Acta Universitatis Sapientiae

Informatica

Volume 12, Number 1, 2020

Sapientia Hungarian University of Transylvania
Scientia Publishing House

Acta Universitatis Sapientiae, Informatica is covered by the following services:

DOAJ (Directory of Open Access Journals)
EBSCO (relevant databases)
EBSCO Discovery Service
io-port.net
Japan Science and Technology Agency (JST)
Micosoft Academic
Ulrich's Periodicals Directory/ulrichsweb
Web of Science – Emerging Sources Citation Index
Zentralblatt für Mathematik

Contents

Acta Univ. Sapientiae, Informatica **12,** 1 (2020) 5–21

DOI: 10.2478/ausi-2020-0001

Efficiency centrality in time-varying graphs

Péter MARJAI
Eötvös Loránd University
Budapest, Hungary
email: g7tzap@inf.elte.hu

Attila KISS
J. Selye University
Komárno, Slovakia
email: kissae@ujs.sk

Abstract. One of the most studied aspect of complex graphs is identifying the most influential nodes. There are some local metrics like degree centrality, which is cost-effective and easy to calculate, although using global metrics like betweenness centrality or closeness centrality can identify influential nodes more accurately, however calculating these values can be costly and each measure has it's own limitations and disadvantages. There is an ever-growing interest in calculating such metrics in time-varying graphs (TVGs), since modern complex networks can be best modelled with such graphs. In this paper we are investigating the effectiveness of a new centrality measure called efficiency centrality in TVGs. To evaluate the performance of the algorithm Independent Cascade Model is used to simulate infection spreading in four real networks. To simulate the changes in the network we are deleting and adding nodes based on their degree centrality. We are investigating the Time-Constrained Coverage and the magnitude of propagation resulted by the use of the algorithm.

1 Introduction

In recent years, there is an ever-growing interest in finding the best propagator in complex networks. The importance of a node is a basic subject when one

Computing Classification System 1998: E.1, G.2.2
Mathematics Subject Classification 2010: 60J60, 37M05, 05C82, 91D30
Key words and phrases: efficiency centrality, time-varying graphs, time-constrained coverage

is analyzing the organization and the structure of networks, because such mechanisms like spreading control [13] and self-similarity [17] can be managed by a few crucial nodes of the network.

Over the past years numerous centrality measures of identifying the most important nodes and analyzing the spreading dynamics was introduced, [5, 7]. Degree centrality (DC) [8] is a simple metric, however it is not a global algorithm and does not examine the structure of the network. Closeness centrality (CC) [6] examines the structure of the whole network however it is incapable in the case of large scale networks. The same applies to the Betweenness centrality (BC) [5] as well. There are some other approaches like Page rank (PR) [15], Eigenvector Centrality (EVC) [2] and many more.

The above mentioned measures only investigate static networks, not taking into account other aspects. In [3] the connections between the centrality measure and the types of the network flow are outlined. In the past couple years there is an ever-increasing interest to study the dynamics and centrality measures in dynamic complex networks. These networks change over time, edges and nodes are deleted and added which can alter the values calculated by the centrality measures. There are some different models for changing graphs mentioned in [20] like taking snapshots, studying the whole graph etc.

2 Related work

In this paper we study a new algorithm proposed in [18] which focuses on the influence of each node. With this measure, we identify the most influential nodes, then we examine the propagation of the infection through the dynamically changing graph based on the degree centrality of the nodes and using the Independent Cascade Model used in [10]. After that we inspect Time-Constrained Coverage introduced in [4] and the rate of the infected and non infected nodes.

The organization of the rest of this paper is as follows. In section 3 the definition of graph and the used centrality measures are briefly presented, the investigated Efficiency centrality is illustrated. The model of infection propagation, a brief summary of TVGs and the changing of the initial graph is also explained here. Section 4 explains the conducted experiments in detail and numerical examples of real-life networks are shown to compare the efficiencies of the measures. Section 5 is the conclusion of this paper and discuss what could be done to further investigate the matter at hand.

3 Concepts and problems

3.1 Existing centrality measures

For ease of reference consider an undirected simple network as $G = (V, E)$, where V represents the set of nodes, while E is the set of edges that connect the nodes. The number of the nodes is expressed as $N = |V|$, the number of the edges as $M = |E|$. The centrality measures DC, CC, EC are defined as follows.

3.1.1 Degree centrality (DC)

Degree centrality is defined as the number of incident edges. The degree centrality of node i, expressed as k_i, is defined as:

$$k_i = \sum_{j}^{N} x_{ij} \tag{1}$$

where j indicates the nodes that are connected to i, and x_{ij} represents the link between i and j. The value of x_{ij} is 1 if there is a link between i and j, and 0 otherwise.

3.1.2 Closeness centrality (CC)

The average length of the shortest path between a node and all other nodes is the normalized closeness centrality value of the node, in case of a connected graph. Closeness was defined by Alex Bavelas (1950) [1] as the reciprocal of the farness:

$$c_i = \frac{1}{\sum_{j}^{N} d(i,j)} \tag{2}$$

where $d(i,j)$ is the distance between nodes i and j.

