RESEARCH ARTICLE OPEN ACCESS

Application of ANN Based Graph Theory in Wireless Sensor Networks

D. Jeni Jeba Seeli [1], Dr.K.K.Thanammal [2]

[1] Asst Professor Department of Computer Science Scott Christian College - Nagercoil
[2] Associate Professor Department of Computer Science S.T.Hindu College - Nagercoil

ABSTRACT
Numerous learning approaches demand adopting graph data that comprises large relation information within elements. Graph theory based approaches are widely used in networks for the efficient identification of shortest path. In learning of non-structural data including texts as well as images, the reasoning on extracted structures is regarded as a major research area which also demands techniques with graph reasoning. Graph theory models are efficient approaches which capture the graph dependence through transmitting of message among the nodes of graphs. Graph neural networks maintain a state which represents neighborhood information with random depth. Even though traditional graph based approaches are tedious for training in a particular point, present developments in network models, optimization approaches, as well as parallel computation have generated improved learning with them.
Keywords:- Artificial Neural Networks, Wireless Sensor Networks, signal-to-interference-plus-noise ratio, shortest path, Multipartite graph.

I. INTRODUCTION

Networks are commonly utilized for the construction as well as organization of wide range of objects denoting relationships in interconnected world. In particular, biological networks, social networks, transportation networks, as well as bibliographic information networks are facing increasing demand [1]. Owing to the ability of networks to provide the physical layer's secrecy, information-theoretic security has gained broad attention of the research community. Specifically, cooperative transmission methodology has become an enhanced technique to increase the physical-layer security of networks, generally when the state of channel between the nodes is worse [2].

Researchers have widely examined the rigidness of complicated networks and several methods have been introduced in which graph theory based methods are considerable ones. A complicated network is modelled as a graph comprised of nodes as well as edges in which the nodes indicate the elements of the focal network as well as the edges indicate the interactions among the elements [3]. The early analysis of graph data is generally performed by network embedding, and the basic concept behind several embedding methods is to utilize the techniques for the reduction of dimension. The high dimensional topology structure information as well as the relationship among nodes has to be distilled into a dense representation space [4]. Graphs model various categories of relations and connections and because of the improvements in graph theory simulation tools, it is obtaining increased momentum regarding social networks, computer networking, as well as security [5].

At present, Graph Convolutional Networks (GCN) are utilized for image-based convolution and convolution is performed by the integration of all neighboring nodes at every place. Spectral GCNs initially convert graphs to corresponding spectrum and further performs convolution and convolutional kernel developed in spectral domain by convolution theorem. It frequently interprets the graph from the viewpoint of graph node, but the significance of graph edges gets neglected, generally when graph edge has significant physical meanings in certain events [6]. Computer networks are generally denoted as neural network graphs are not developed for learning graph-structured information. Hence, the models trained generated restricted accuracy and do not generalize regarding topologies or routing configurations [7].

Distributed line graphs possess certain beneficial parameters similar to network topology, like out-regular indicating every user has a fixed size of routing table and small diameter indicating reasonably minimal hop count is needed to attain a target user. It indicates simple identification of new path but certain edges can be neglected [8]. Another graph embedding approach, named Deep Direct, is proposed which preserves network topology but requires further improvement [9]. Location-Based Social Networks (LBSN) graph-embedding method RLINE depending on a large-scale information-network-embedding approach in which side information is efficiently embedded, like the user's social information and geographical distance information, However, in LBSNs, traditional recommendation approaches, like collaborative filtering, are not suitable to complicated scenarios [10].

Cascading faults graph (CFG) is another approach that adopts graph theory considering the physical, operational as well as structural attributes of electrical networks. The CFG reveals scale-free features of the nature of propagations of fault in transmission network by converting the electrical physical network to relationships of branches in cascading failures; Due to this, certain parameters of CFG are unstable [11]. Directed graphs are also utilized for graph based applications in which the outputs for construction of graphs is known with a target directed degree series yet directed degree correlation is not previously defined [12]. Graph-dependent SSL approaches specifically consider that true labels are "smooth" in graph, which commonly encourages leveraging the network topology to proliferate labels thus increasing learning performance [13]. Knowledge Graph Embedding (KGE) dependent techniques embed entities as well as relations to low-dimensional distributed representations relying on existing triples in KGs. Entity embedding as well as relation embedding are attained with the optimization of a scoring function of every fact to estimate its plausibility [14]. Graph partitioning (GP) is utilized in enhanced large-scale parallel graph algorithms which is an approach adopted by web search engines for ranking user's web pages. When networks increase, a parallel evaluation demanding a sensible distribution of the graphs is needed [15].

