Afyon Kocatepe Üniversitesi Fen ve Mühendislik Bilimleri Dergisi
Afyon Kocatepe University Journal of Science and Engineering

AKÜ FEMÜBİD **21** (2021) 035101 (559-564) AKU J. Sci. Eng. **21** (2021) 035101 (559-564)
DOI: 10.35414/akufemubid.870835

Araştırma Makalesi / Research Article

The Effect of Scoring Factor for Leiden Algorithm

Serkan GÜLDAL[1*]
[1]*Adiyaman University, Faculty of Arts and Sciences, Physics Department, Adiyaman, 02040, Turkey*

Corresponding Author e-mail: sguldal@adiyaman.edu.tr ORCID ID: https://orcid.org/0000-0002-4247-0786

Geliş Tarihi: 29.01.2021 Kabul Tarihi: 10.05.2021

Keywords
Clustering;
Graph;
Leiden Algorithm;
N-KCP

Abstract

Leiden algorithm is a widely utilized algorithm to cluster network graphs. It divides the specified network into smaller clusters. The clusters are relatively dense networks of vertices. In the process, the networks are divided based on quality factors. In this study, we compare the result of the Leiden algorithm with changing quality factors, namely Modularity and Constant Potts Model (CPM). For our analysis, we used 3×3 knight graph. Our investigation is completed for resolutions from 0.1 to 4.0 for Modularity and from 0.1 to 1.0 for CPM. The maximum quality scores are 0.9 and 0.59375 for Modularity and CPM respectively. The continuous decrease in the quality was recorded for both cases with respect to the increasing resolution. Both scoring factors are followed similar trends, but CPM has a relatively rapid division of the specified graph.

Leiden Algoritmasında Kalite Faktörünün Etkisi

Anahtar kelimeler
Kümeleme;
Çizge;
Leiden Algoritması;
N-AKP

Öz

Leiden algoritması, çizgeleri kümelemek için yaygın olarak kullanılan bir algoritmadır ve belirtilen çizgeyi daha küçük kümelere böler. Bu kümeler, nispeten yoğun düğüm çizgeleridir. Süreçte çizgeler kalite faktörlerine göre kümelenir. Bu çalışmada Leiden algoritmasını Modülerlik ve Sabit Potts Modeli (CPM) kalite faktörleri ile değişimini karşılaştırılmıştır. Analiz için 3×3 at çizgesi kullanıldı. İnceleme Modülerlik için 0,1'den 4,0'a ve CPM için 0,1'den 1,0'a kadar olan çözünürlükler için tamamlandı. Maksimum kalite puanları Modülerlik ve CPM için sırasıyla 0,9 ve 0,59375'tir. Kalitede artan çözünürlüğe göre her iki durumda da sürekli düşüş kaydedildi. Her iki puanlama faktörü de benzer eğilimler izlendi, ancak CPM nispeten konu edilen çizgeyi daha hızlı kümeledi.

1. Introduction

The Leiden algorithm is used to cluster the specified graph such as medical (Wang *et al.* , Clifford *et al*. 2019, Dong and Yuan 2020, Gibbs *et al*. 2020, Li *et al*. 2020), economics (Pasimeni 2020), and text analysis (Ahlgren *et al*. 2019, Boy 2020, Miura *et al*. 2020) subject. It is a well-known clustering algorithm which efficiently divides the whole network into smaller clusters of nodes. Each cluster presents a relation of included nodes in the same cluster. It is extensively utilized since it is considered computationally efficient (Traag *et al*. 2019). In this study, we used the Leiden algorithm combined with Modularity and Constant Potts Model (CPM) quality functions to cluster 3-KCP (Girvan and Newman 2002, Delvenne *et al*. 2013, van Laarhoven and Marchiori 2013, Van Laarhoven and Marchiori 2014, Esmailian and Jalili 2015, Devi and Poovammal 2016). The subjected quality functions are used widely to measure the appropriateness of the clusters.