3.1.3 Eigenvector centrality (EVC)

In graph theory, eigenvector centrality is used as a measure of the influence of a node in a network. Based on the concept that connections to high-scoring nodes contribute more to the score of the node in question than equal connections to low-scoring nodes, relative scores are assigned to the nodes in the network. A high eigenvector score means that a node has links to many nodes who themselves have high scores. [14]

For a given graph G let $A = \{a_{uv}\}$ be the adjacency matrix, i.e. $a_{uv} = 1$ if node u is linked to node v, and $a_{uv} = 0$ otherwise. The relative centrality, x_u, score of node u can be defined as:

$$x_u = \frac{1}{\lambda} \sum_{v \in N(u)} x_v = \frac{1}{\lambda} \sum_{v \in G} a_{uv} x_v \tag{3}$$

where $N(u)$ is a set of the neighbours of u and λ is a constant. With a small rearrangement this can be rewritten in vector notation as the eigenvector equation:

$$\mathbf{Ax} = \lambda \mathbf{x} \tag{4}$$

3.2 Efficiency centrality (EC)

3.2.1 Overview

To this point, many centrality measures were proposed to find the most influential nodes in a network, however each has their drawbacks. Measuring the degree of a node is improper to correctly find the most influential node since a node having a few highly influential neighbours may can propagate the information better than a node with lots of not influential nodes. Eigenvector centrality also have limitations. Most weights of the eigenvector can concentrate in a very few nodes (like hubs), depending on the architecture of the network. In this case, most of the nodes centrality values will be close to zero and, therefore, the importance of nodes is not well quantified. Closeness centrality's limitation is that it is based only on the shortest distances and, therefore, due to the small diameter of networks, the range of variation is too low. Since the typical distance increases with the logarithm of the number of nodes. most complex networks present small average shortest path length.

The performance of a network can be evaluated in many ways. One of them is measuring the network efficiency which considers how efficiently the information exchanges within the nodes in the network [11] This can be applied to local and global ranges in the network. Globally, efficiency means the exchange of information across the whole network where information is exchanged simultaneously. Local efficiency is a small determinant of the network's resilience to failure.

The proposed method in [18] which we are investigating inspects the network efficiency based on information centrality. The information of a node not only contains the node, but also information about the links to it's neighbours. Therefore not all nodes are equivalent, there are more important nodes

which are necessary to be present in the network. Considering this, they remove nodes one-by-one in the presented algorithm, and remeasure the network efficiency after each deletion. The eradication of a node can result in considerable changes in the network efficiency and structure (e.g. example it can alter the connectivity of the network, or the shortest path between two nodes can change), or can happen without any effect on the network.

3.2.2 Efficiency

The efficiency between nodes i and j is defined as the inverse of the shortest path length between the nodes:

$$\varepsilon_{ij} = \frac{1}{d_{ij}} \tag{5}$$

3.2.3 Network efficiency

The efficiency of the whole network $E(G)$ is defined as the average of the sum of node efficiencies and indicates the throughput of information in G [11].

$$E(G) = \frac{\sum_{i \neq j \in G} \varepsilon_{ij}}{N(N-1)} = \frac{1}{N(N-1)} \sum_{i \neq j \in G} \frac{1}{d_{ij}} \tag{6}$$

If there is no link between i and j they take d_{ij} as $+\infty$ and ε_{ij} consistently. $E(G)$ ranges between 0 and 1.

3.2.4 Node efficiency centrality

The efficiency centrality(C_i^{EC}) of a node i is based on the relative drop of the efficiency in the network after the deletion of i from the network G.

$$C_i^{EC} = \frac{\Delta E}{E} = \frac{E(G) - E(G_i')}{E(G)}, i = 1, 2, ..., N \tag{7}$$

here, the stated G_i' subgraph represent the graph after the elimination of node i from the network.

3.3 Graph models

3.3.1 Independent cascade model

Finding the most influential node can be viewed as an *influence maximization problem*. The goal is to find the set of the most important nodes among the

node sets of the same cardinality under a given model representing the spread of the information in the network.

In the information diffusion models, a set of nodes S is chosen from which the information start to propagate. The elements of this set are considered to be *active*. In the *Independent Cascade Model* [9] which we used, in the ith iteration each node that has become active in the previous iteration may activate *(infect)* it's non-active neighbours with a fixed p probability. These neighbours are considered after each another and they either get activated or not. A node may only activate one of it's neighbouring node at most once. The algorithm stops when no node have been activated in the previous or all nodes have became active. The influence of S, $\sigma(S)$ is defined as the number of active nodes after the completion of the algorithm. In the problem, for a given constant k one is to find the set of nodes S with cardinality k to which $\sigma(S)$ is maximal.

3.3.2 Time-varying graph

Time-varying graphs (TVGs) are graphs in which nodes, or edges may vary in time. We adopted the model proposed in [19]. The model represents a TVG as an object $H = (V, E, T)$, where V is the set of nodes, T is the finite set of time instants for which the TVG is defined, and $E \subseteq V \times T \times V \times T$ is the set of edges. As a matter of notation, we denote $V(H)$ as the set of all nodes in H, $E(H)$ the set of all edges in H, and $T(H)$ the set of all time instants in H. An edge $e \in E(H)$ is defined as an ordered quadruple $e = (u, t_a, v, t_b)$, where $u, v \in V(H)$ are the origin and destination nodes (which can be equal), while $t_a, t_b \in T(H)$ are the origin and destination time instants (which can be equal), respectively. Therefore, $e = (u, t_a, v, t_b)$ should be understood as a directed edge from node u at time t_a to node v at time t_b. We also define a *temporal node* as an ordered pair (u, t_a), where $u \in V(H)$ and $t_a \in T(H)$. The set $VT(H)$ of all temporal nodes in a TVG H is given by the cartesian product of the set of nodes and the set of time instants, i.e. $VT(H) = V(H) \times T(H)$. As a matter of notation, a temporal node is represented by the ordered pair that defines it, e.g. (u, t_a).

The usage of the object $H = (V, E, T)$ to represent a TVG is formally introduced in [19]. We however define a new method to generate the changes between the time instances. We use the degree centrality value of a node to determine whether if the node will be deleted at the given time instant t_i or not. To keep the scale of the network, as many node is added as was deleted. We use the same to generate edges between newly added nodes and already

existing nodes. We chose this method based on real life experiences, where the infectious entity with most connections is seperated first.