In this paper, the graph theory based approaches are studied considering the limitations in various existing approaches. The determination of shortest path in a network is analyzed and the accuracy is estimated.

II. GRAPH THEORY BASED ANN FOR WIRELESS NETWORK

A. Graph Theory

The graph theory concept has been introduced among the seven Konigsberg bridges along with Euler. It is utilized in different branches of Mathematics, Computer Science, Electrical and Electronics Engineering, Electronics and Communication Engineering, and Operations Research to find communities in networks. A graph is made up of two sets (V, E) with V represents the group of vertices, E denotes the group of edges created by pair of vertices. Adopting graph, the nodes connected to the networks are indicated as group of points, lines are communicated by means of link. Different application depending on graph tree are Binary tree, Reliable communication networks, Assignment of radio frequencies, Latin Square, Membership problem, Mathematical analysis etc.

Graphs are a category of Non-Euclidean data, existing in 3D, and contrast to data types like images, text, and audio. Graphs possess specific properties limiting the possible actions as well as analysis to be performed on them.

Graph neural networks are exploited in broad range of problem domains in supervised, semi-supervised, unsupervised and reinforcement learning settings. The applications are divided into three scenarios: (1) Structural scenarios in which the data has explicit relational structure, like physical systems, molecular structures and knowledge graphs; (2) Non-structural scenarios in which the relational structure is not explicit including image, text, etc. (3) Other application scenarios like generative models and combinatorial optimization issues.

B. ANN

Neural Networks are termed as universal function approximates. Different architectures are adopted for approximating any nonlinear function. Numerous architectures permit generation of functions of adverse complexity as well as power. Figure 1 represents the ANN with Network size: n x m x r = 2x5x1

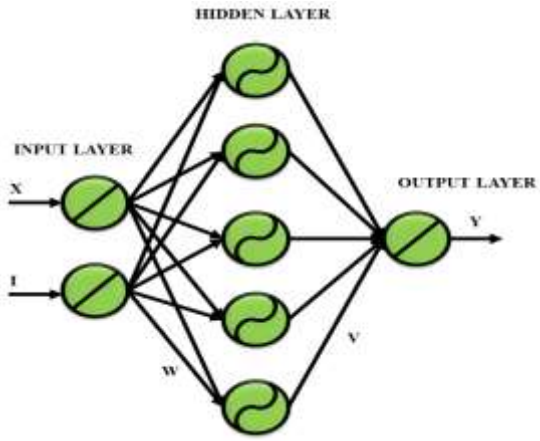

Fig.1. Artificial Neural Network

Input layer: Count of neurons in this layer depends on the total inputs to the neuronal network. This layer comprises of passive nodes that do not involve in the actual signal modification, yet performs transmission of signal to the following layer.

Hidden layer: This layer has random number of layers with random number of neurons. The nodes in this layer performs signal modification and they are active.

Output layer: The count of neurons in the output layer denotes the total output values of the neural network. In this layer the nodes are active ones.

C. System Model

Several transmission control problem has been detected while transmitting the information or power through the k-user interference channel with K-single antenna transceiver pair. The signal received by the k-th receiver is given as follows,

$$y_k = h_{kk} s_k + \sum_{j \neq k} h_{kj} s_j + n_k \qquad (1)$$

Where $h_{kk} \epsilon$ C represents the channel link among the k-th transmitter and receiver, h_{kj} ϵC represents the channel cross-link between the transmitter j and the receiver k, s_k ϵC represents the k-th receiver data symbol and the Gaussian noise is represented as $n_k \sim CN(0, \sigma_k^2)$.