3-KCP graph is based on the knight moves on the 3×3 board (See Figure 1). The knights as chess pieces are extensively used to build a defense and orchestrate attacks because their moves are not emulated by the other pieces. Thus, knights are being subjected to the various problems based on the knights' movements on the chess-board-likes without any known analytical solutions. The well-known knight's tour problem is one of them. The knight's tour problem puzzled many mathematicians and computer scientists. Many algorithms are implemented and tested to obtain solutions (Parberry 1997, Hingston and Kendall 2004, Bai *et al*. 2006, Jian and Sen 2009, Bai *et al*.

2013). Moreover, further problems are introduced in the same extend (Kumar 2008, Demaio and Mathew 2011, Aliquippa and Pennsylvania 2020). The knight moves are also utilized for image encryption (Hou *et al*. 2004, Delei *et al*. 2008, Philip 2013, Kumar and Nirmala 2015, Singh *et al*. 2015). Additional moves of single knights on a chess-board-likes with the legal chess moves were inspiring for the various versions of problems which are called the Knight Covering Problem (KCP, a.k.a. N-KCP) (Jackson and Pargas 1991, Fisher 2003, Lemaire 2003, Rubin 2003, 2004, 2005, 2007, Wei 2014). Although it has no known analytical solution, there are many approximations such as the independent set method (Güldal *et al*. , Güldal *et al*. 2019) and the Girvan-Newman clustering algorithm (Güldal 2019), and Modularity (Güldal 2020) employing knight graphs of KCP. In this study, the knight graph representation of 3-KCP is used as a case study to investigate the effect of scoring factor for the Leiden clustering algorithm. In Figure 1, the legal knights' moves are shown for every cell for 3×3 board. Thus, the corresponding graph is shown in Figure 2.

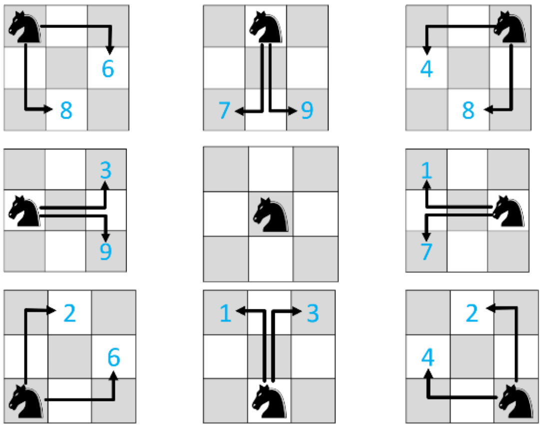
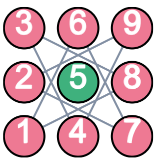


Figure 1. 3-KCP has 9 cells to position the knight.



Color code	Degree	Number of Nodes	Percentage in the graph (%)
green	0	1	11.11
pink	2	8	88.89

Figure 2. Graph representation of 3-KCP. The nodes are colored proportional to the degrees of nodes.

The 3×3 knight graph is shown color-coded with changing degree in Figure 2. Each vertex represents a cell on the board and is labeled by the corresponding index number. The cells on the edges (colored pink) can attack 2 cells. The cell on the center (colored green), namely cell 5, threats no other. Thus, the graph form of the 3×3 knight graph is composed of 9 vertices and 8 edges. The nodes have 2 degrees and no degree by the portion respectively 88.89% and 11.11%. Every knight and their relations on the board is explicitly shown in Figure 1. The graph density is 0.222, so the meaningful resolutions should be lower.

The details of the implementation and results are discussed in the following sections.

2. Method: Leiden Algorithm

We used the Leiden algorithm to identify the clusters for the 3-KCP. The clustering quality is measured by a score is calculated by Modularity and CPM. The formula for Modularity is as follows (Newman 2004):

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

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. 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. Lastly, γ is the resolution parameter.