3.3.3 Time-constrained coverage (TCC)

The *Time-Constrained Coverage (TCC)* is a metric introduced in [4]. It measures the coverage achieved by a diffusion process after a defined number of iterations. Specifically, for a diffusion starting at time t_i, $\mathsf{TCC}(t_i, \phi)$ calculates the average fraction of the nodes reached by the diffusion in ϕ iterations. More formally,

$$\mathsf{TCC}(t_i, \phi) = \frac{1}{|V(H)|^2} \sum_{u \in V(H)} d_c(t_i, u, \phi) \tag{8}$$

where $d_c(t_i, u, \phi)$ is the number of nodes the diffusion reached after the previously defined ϕ iterations, where the diffusion process starts at t_i time instant from u node.

4 Experiments and results

4.1 Data

In this section, we employ four real networks to investigate the the effectiveness of the method proposed in [18] in TVGs with Independent Cascade Model. We chose the networks to be different in the scale of their size. They are Infect Dublin network consisting of 410 nodes and 3K links between them, Wiki-Vote network, which consist of 889 nodes and 3K links, Hamsterster network with 2K nodes and 17K link and Facebook network made of 4K nodes and 88K links, respectively. The data for Infect-Dublin, Wiki-Vote and Hamsterster can be acquired from [16]. The Facebook data can be obtained from [12].

4.2 Experimental analysis

Three centrality measures DC, CC and EVC are chosen to measure the centrality values of each node based on the nature of the empirical networks. The experimental analyses are divided into the following three parts, which are demonstrated below.

4.2.1 Experiment 1: comparing the five most influential nodes between the existing centrality measures and the EC method

First of all, we calculate the raw data in the four networks, with different centrality measures. Then we focus on the top-10 nodes sorted by these measures. The investigated EC [18] will be compared with DC, CC, EVC, and the results are shown in Table 1.

Infect Dublin								
Rank	**DC**		**CC**		**EVC**		**EC**	
1	157	*(0.12225)*	274	*(0.41565)*	286	*(0.20699)*	274	*(0.03194)*
2	304	*(0.11491)*	157	*(0.40137)*	291	*(0.20190)*	304	*(0.02897)*
3	148	*(0.10513)*	243	*(0.37870)*	116	*(0.19297)*	157	*(0.02741)*
4	372	*(0.10513)*	333	*(0.37489)*	410	*(0.19213)*	243	*(0.01880)*
5	211	*(0.08313)*	1	*(0.36913)*	282	*(0.18614)*	148	*(0.01417)*
Wiki-Vote								
Rank	**DC**		**CC**		**EVC**		**EC**	
1	431	*(0.11487)*	273	*(0.39964)*	273	*(0.28524)*	431	*(0.04123)*
2	273	*(0.10360)*	431	*(0.37884)*	431	*(0.27910)*	273	*(0.03242)*
3	170	*(0.07432)*	204	*(0.36862)*	536	*(0.22125)*	170	*(0.02310)*
4	536	*(0.06757)*	536	*(0.35720)*	399	*(0.21352)*	8	*(0.01413)*
5	399	*(0.06306)*	550	*(0.35210)*	416	*(0.21046)*	550	*(0.01296)*
Hamsterster								
Rank	**DC**		**CC**		**EVC**		**EC**	
1	73	*(0.11258)*	73	*(0.34904)*	73	*(0.21148)*	2311	*(0.01754)*
2	121	*(0.09196)*	69	*(0.34553)*	121	*(0.18589)*	73	*(0.01103)*
3	301	*(0.07134)*	121	*(0.33698)*	202	*(0.14365)*	69	*(0.00913)*
4	202	*(0.06351)*	622	*(0.32741)*	617	*(0.12366)*	6	*(0.00905)*
5	6	*(0.06227)*	617	*(0.32643)*	242	*(0.12047)*	416	*(0.00793)*
Facebook								
Rank	**DC**		**CC**		**EVC**		**EC**	
1	107	*(0.25879)*	107	*(0.45970)*	1912	*(0.09541)*	107	*(0.11585)*
2	1684	*(0.19614)*	58	*(0.39740)*	2266	*(0.08698)*	1684	*(0.09310)*
3	1912	*(0.18697)*	428	*(0.39484)*	2206	*(0.08605)*	698	*(0.05891)*
4	3437	*(0.13546)*	563	*(0.39391)*	2233	*(0.08517)*	0	*(0.05202)*
5	0	*(0.08593)*	1684	*(0.39361)*	2464	*(0.08428)*	3437	*(0.04751)*

Table 1: Top-five nodes by DC value, CC value, EVC, value and EC value.

According to Table 1, comparing the investigated EC with CC and DC there 3 same nodes in the top-5 list and none with EVC in the Infect Dublin network. In Wiki-Vote, these numbers are 3 in the case of DC and CC and 2 in the case of EVC. In Hamsterster the number of the same nodes between EC and DC, CC, EVC are respectively one, two and two. In Facebook there are four, two same nodes between EC and DC, CC and none between EC and EVC. Almost in every case, at least one of the top-2 nodes are same between EC and other measures. From this a conclusion can be drawn that the EC has a good performance in all the investigated networks.

4.2.2 Experiment 2: comparing the TCC with different iteration limits

To get a more detailed picture of coverage, we also study the *Time-Constrained Coverage (TCC)* metric. We launch the *Independent Cascade Model* diffusion process from every node, and measure the reached coverage in each network after 25, 50 and 75 iterations. The results are shown in Fig. 1.