The signal-to-interference-plus-noise ratio (SINR) of the k-th receiver is given as,

$$SINR = \frac{|h_{kk}|^2 p_k}{\sum_{i \neq k} |h_{kk}|^2 p_i + \sigma_k^2} \quad (2)$$

Where $p_k = E[s_k^2]$ is the power of the k-th transmitter and $0 \leq p_k \leq p_{max}$, p = $[p_k \ldots p_k]$ denotes the power allocation vector. The main objective is to maximize the weighted sum rate by finding the maximum power allocation. The problem is represented as,

Maximize p $\sum_{k=1}^{K} w_k log_2 (1 + SINR_k)$ (3)

Where p_k ranges from $0 \leq p_k \leq p_{max}$, w_k is the weight of k-th pair. Thus the channel matrix is defined as H = $[h_k \ldots h_k]^T$ and $h_i = [h_i \ldots h_{ki}]^T$, i = 1,…k. Therefore, the GNN method is used to solve the input –output mapping issues.

D. Graph representation and geometric properties

The K-user interference channel has been modelled as a complete graph vertex and edge labels in this section. The i-th transmitter-receiver pair is referred to as the i-th vertex. The vertex mark includes the weight of the i-th pair as well as the state of the direct channel and it is given as (h_{ii}, w_i). An interference connection is indicated by one edge between two vertices, with tag as the states of the interference channels h_{ij} and h_{ji}. Figure 2 depicts the 3-user interference graph.

By checking the map from the channel matrix and weights to the optimal power control, the geometry of the interference channel has been discussed.

Proposition 1: Let fi() denotes the function that maps the channel matrix and weights to the optimal power allocation of the i-th transmitter, i.e., $p_i^* = f_i(H, w)$ and let $\boldsymbol{\Pi}$ represents any permutation matrix satisfying $f_i(\Pi^T H \boldsymbol{\Pi})_{ij} = h_{ij}$.Therefore,

$$p_i^* = f_i(H, w) = f_i(\Pi^T H \boldsymbol{\Pi}, \Pi^T \boldsymbol{w}).$$

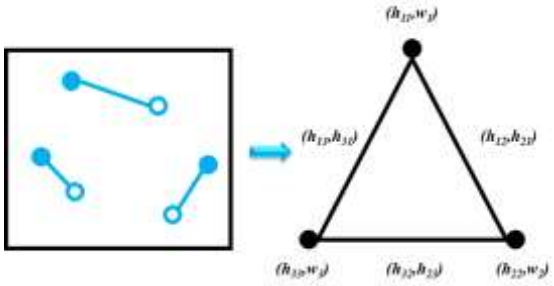

Fig.2. Graphical representation of 3-usr interference channel

This is translated as the disordered property of interference channels: what matters is the set of interference channel coefficients rather than their ordering. The permutation irrelevance of the channel is due to the irrelevance of the ordering. Since the elements are no longer similar to each other after the permutation, this property implies that only considering the neighborhood elements as in CNN is meaningless as the elements are not close to each other for a long time after permutation.

Thus this invariance property denotes that all the edges with the same end node are homogeneous, allowing us to share weights across all of the node's edges. Therefore, we can limit the hypothesis space of the constructed neural network for a single node to the space of fixed functions resulting in graph based neural networks.

E. Graph theory based ANN

Artificial Neural Networks are generally just graphs and NNs are a special graph, having the same structure and hence similar terminology, concepts, and rules. The specific type of graph that NNs most resemble are multipartite graphs. A multipartite graph is a graph which is split into various groups of nodes. The nodes in every set can share edges among sets, but not within every set.

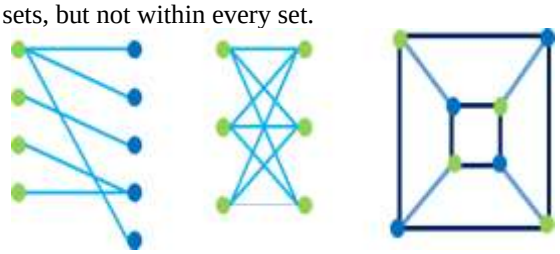

Fig.3. Isomorphic bipartite graphs (Courtesy of Wolfram Math World)

The ANN based Graph Theory is for detecting the shortest path in Wireless Sensor Networks. A graph is represented by G = (V, E), where V = $\{v_i | i=1\ldots, N\}$ is a node set with N elements, E = $\{e_{ij} | v_i, v_j$ are connected$\}$ is an edge set comprising all edges of G. Several approaches are utilized for facilitating the explanation of graph neural network.