The formula for CPM is as follows (Traag, Waltman, and van Eck 2019):

$$Q_{CPM} = \sum_{c} \left(\text{e}_c - \gamma \binom{n_c}{2} \right) \quad (2)$$

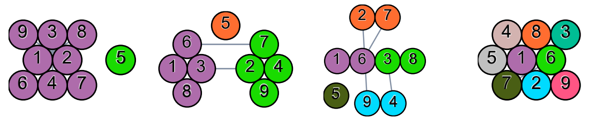
where n_c is the number of nodes in community c. e_c is the weight of edge c. The interpretation of the resolution parameter is represented by γ.

For our analysis, we used the Gephi 0.9.2 (Blondel *et al*. 2008, Lambiotte *et al*. 2008, Bastian *et al*. 2009) with Leiden Algorithm 1.0.0 plugin. The resolutions are limited from 0.1 to 4.0 for Modularity and 0.1 to 1.0 for CPM which are selected by the morphology of the 3-KCP graph. The analysis and implementation results will be given in the Results and Discussion section.

3. Results and Discussion

The investigated clustering information of the 3-KCP graph by Modularity and CPM score is extracted the number of communities from 1 to 9 with respect to changing resolution.

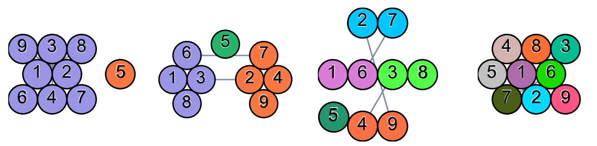
The Modularity scoring identifies the strong relationships between connected vertices. In Figure 3, the Leiden algorithm with Modularity results on 3-KCP (for resolution = 0.1 – 4.0) graphs are depicted. The resolution 0.1 is extracted 2 clusters which have no edge at all in the 3-KCP graph as shown in Figure 3.a. In Figure 3.b, 3 clusters are generated by Modularity scoring with the resolutions 0.5. For the resolution = 1.0 in Figure 3.c, the 3-KCP graph is divided into 5 clusters. For resolution 4.0, the 3-KCP is divided into 9 clusters.



a) Resolution = 0.1 b) Resolution = 0.5 c) Resolution = 1.0 d) Resolution = 4.0

Figure 3. Leiden algorithm with Modularity quality measurement is applied to 3-KCP graphs for the resolutions from 0.1 to 4.0. Increasing resolution divides 3-KCP graph the smaller clusters.

The CPM quality function provides a similar result with lower resolutions as shown in **Figure 4**.



a) Resolution = 0.1 b) Resolution = 0.2 c) Resolution = 0.3 d) Resolution = 1.0

Figure 4. Leiden algorithm with CPM quality measurement is applied to 3-KCP graphs for the resolutions from 0.1 to 1.0. Increasing resolution divides 3-KCP graph the smaller clusters.

The change of quality scores against the resolution is shown in Figure 5. The quality of the cluster decreases with the increasing resolution for both quality measurements. Thus, the presented 2 clusters as a divided network are the optimum division of the network. The other point, quality of the clusters is higher for Modularity for all clusters. Lastly, CPM has a rapid decrease of network quality score with increasing resolution, while the Leiden algorithm divides the whole graph into smaller clusters. Thus, our analysis shows CPM is more promising method for the resolution testing. This agrees with the available literature (Traag *et al*. 2011).

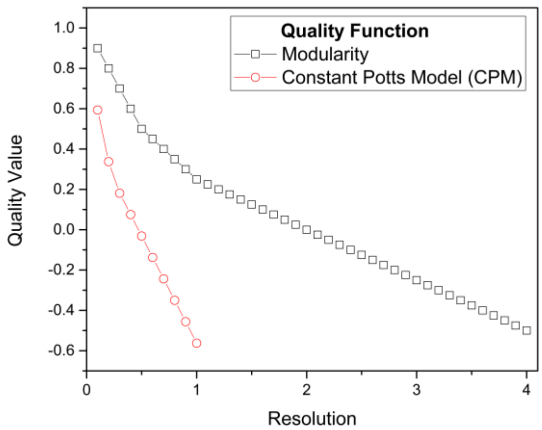


Figure 5. (Color online) Modularity and Constant Potts Model presents significant differences for 3-KCP.