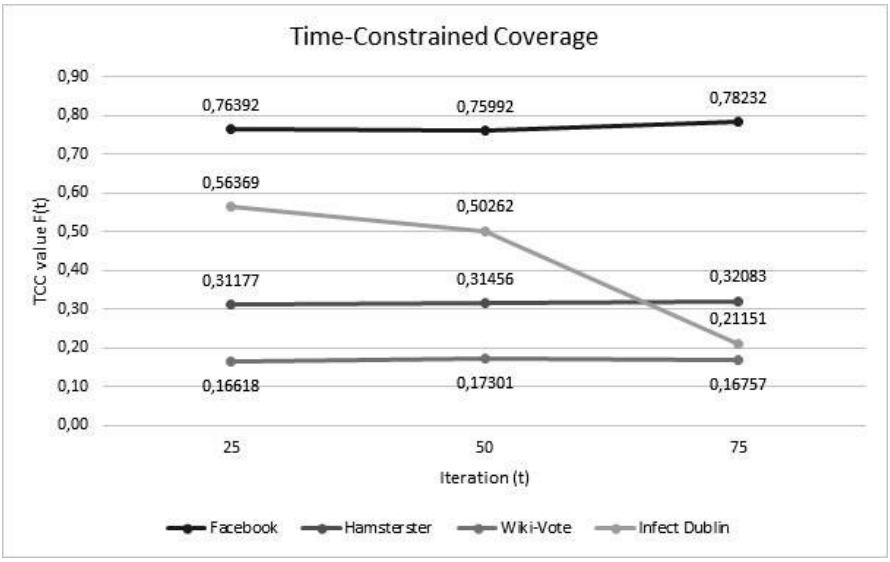


Figure 1: The TCC values in Facebook, Hamsterster, Wiki-Vote and Infect Dublin network after 25, 50 and 75 iterations.

As expected networks with higher node-edge rate has a better TCC value. This is because the more neighbour a node has, the more node it can infect in

the ith iteration of the diffusion process. In the case of Infect Dublin network, a large loss can be seen in the number of the infected nodes. This is happens because of the low number of nodes in the network, since the more iteration the diffusion process takes, the bigger the chance is for the TVG to delete already reached nodes.

4.2.3 Experiment 3: comparing the cover rates and iteration numbers

In order to examine the cover rates we launch the above mentioned *Independent Cascade Model* diffusion from the top two most influential node based on DC, CC, EVC, EC values, and inspect if the propagation can or cannot reach the given portion of the nodes, while the network dynamically changes in each iteration. These fractions are 40%, 75%, and 90%. The experiment either stops if the diffusion process stops, or if 100 iterations are reached. We compare the number of the successful (the given fraction is reached) experiments. The results are obtained by averaging over 1000 implementations. The results are shown in Figs. 2–5.

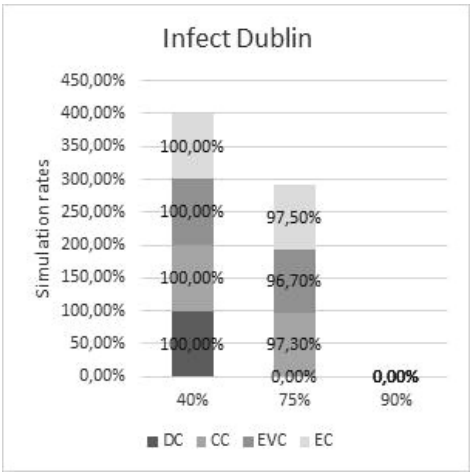


Figure 2: Infect Dublin network cover rates

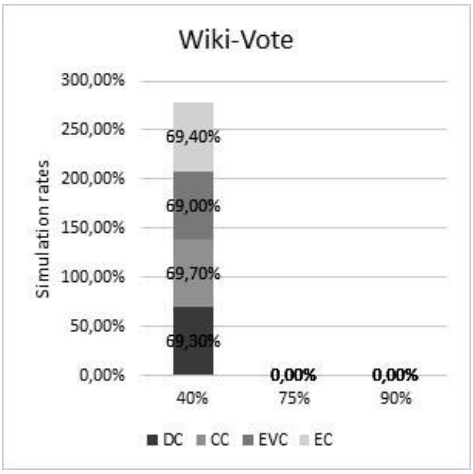


Figure 3: Wiki-Vote network cover rates

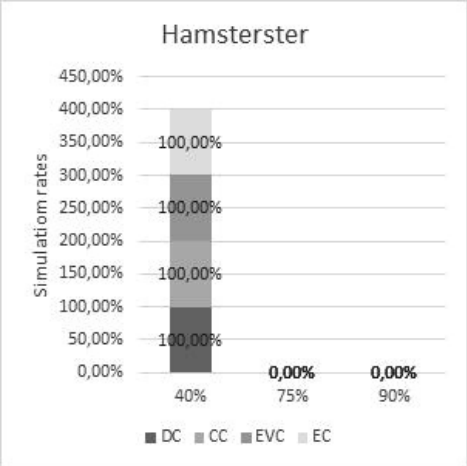


Figure 4: Hamsterster network cover rates

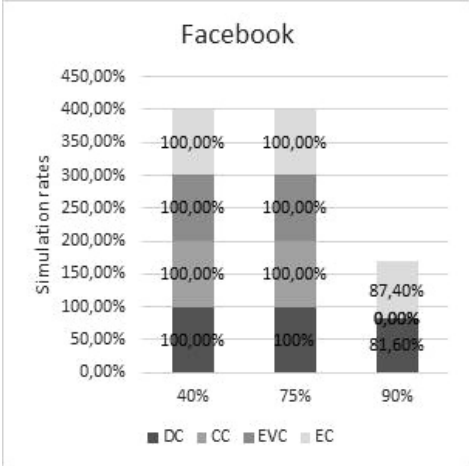


Figure 5: Facebook network cover rates

The effectiveness of the investigated method can be measured with the reached fractions of the nodes. In Infect Dublin network, all centrality measures used in the experiment reaches 40% of the nodes, however in the case of 75% only DC, CC and the investigated EC could. In addition, none of the inspected measures could reach 90% of the nodes. In Wiki-Vote and Hamsterster network, only 40% coverage is reached by all the measures, none of them could reach 75% or 90%. This can be explained with the low node-edge rate in both networks. The diffusion process stops more easily because there is less link through which information could spread. In Facebook because of the high rate of links every measure is capable of reaching 40% and 75% in every simulation. In addition EC and DC could reach 90% as well. EC is and effective measure, since in every case it has the highest rates above DC which comes on second place. It also can be seen that if we start the diffusion process from the top-K nodes, better coverage is achieved than the TCC value of the network.