1. **Path between Two Edges**: A path is a group of specific nodes as well as edges connecting two edges in a graph [Fig. 4(a)]. The number of paths between two edges may be more than two.
2. **Length of Path between Two Edges**: Consider a pair of edges, numerous paths have various count of nodes as well as edges. For a specific path, the count of contained nodes is denoted as its length. Paths with various lengths are given in Fig. 4(b).
3. **Shortest Path between Two Edges**: Consider a pair of edges, within all paths connecting the two edges, the path with the minimal length is denoted as the shortest path between the two edges as in Fig. 4(b).
4. **Distance Between Two Edges**: Consider a pair of edges, the length of the shortest path indicates the distance between the two edges, denoted as D (·,·) as in Fig. 4(b).

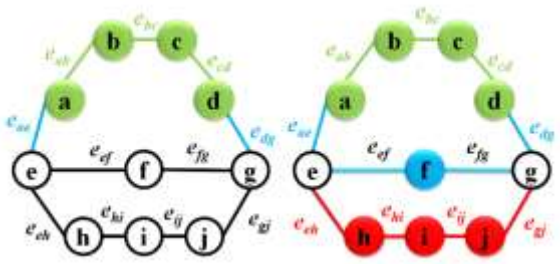

Fig.4. (a) and (b) path between e_{ae} and e_{dg}

Fig.4. (a) Blue parts (nodes a–d and edges e_{ab}, e_{bc}, and e_{cd}) form a path between e_{ae} and e_{dg} with a length of 2. Fig.4.(b) Parts in blue, green, as well as mauve indicate three different various between e_{ae} and e_{dg}. Node e and g have two colors as they are shared by two paths. The lengths of the blue, green, and mauve path are 4, 3, and 5, respectively. The green path is the shortest path between e_{ae} and e_{dg}. Thus, the length of the green path, which is 3, is the distance between e_{ae} and e_{dg}.

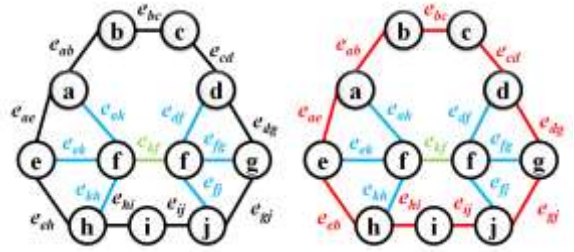

Fig.5.(a) 1-hop neighbors of the edge e_{kf}. (b) Both 1- and 2-hop neighbors of edge e_{kf}.

For performing convolution, receptive field is required, termed as neighborhood and for a particular edge e_{pq} it is defined as,

$$N(e_{pq}) = \{ e_{kl} \mid e_{kl} \in E \text{ and } D(e_{pq}, e_{kl}) \leq R \}$$

Where, $D(e_{pq}, e_{kl})$ is the distance among the root edge e_{pq} and edge e_{kl} and R is an integer indicating maximum distance among the root edge and adjacent edges. Hence, the neighborhood of a root edge is also expressed as edges within a particular distance R from the root edge. In our convolution, we set R as 1 and the attained neighborhood has only the edges which are connected to the root edge by a single node, i.e., the neighborhood of an edge e_{pq} is,

$$N(e_{pq}) = \{ e_{kl} \mid e_{kl} \in E \text{ and } D(e_{pq}, e_{kl}) \leq 1$$

The convolution at a particular edge e_{pq} is a weighted summation on all the adjacent edges. Generally, the convolution output at edge e_{pq} is given by

$$e_{pq}^{out} = \sum_{e_{kl} \epsilon N(e_{kl})} e_{kl} \cdot w(l(e_{kl})) \quad (4)$$

Where, e_{pq}^{out} is the convolution output at the edge e_{pq}, e_{kl} denotes all adjacent edges being added, and $w(l(e_{kl}))$ are the respective weights.