The applied resolutions for the Modularities are between 0.1 to 4.0 to cluster the 3-KCP graph. In Figure 6, the increasing resolutions divide the network into greater numbers of clusters. The Modularity quality score to identify the quality of exhibited clusters does not correlate with the number of clusters. The highest Modularity score is 0.9 (for resolution 0.1) and, the lowest Modularity score is -0.5 for the resolution 4.0.

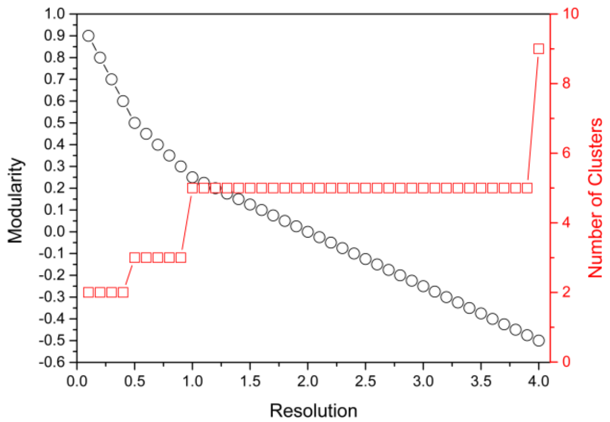


Figure 6. (Color online) Leiden algorithm is applied with Modularity as quality measurement.

In Figure 7, CPM scoring is presented with changing resolution from 0.1 to 1.0. Although it shows a similar trend with the Modularity quality score, it speeds up the clustering process. The continuous decrease in the resolution score, from 0.59375 to -0.5625, shows that 2 clusters are the most appropriate division of the network.

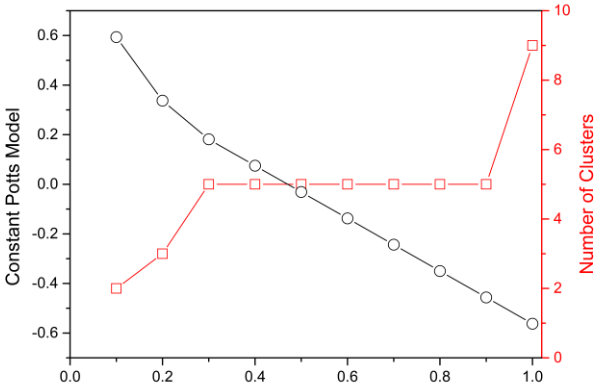


Figure 7. (Color online) Leiden algorithm is applied with Constant Potts Model as quality measurement.

4. Conclusion

In this study, we have applied the Leiden algorithm with Modularity and CPM network quality scoring factor into 3×3 knight graph. Therefore, we compared the specified quality scoring factors for the dense and regular graph by means of Leiden algorithm. The analyses show resolution 0.1 is the computationally optimal resolution to find the best clusters for both quality factors. The Modularity has a higher quality score, 0.9, than CPM, 0.59375, for the resolution 0.1. The continuous decrease in the quality of clusters observed for both cases regards the increasing resolution. On the other hand, the CPM is more receptive to the resolution change, so our analysis agrees with the previous studies.

Based on our analysis, the CPM is a promising method to speed up clustering the network into smaller clusters with a lower range of the resolution change.

Acknowledgment

The authors declare no conflict of interest.

5. References

Ahlgren P, Chen Y, Colliander C, van Eck NJ. 2019. Community Detection Using Citation Relations and Textual Similarities in a Large Set of PubMed Publications. Paper presented at the ISSI.