4.2.4 Experiment 4: comparing the average infection capacity of top 1 nodes

In order to identify the influence of the node *Independent Cascade Model* is used to measure the propagation ability of the node. We start the diffusion from the top one most influential node based on DC, CC, EVC, EC values and inspect the rate of the active and not-active nodes in each round, while the

network dynamically changes in each iteration. We stop the diffusion process, if 50 iteration is reached. The results are obtained by averaging over 1000 implementations. The results are shown in Figs. 6–9.

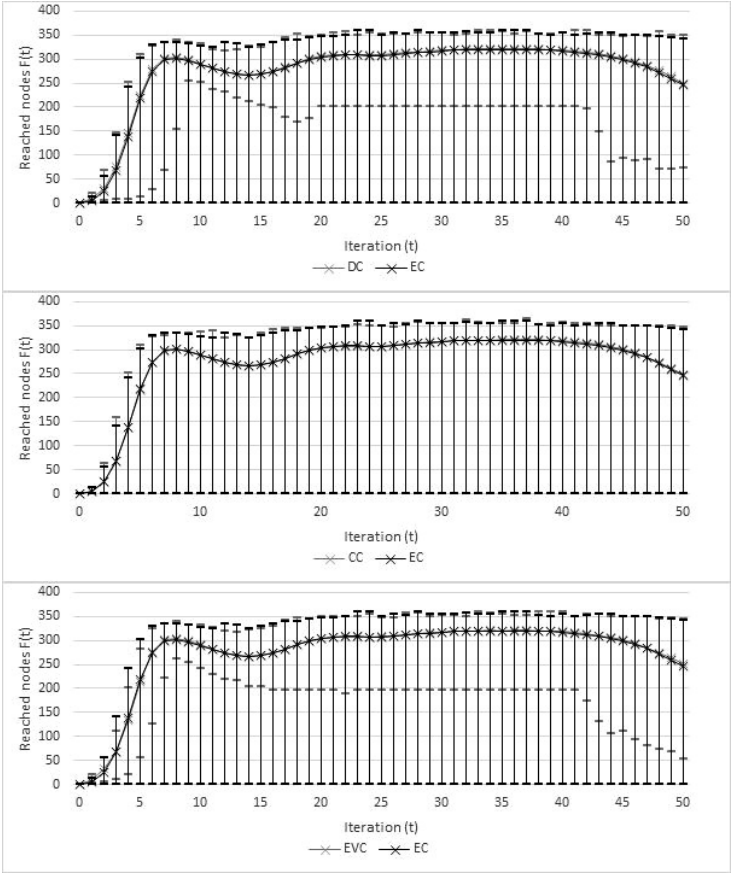


Figure 6: The cumulative number of infected nodes as a function of time with 50 iterations in Infect Dublin network by the investigated EC and different centrality measures.

In this experiment there could be sequences where the number of infected nodes decrease. This is explained with the property of the *Independent Cascade Model* that the diffusion does not stop until there is no other reachable node or all nodes are reached, however because of the dynamic changes of the network, these expectations are almost never met. With the deletion of already infected nodes, and with the addition of new nodes and new links, the number of the infected nodes can decrease as well.

In Infect Dublin network this phenomenon can be seen between iterations 10 and 20, or between iterations 40 and 50. In this experiment EC and CC provided very low minimum infected nodes in each round numbers. This is also explained with the property of TVG. If each node that got infected in the ith iteration are deleted at the end of the iteration, then the diffusion process stops. Because of this, an experiment can stop in the first iteration, resulting in a low minimum number. We can infer that the centrality measure with the higher final number of infected nodes is more effective. This means EC is slightly outperformed by the other measures.

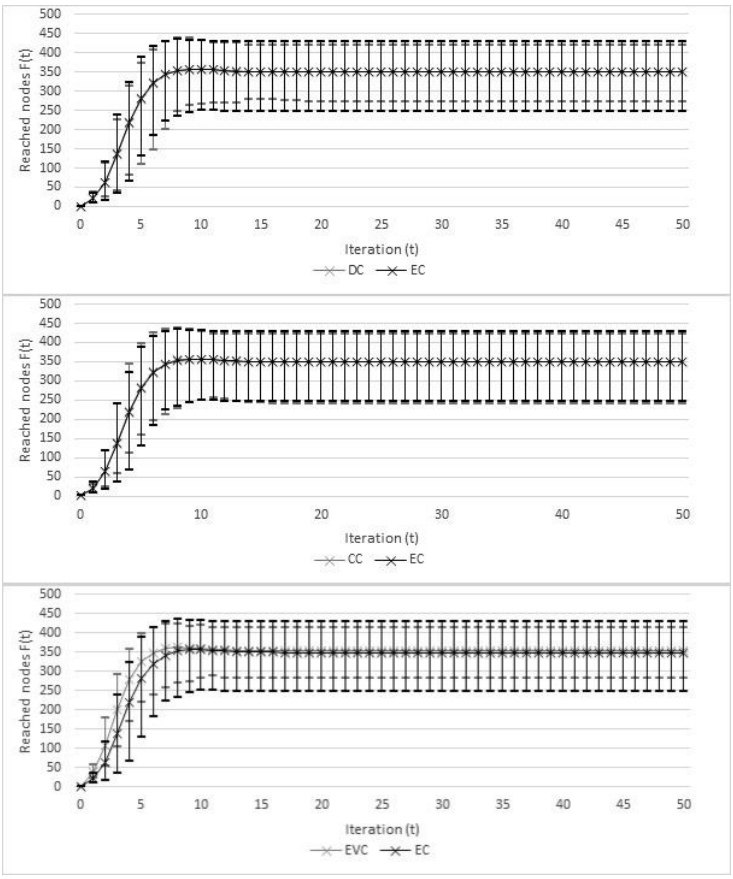


Figure 7: The cumulative number of infected nodes as a function of time with 50 iterations in Wiki-Vote network by the investigated EC and different centrality measures.