Adjacent edges in a graph not possessing an implicit spatial order like pixels in an image. For assigning weights to adjacent edges, the edges are initially sorted following the use of labelling function l to align each edge with a corresponding weight in order. Assign a labelling value $I(e_{kl})$ to each edge e_{kl} in the neighborhood, denoting its order, and a weight will be assigned according to $I(e_{kl})$. We can't generate particular count of weights for edges in a graph, unlike photos, because the count of adjacent edges can vary.

Hence, instead of providing every adjacent edge a unique labeling value, function I will map adjacent edges to particular count of subsets, as well as edges in the same subset will possess similar labeling value, i.e., $I: N(e_{pq}) \rightarrow \{1,\ldots, K\}$. Label every edge in the neighborhood with an integer ranging from 1 to K as its order. As a consequence, even if the count of adjacent edges is not set, K weight values can always be assigned to them. The weight function w assigns weights on the basis of labeling values. After multiplying by the weights, the summation for all adjacent edges is performed, and the convolution operation is completed. It is easy to extent our model to graphs with multiple channels by replacing the edge feature and weight with the edge feature vector and weight vector, similar to the image convolution described before.

Normalizing term is added to the equation to balance the contribution of edges with various weights.

$$e_{pq} = \sum_{e_{kl} \epsilon N(e_{pq})} \frac{1}{Z_{pq}(e_{pq})} e_{kl} \cdot w(l(e_{kl})) \quad (5)$$

$Z_{pq}(e_{pq})$ Represents the count of adjacent edges with similar labeling value as e_{kl} in the edge e_{pq}'s neighborhood, that is

$$Z_{pq}(e_{kl}) = \{ e_{mn} \mid e_{mn} \epsilon N(e_{pq}) \text{ and } l(e_{mn}) = l(e_{kl}) \} \quad (6)$$

Thus the adjacent edges are not partitioned evenly in to K subsets, $(1/Z_{pq}(e_{kl}))$ is the term utilized to offset the contribution of adjacent edges with various labeling values.

III. RESULTS AND DISCUSSION

Table 1 reports the classification accuracy for each class for the approaches like node convolution, edge convolution, SLHM, BPLHM. Various classes in which the graph theory based approaches significantly exceed other existing approaches are considered and the corresponding graph is plotted in figure 5.

Table 1: Comparison of Accuracy

Methods	Accuracy
Node-convolution	0.85
Graph theory based ANN	0.94
SLHM	0.88
BPLHM	0.92

Fig.6. Comparison of Accuracy

Fig.6. indicates the comparison of relative performance with noise variance. It shows that the relative performance is 91.5% when the relative noise variance is 10%.

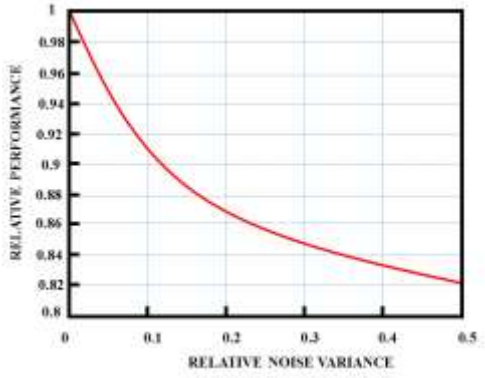

Fig.7. Comparison of relative performance with noise variance

Thus the proposed graph theory based approach not only exceeds the performance of existing methods based on learning, but also outputs enhanced accuracy when compared to various optimization-based approaches. It is also rigid towards noisy channel state information

IV. CONCLUSION

In this paper, graph theory based methods are analyzed which effectively detects the path of shortest length in the network. The improved mobility of every node for transmitting information from the source node to the destination node is evaluated by representing every edges with the integration of all neighbor edges. The graph theory based approaches exhibited advantages including scalability, ability to incorporate instantaneous channel state information, and the capability to tackle weighted problem. The accuracy values of graph based approaches are compared and the efficiency is analyzed.

REFERENCES

[1] Jinli Zhang, Zongli Jiang, Zheng, Chen, Xiaohua Hu, *"WMGCN: Weighted Meta-Graph Based Graph Convolutional Networks for Representation Learning in Heterogeneous Networks",*IEEE Access, Vol: 8, pp: 40744 – 40754, 2020.