Aliquippa P, Pennsylvania, 2020. Thematic Knight's Tour Quotes

Bai S, Liao X, Qu X, Liu Y. 2006, 3-6 Nov. 2006. Generalized Knight's Tour Problem and Its Solutions Algorithm. Paper presented at the 2006 International Conference on Computational Intelligence and Security.

Bai S, Zhu G, Huang J. 2013, 14-15 Dec. 2013. An Intelligent Algorithm for the (1,2,2)-Generalized Knight's Tour Problem. Paper presented at the 2013 Ninth International Conference on Computational Intelligence and Security.

Bastian M, Heymann S, Jacomy M. 2009. Gephi: an open source software for exploring and manipulating networks. Paper presented at the International AAAI Conference on Weblogs and Social Media.

Blondel VD, Guillaume J-L, Lambiotte R, Lefebvre EJJosmt, experiment, 2008. Fast unfolding of communities in large networks. **2008** (10), P10008.

Boy JDJJoOSS, 2020. textnets: A Python package for text analysis with networks. **5** (54), 2594.

Clifford T, Bruce J, Matta J. 2019. Using node-based resilience clustering to predict and analyze medical data. Paper presented at the 2019 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB).

Delei J, Sen B, Wenming D. 2008, 12-14 Dec. 2008. An Image Encryption Algorithm Based on Knight's Tour and Slip Encryption-Filter. Paper presented at the 2008 International Conference on Computer Science and Software Engineering.

Delvenne J-C, Schaub MT, Yaliraki SN, Barahona M. 2013. The stability of a graph partition: A dynamics-based framework for community detection. In Dynamics On and Of Complex Networks, Volume 2, 221-242, Springer.

Demaio J, Mathew B, 2011. Which Chessboards have a Closed Knight's Tour within the Rectangular Prism? *Electr. J. Comb.*, **18**. doi:10.37236/495

Devi JC, Poovammal EJPCS, 2016. An analysis of overlapping community detection algorithms in social networks. **89** 349-358.

Dong R, Yuan G-CJBb, 2020. GiniClust3: a fast and memory-efficient tool for rare cell type identification. **21** 1-7.

Esmailian P, Jalili M, 2015. Community Detection in Signed Networks: the Role of Negative ties in Different Scales. *Scientific Reports*, **5** (1), 14339. doi:10.1038/srep14339

Fisher DC, 2003. On the n x n Knight Cover Problem. *Ars Comb.*, **69**.

Gibbs H, Liu Y, Pearson CA, Jarvis CI, Grundy C, Quilty BJ, Diamond C, Eggo RMJNc, 2020. Changing travel patterns in China during the early stages of the COVID-19 pandemic. **11** (1), 1-9.

Girvan M, Newman MEJ, 2002. Community structure in social and biological networks. **99** (12), 7821-7826. doi:10.1073/pnas.122653799 %J Proceedings of the National Academy of Sciences

Güldal S. (2019). Connectives of Knights Covering Problem By Girvan-Newman Clustering. Paper presented at the SDPS 2019 Workshop, Madrid, Spain.

Güldal S, 2020. Identification of Knights' Relations For 5×5 Knight Graph by Modularity. *Journal of Engineering Science of Adiyaman University* **7**(13), 104-114.

Güldal S, Lipscomb M, Tanik MM. (2019). Solving Knights Covering Problem: Backtracking, Permutation, Bipartite Graph, and Independent Set. Paper presented at the Nineteenth Annual Early Career Technical Conference, Birmingham, Alabama USA.

Güldal S, Tanik MM, Lipscomb MM. Solving Knights Covering Problem by a Hybrid Algorithm. Paper presented at the IEEE SouthEastConn, Huntsville, Alabama.

Hingston P, Kendall G. (2004). Ant Colonies Discover Knight's Tours (Vol. 3339).