In Wiki-Vote network the investigated EC is slightly outperformed by EVC, and it's performance are almost identical with DC and CC. In addition, the number of the infected nodes stabilizes after the 10th iteration. This can be explained with the property of the *Independent Cascade Model*, that there is no other reachable node and the diffusion stops.

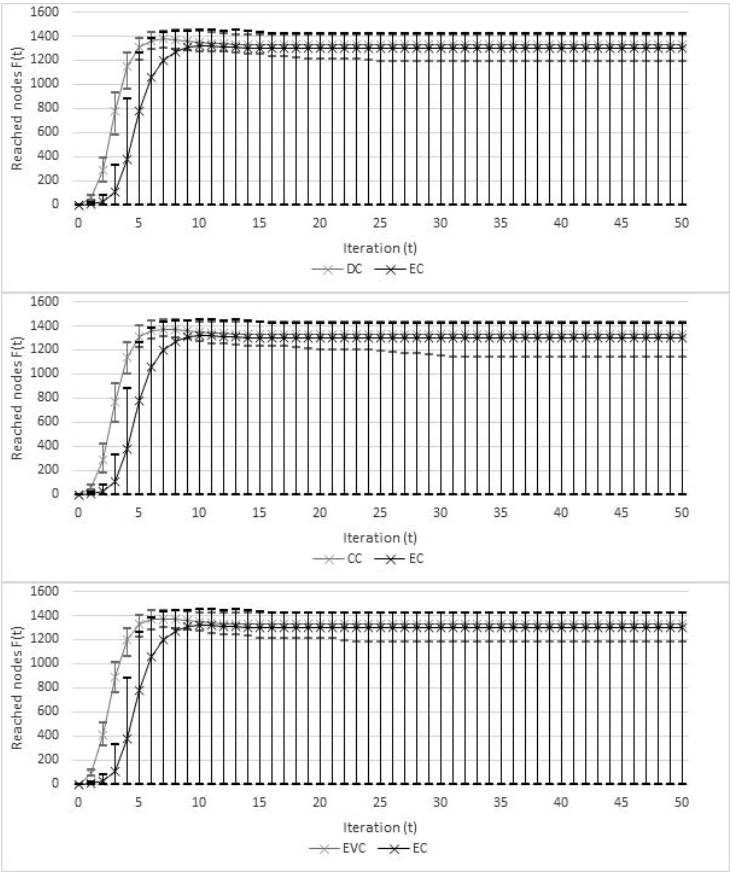


Figure 8: The cumulative number of infected nodes as a function of time with 50 iterations in Hamsterster network by the investigated EC and different centrality measures.

In Hamsterster network, the above mentioned phenomenon with the low minimum numbers can be seen. In addition EC is significantly outperformed by every other centrality measures at the first 10 iterations of the diffusion process and remains slightly outperformed after the stabilization of the number of the infected nodes.

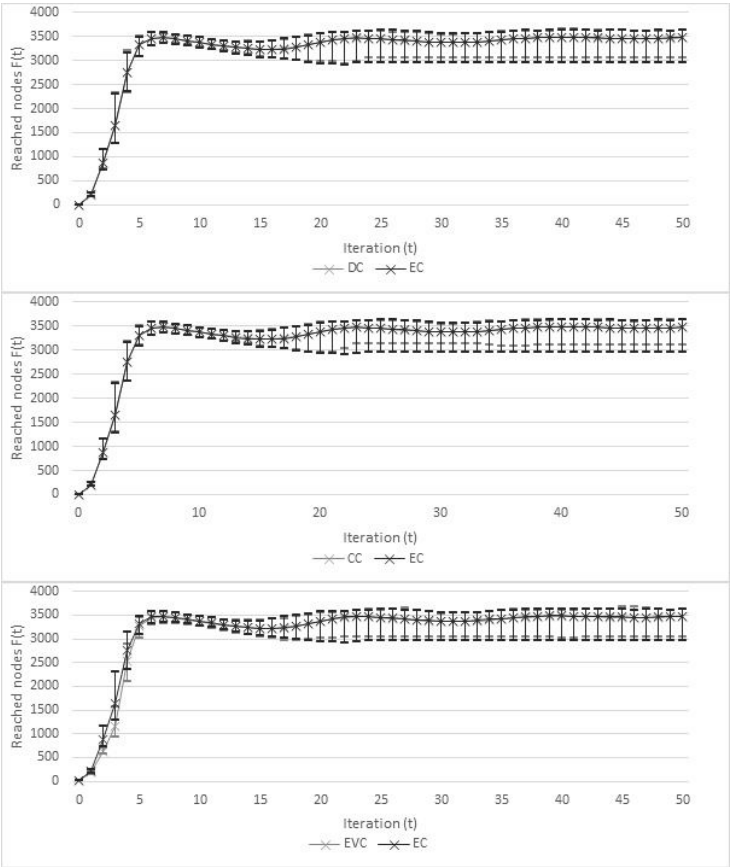


Figure 9: The cumulative number of infected nodes as a function of time with 50 iterations in Facebook network by the investigated EC and different centrality measures.

In Facebook network the investigated EC outperforms DC and EVC, and it's results are almost identical with CC's. The infected number only stabilizes after 40 iterations in the case of every measure. This can be explained with the high rate of the nodes and edges in the network, since the average number a node comes in connection with is greater than in any of the networks used in the experiments.

5 Conclusion and future work

In this paper we investigated the effectiveness of a recently proposed centrality measure, called Efficiency Centrality in the case of Time-Varying Graphs. The algorithm ranks the nodes based on their necessity to exist in the network. To measure this, they calculate the efficiency of the network before and after the removal of the node. We investigated this method in the case of TVGs to get a more detailed picture. These graphs change over time by the deletion and addition of nodes. To evaluate the performance in such graphs, we apply the investigated method on four real networks and use Independent Cascade model as the diffusion process. The experimental results show that Efficiency Centrality has a better performance, in case of larger changing networks with a high number of edges, however slightly falls back behind other measures when smaller networks with few links are used.

