004.
[10] F. Rubin, "A Family of Efficient Knight Covering Patterns," *Journal of Recreational Mathematics,* Article vol. 33, no. 3, pp. 165-175, 2005.
[11] F. Rubin, "An Improved Method for Finding Knight Covers," *Ars Comb.,* vol. 82, / 2007.
[12] B. Lemaire, "Knights Covers on NxN Chessboards," *J. Recr. Math.,* vol. 31, pp. 87-99, 2003.
[13] A. H. Jackson and R. P. Pargas, "Solutions to the NxN Knights Covering Problem," *J. Recr. Math. ,* vol. 23, pp. 255-267, 1991.
[14] F. Wei, "Research on Knight Covering Based on Breadth First Search Algorithm," (in English), *APPLIED MECHANICS AND MATERIALS,* vol. 686, pp. 377-380, 2014.
[15] S. Güldal, M. Lipscomb, and M. M. Tanik, "Solving Knights Covering Problem: Backtracking, Permutation, Bipartite Graph, and Independent Set," presented at the Nineteenth Annual Early Career Technical Conference, Birmingham, Alabama USA, 2019
[16] S. Güldal, M. M. Tanik, and M. M. Lipscomb, "Solving Knights Covering Problem by a Hybrid Algorithm," presented at the IEEE SouthEastConn, Huntsville, Alabama, April 11 - 14 2019,
[17] S. Güldal, "Connectives of Knights Covering Problem By Girvan-Newman Clustering," presented at the SDPS 2019 Workshop, Madrid, Spain, 25-26, November 2019,
[18] M. Gençer, "Sosyal Ağ Analizi Yöntemlerine Bir Bakış," *Yildiz Social Science Review,* vol. 3, pp. 19-34, 12/15 2017.
[19] P. Nerurkar, M. Chandane, and S. Bhirud, "Understanding attribute and social circle correlation in social networks," *Turkish Journal of Electrical Engineering and Computer Sciences,* vol. 27, pp. 1228-1242, 03/22 2019.
[20] N. Almolhem, Y. Rahal, and M. Dakkak, "Social network analysis in Telecom data," *Journal of Big Data,* vol. 6, 12/01 2019.
[21] "Analysis of the Application of Artificial Intelligence in Computer Networks Technology," *IOP Conference Series: Materials Science and Engineering,* vol. 750, p. 012097, 03/24 2020.
[22] C. Engström and S. Silvestrov, "PageRank for networks, graphs, and Markov chains," *Theory of Probability and Mathematical Statistics,* vol. 96, p. 1, 10/05 2018.
[23] Y. L. Karpov, I. Volkova, A. A. Vylitok, L. Karpov, and Y. G. Smetanin, "Designing classes' interfaces for neural network graph model," *Proceedings of the Institute for System Programming of RAS,* vol. 31, pp. 97-112, 10/01 2019.
[24] L. Johnsen, "Graph Analysis of Word Networks," *http://ceur-ws.org/Vol-2021/,* vol. 2021, 01/01 2017.
[25] E. Hasanah and D. Agustiningsih, *Analysis of "Halal" Word in Social Media Using Text Mining and Word Networking*. 2020.
[26] R. Ozcelik, G. Uludoğan, S. Parlar, Ö. Bakay, O. Ergelen, and O. Yildiz, *User Interface for Turkish Word Network KeNet*. 2019, pp. 1-4.
[27] K. Kugler, L. Mueller, A. Graber, and M. Dehmer, "Integrative Network Biology: Graph Prototyping for Co-Expression Cancer Networks," *PloS one,* vol. 6, p. e22843, 07/29 2011.
[28] B. Eckman and P. Brown, "Graph data management for molecular and cell biology," *IBM Journal of Research and Development,* vol. 50, pp. 545-560, 12/01 2006.
[29] V. Gadiyaram, S. Vishveshwara, and S. Vishveshwara, *From Quantum Chemistry to Networks in Biology: A Graph Spectral Approach to Protein Structure Analyses*. 2019.
[30] S. Valverde, B. Vidiella Rocamora, R. Montañez Martínez, A. Fraile, S. Sacristán, and F. García-Arenal, "Coexistence of nestedness and modularity in host–pathogen infection networks," *Nature Ecology & Evolution,* pp. 1-10, 03/09 2020.

[31] T. Shiino, "Phylodynamic analysis of a viral infection network," *Frontiers in microbiology,* vol. 3, p. 278, 07/31 2012.
[32] M. E. J. Newman, "Analysis of weighted networks," *Physical Review E,* vol. 70, no. 5, p. 056131, 11/24/ 2004.
[33] M. Bastian, S. Heymann, and M. Jacomy, "Gephi: an open source software for exploring and manipulating networks," in *International AAAI Conference on Weblogs and Social Media*, 2009.
[34] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, E. J. J. o. s. m. t. Lefebvre, and experiment, "Fast unfolding of communities in large networks," vol. 2008, no. 10, p. P10008, 2008.
[35] R. Lambiotte, J.-C. Delvenne, and M. J. a. p. a. Barahona, "Laplacian dynamics and multiscale modular structure in networks," 2008.