Hou QB, Yang XF, Wang YS, Huang XS, 2004. An image scrambling algorithm based on wavelet transform and Knight's tour. **41,** 369-375.

Jackson AH, Pargas RP, 1991. Solutions to the NxN Knights Covering Problem. *Journal of Recreational Mathematics* **23,** 255-267.

Jian H, Sen B. 2009, 7-8 March 2009. An Efficient Algorithm for the Generalized (1,k)-Knight's Tours Problem. Paper presented at the 2009 First International Workshop on Education Technology and Computer Science.

Kumar A, 2008. Non-crossing Knight's Tour in 3-Dimension.

Kumar J, Nirmala S. 2015, 10-13 Aug. 2015. Securing the contents of document images using knight moves and genetic approach. Paper presented at the 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI).

Lambiotte R, Delvenne J-C, Barahona MJapa, 2008. Laplacian dynamics and multiscale modular structure in networks.

Lemaire B, 2003. Knights Covers on NxN Chessboards. *J. Recr. Math.*, **31,** 87-99.

Li B, Gould J, Yang Y, Sarkizova S, Tabaka M, Ashenberg O, Rosen Y, Slyper M, Kowalczyk MS, Villani A-CJNM, 2020. Cumulus provides cloud-based data analysis for large-scale single-cell and single-nucleus RNA-seq. **17** (8), 793-798.

Miura T, Asatani K, Sakata I. 2020. Classifying Sleeping Beauties and Princes Using Citation Rarity. Paper presented at the International Conference on Complex Networks and Their Applications.

Newman MEJ, 2004. Analysis of weighted networks. *Physical Review E*, **70** (5), 056131. doi:10.1103/PhysRevE.70.056131

Parberry I, 1997. An Efficient Algorithm for the Knight's Tour Problem. *Discrete Applied Mathematics*, **73,** 251-260. doi:10.1016/S0166-218X(96)00010-8

Pasimeni F, 2020. The Origin of the Sharing Economy Meets the Legacy of Fractional Ownership.

Philip A, 2013. A Generalized Pseudo-Knight?s Tour Algorithm for Encryption of an Image. *IEEE Potentials*, **32** (6), 10-16. doi:10.1109/MPOT.2012.2219651

Rubin F, 2003. Improved Knight Coverings. *Ars Combinatoria*, **69**.

Rubin F. (2004). Knight Covers for the 50x50 Chessboard. Paper presented at the Mathfest 2004, Providence RI.

Rubin F, 2005. A Family of Efficient Knight Covering Patterns. *Journal of Recreational Mathematics*, **33** (3), 165-175.

Rubin F, 2007. An Improved Method for Finding Knight Covers. *Ars Combinatoria*, **82**.

Singh M, Kakkar A, Singh M, 2015. Image Encryption Scheme Based on Knight's Tour Problem. *Procedia*

Computer Science, **70,** 245-250. doi:10.1016/j.procs.2015.10.081

Traag VA, Van Dooren P, Nesterov YJPRE, 2011. Narrow scope for resolution-limit-free community detection. **84** (1), 016114.

Traag VA, Waltman L, van Eck NJ, 2019. From Louvain to Leiden: guaranteeing well-connected communities. *Scientific Reports*, **9** (1), 5233-5233. doi:10.1038/s41598-019-41695-z

van Laarhoven T, Marchiori EJC, 2013. An axiomatic study of objective functions for graph clustering.

Van Laarhoven T, Marchiori EJTJoMLR, 2014. Axioms for graph clustering quality functions. **15** (1), 193-215.

Wang C, Wang F, ssOnega TJTiG, Network optimization approach to delineating health care service areas: Spatially constrained Louvain and Leiden algorithms.

Wei F, 2014. Research on Knight Covering Based on Breadth First Search Algorithm. *Applied Mechanic and Materials*, **686,** 377-380.