While we only investigated the algorithm in a specific model, it is possible that the algorithm could achieve better results in different models. It also could be compared with different centrality measures, which we plan to do in the future.

Acknowledgements

The project has been supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.3-VEKOP-16-2017-00002).

References

[1] A. Bavelas, Communication in task-oriented groups, *The Journal of the Acoustical Society of America*, **22,** 6 (1950) 725–730. ⇒7

[2] P. Bonacich, Power and centrality: A family of measure, *American journal of sociology*, **92,** 5 (1987) 1170–1182. ⇒6

[3] S. P. Borgatti, Centrality and network flow, *Social networks*, **27,** 1 (2005) 55–71. ⇒6

[4] E. C. Costa, A. B. Vieira, K. Wehmuth, A.r Ziviani, A. P. Couto Da Silva, Time centrality in dynamic complex networks, *Advances in Complex Systems*, **18,** 07n08 (2015), p. 1550023. ⇒6, 11

[5] L. C. Freeman, A set of measures of centrality based on betweenness, *Sociometry*, **40,** 1 (1977) 35–41. ⇒6

[6] L. C. Freeman, Centrality in social networks conceptual clarification, *Social networks*, **1,** 3 (1978) 215–239. ⇒6

[7] N. E. Friedkin, Theoretical foundations for centrality measures, *Americal journal of Sociology*, **96,** 6 (1991) 1478–1504. ⇒6

[8] S. Gao, J. Ma, Z. Chen, G. Wang, C. Xing, Ranking the spreading ability of nodes in complex networks based on local structure, *Physica A: Statistical Mechanics and its Applications*, **403,** (2014) 130–147. ⇒6

[9] D. Kempe, J. Kleinberg, É. Tardos, Maximizing the spread of influence through a social network, *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, (2003) 137–146. ⇒10

[10] B. Kósa, M. Balassi, P. Englert, G Rácz, Z Pusztai, A. Kiss, A basic network analytic package for rapidminer, *Rapid Miner World*, (2014) ⇒6

[11] V. Latora, M. Marchiori, Efficent behavior of small-world networks, *Physical review letters*, **87,** 19, (2001), p. 198701. ⇒8, 9

[12] J. Leskovec, A. Krevl, SNAP Datasets: Stanford large network dataset collection, (2014), http://snap.stanford.edu/data ⇒11

[13] M. Nekovee, Y. Moreno, G. Bianconi, M. Marsili, Theory of rumour spreading in complex social networks, *Physica A: Statistical Mechanics and its Applications*, **374,** 1 (2007) 457–470. ⇒6

[14] M. EJ. Newman, Mathematics of networks, *The new Palgrave dictionary of economics*, (2016) 1–8. ⇒7

[15] L. Page, S. Brin, R. Motwani, T. Winograd, The pagerank citation ranking: Bringing order to the web, *Technical report, Standord InfoLab*, (1999) ⇒6

[16] R. A. Rossi, N- K. Ahmed, The network data repository with interactive graph analytics and visualization, (2015), http://networkrepository.com ⇒11

[17] Ch. Song, S. Havlin, H. A. Makse, Self-similarity of complex networks, *Nature*, **443,** 7024, (2005), p. 392. ⇒6

[18] S. Wang, Y. Du, Y. Deng, A new measure of identifying influential nodes: Efficiency centrality, *Communications in Nonlinear Science and Numerical Simulation*, **47,** (2017) 151–163. ⇒6, 8, 11, 12

[19] K. Wehmuth, A. Ziviani, E. Fleury, A unifying model for representing time-varying graphs, *2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, IEEE, (2015) 1–10. ⇒10

[20] A. Zaki, M. Attia, D. Hegazy, S. Amin, Comprehensive survey on dynamic graph models, *International Journal of Advanced Computer Science and Applications*, **7,** 2 (2016) 573–582. ⇒6

Received: January 19, 2020 • Revised: February 18, 2020

Acta Univ. Sapientiae, Informatica **12,** 1 (2020) 22–38

DOI: 10.2478/ausi-2020-0002

Formal concept analysis for amino acids classification and visualization

Adrian-Sorin TELCIAN
Babes-Bolyai University
Cluj-Napoca, Romania
email: adriant@cs.ubbcluj.ro

Daniela-Maria CRISTEA
Babes-Bolyai University
Cluj-Napoca, Romania
email: danielacristea@cs.ubbcluj.ro

Ioan SIMA
Babes-Bolyai University
Cluj-Napoca, Romania
email: sima.ioan@cs.ubbcluj.ro

Abstract. Formal concept analysis (FCA) is a method based on lattice theory, widely used for data visualization, data analysis and knowledge discovery. Amino acids (AAs) are chemical molecules that constitute the proteins. In this paper is presented a new and easy way of visualizing of the structure and properties of AAs. In addition, we performed a new Hydrophobic-Polar classification of AAs using FCA. For this, the 20 proteinogenic AAs were clustered, classified by hydrophobicity and visualized in Hasse-diagrams. Exploring and processing the dataset was done with Elba and ToscanaJ, some FCA tools and Conceptual Information System (CIS).

Formal concept analysis (FCA) is a method based on lattice theory and is used for data analysis, knowledge representation, information retrieval and knowledge discovery [12]. FCA is a semantic technology that targets a formalization of concepts for human understanding [3].

Computing Classification System 1998: F.4.1
Mathematics Subject Classification 2010: 03G10
Key words and phrases: formal concepts, lattices, proteins, amino acids, scales, organic compounds

FCA offers efficient algorithms for data analysis and data detection of hidden dependencies. It also makes it easy for the user to visualize the information.

Over the past decades, biological information has risen exponentially. The analysis and interpretation of these data remain a challenge for researchers [12].

The amino acids (AA) are monomers that form the proteins. From the hundreds of amino acids found in living organisms, only 20 AAs take part in the protein buildings. These are called proteinogenic AAs.