[2] Chensi Zhang, Jianhua Ge, Zeyu Xia, Haoyu Du, *"Graph Theory Based Cooperative Transmission for Physical-Layer Security in 5G Large-Scale Wireless Relay Networks"*, IEEE Access, Vol: 5, pp: 236 – 3251, 2017.

[3] Wei Wu, Qingzheng Xu, Zhe Liu, Na Wang, *"A Probabilistic Theory Based Method for Robustness Assessment of Bipartite Networks",* IEEE Access, Vol: 7, pp: 35359 – 35369, 2019.

[4] Zhenghao Zhang, Jianbin Huang, Qinglin Tan, *"SR-HGAT: Symmetric Relations Based Heterogeneous Graph Attention Network",* IEEE Access, Vol: 8, pp: 165631 – 165645, , 2020.

[5] Mosheer J. Daas, Mohammad Jubran, Mohammed Hussein, *"Energy Management Framework for 5G Ultra-Dense Networks Using Graph Theory",* IEEE Access, Vol: 7, pp: 175313 – 175323, 2019.

[6] Xikun Zhang, Chang Xu, Xinmei Tian, Dacheng Tao, *"Graph Edge Convolutional Neural Networks for Skeleton-Based Action Recognition",* IEEE Transactions on Neural Networks and Learning Systems, Vol: 31, No: 8, pp: 3047 – 3060, 2020.

[7] Krzysztof Rusek, José Suárez-Varela, Paul Almasan, Pere Barlet-Ros, Albert Cabellos-Aparicio, *"RouteNet: Leveraging Graph Neural Networks for Network Modeling and Optimization in SDN",* IEEE Journal on Selected Areas in Communications, Vol: 38, No: 10, pp: 2260 – 2270, 2020.

[8] Akiko Manada, Hiroyoshi Morita, *"Graph Theoretical Analysis on Distributed Line Graphs for Peer-to-Peer Networks",* IEEE Transactions on Network Science and Engineering, Vol: 6, No: 3, pp: 326 – 335, 2019.

[9] C. Wang, C. Wang, Z. Wang, X. Ye, J. X. Yu, and B. Wang, *"Deepdirect: Learning directions of social ties with edge-based network embedding",* IEEE Transactions on Knowledge and Data Engineering, Vol: 31, No: 12, pp: 2277–2291, 2018.

[10] Kang Yang, Jinghua Zhu, Xu Guo, *"POI neural-rec model via graph embedding representation",* Tsinghua Science and Technology, Vol: 26, No: 2, pp: 208 – 218, 2021.

[11] Xiaoguang Wei, Shibin Gao, Tao Huang, Ettore Bompard, Renjian Pi;Tao Wang, *"Complex Network-Based Cascading Faults Graph for the Analysis of Transmission Network Vulnerability",* IEEE Transactions on Industrial Informatics, Vol: 15, No: 3, pp: 1265 – 1276, 2019.

[12] Bálint Tillman, Athina Markopoulou, Minas Gjoka, Carter T. Buttsc, *"2K+ Graph Construction Framework: Targeting Joint Degree Matrix and Beyond",* IEEE/ACM Transactions on Networking, Vol: 27, No: 2, pp: 591 – 606, 2019.

[13] Vassilis N. Ioannidis, Antonio G. Marques, Georgios B. Giannakis, *"Tensor Graph Convolutional Networks for Multi-Relational and Robust Learning",* IEEE

Transactions on Signal Processing, Vol: 68, pp: 6535 – 6546, 2020.

[14] Xiyang Liu, Huobin Tan, Qinghong Chen, Guangyan Lin, *"RAGAT: Relation Aware Graph Attention Network for Knowledge Graph Completion",* IEEE Access, Vol: 9, pp: 20840 – 20849, 2021.

[15] Henning Meyerhenke, Peter Sanders, Christian Schulz, "Parallel Graph Partitioning for Complex Networks", IEEE Transactions on Parallel and Distributed Systems, Vol: 28, No: 9, pp: 2625 – 2638, 2017.