Biological data analysis is usually quantitative and based on mathematical statistics. A qualitative method based on Formal Concept Analysis (FCA) is used in this document. As far as we know, FCA has not been applied for the AAs study.

Using Elba and ToscanaJ, some known FCA tools, the 20 amino acid characteristics and hydrophobicity are evaluated and analyzed. FCA uses crosstables (a representation of the "formal contexts"), where rows represent amino acids, columns represent attributes of amino acids, and cells contain information about attribute values (i.e., number of atoms of molecules and other properties). The AAs are clustered into meaningful sets. The clusters, which form a hierarchy, are visually displayed in the Hasse diagrams [3].

FCA has been used for the following:

1. the automated classification of enzymes [2]; the authors used supervised and unsupervised classification, obtaining a correct classification for more than 50% percent from analyzed sequences.
2. knowledge identification, knowledge acquisition, knowledge development, knowledge distribution and sharing, knowledge usage, and knowledge sustainability concepts in the economic field. [17];
3. class hierarchy design in object-oriented programming (OOP) [6];
4. analysis, conception, implementation and validation of class (or object) hierarchies and component retrieval in the field of software engineering (SE) [7];
5. business intelligence (BI) as framework technologies that meaningfully reduce space of OLAP cube on a hierarchy of attributes [10];
6. membership constraints, a problem of consistency to determine if a formal concept exists whose object and attribute set include certain elements and exclude others; [15];
7. modeling and querying with a conceptual graph of data from RDBMS and XML databases. [11].

1 Formal concept analysis

Formal Concept Analysis (FCA) is a field of Applied Mathematics based on formalizing concepts and conceptual hierarchies from a lattice-theoretical perspective. FCA offers algorithms for data analysis and detection of hidden dependencies from sets of data and does this in a highly efficient manner. Representing data in FCA is done in the form of formal contexts, which is the easiest way of specifying what attributes are valid for which objects. Doing so, it makes data visualization an easily understandable way.

Some definitions are needed to understand the FCA.

Formal context - A *formal context* is a triplet (X, Y, I) where X and Y are non-empty sets, and I is a binary relation between X and Y, i.e., $I \subseteq X \times Y$.

In a formal context, items $x \in X$ are called objects and items $y \in Y$ are called attributes. $(x, y) \in I$ shows that object x has a y attribute.

Formal context is represented as a cross-table with n rows and m columns. The corresponding formal context consists of a set $X = \{x_1, x_2, ..., x_n\}$, a set $Y = \{y_1, y_2, ..., y_m\}$, and a binary relation I defined by: $(x_i, y_j) \in I$ if the table entry corresponding to row i and column j contains "$\times$" (see Fig. 1).

I	y_1	y_2	y_3	y_4
x_1	$\times$	$\times$	$\times$	$\times$
x_2	$\times$		$\times$	$\times$
x_3		$\times$	$\times$	$\times$
x_4		$\times$	$\times$	$\times$
x_5	$\times$			

Figure 1: The cross-table corresponding to the formal context

Components of X are known as objects and refer to table rows, components of Y are known as attributes and refer to table columns, and for $x \in X$ and $y \in Y$, $(x, y) \in I$ suggests that object x has an attribute y, while $(x, y) \notin I$ implies that x does not have attribute y. For example, Fig. 1 displays a cross-table (aka logical attribute table) corresponding to triplet (X, Y, I), given by $X = \{x_1, x_2, x_3, x_4, x_5\}$, $Y = \{y_1, y_2, y_3, y_4\}$ and $I = \{(x_1, y_1), (x_1, y_2), ..., (x_2, y_1), \ldots, (x_5, y_1)\}$. Notice that $(x_1, y_1) \in I$, whereas $(x_2, y_2) \notin I$, etc.

In a formal context, always we have a pair of operators, called concept-forming operators.

Concept-forming operators - For a formal context (X, Y, I), operators $\uparrow: 2^X \to 2^Y$ and $\downarrow: 2^Y \to 2^X$ are defined for every $A \subseteq X$ and $B \subseteq Y$ by

$A^{\uparrow} = \{y \in Y \mid \text{for each } x \in A : (x, y) \in I\}$,

$B^{\downarrow} = \{x \in X \mid \text{for each } y \in B : (x, y) \in I\}$.

In FCA, the notion of a formal concept is essential, they are specific clusters in cross-tables defined through attribute sharing.

Formal concept - A formal concept in the formal context (X, Y, I) is a pair (A, B) of $A \subseteq X$ and $B \subseteq Y$ such that $A^{\uparrow} = B$ and $B^{\downarrow} = A$.

For a formal concept (A, B) in the formal context (X, Y, I), A and B are called the extent and intent of (A, B), respectively. Formal concepts can be described as: (A, B) is a formal concept iff A includes only objects sharing all attributes from B and B contains only attributes shared by all objects from A.

The formal concept as term can be seen as a mathematization of a well-known idea that evokes Port-Royal logic. In that logic, a concept is determined by a collection of objects (extent) that fall under the concept and a collection of attributes (intent) covered by the concepts. In the cross-table from Fig. 1 we can see some formal concepts. Thus, $(\{x_1\}, \{y_1, y_2, y_3, y_4\})$ is a formal concept with extent $\{x_1\}$ and intent $\{y_1, y_2, y_3, y_4\}$.

Concepts are usually ordered based on the subconcept-superconcept relation which is based on the inclusion relation defined on objects and attributes.

The subconcept-superconcept relationship is formally defined as follows:

Subconcept-superconcept ordering - For formal concepts (A_1, B_1) and (A_2, B_2) of formal context (X, Y, I), put $(A_1, B_1) \leq (A_2, B_2)$ iff $A_1 \subseteq A_2$ and $B_2 \subseteq B_1$.

A concept lattice, another fundamental notion in FCA, is the collection of all formal concepts of a specified formal context.

Concept lattice - Denote by $\beta(X, Y, I)$ the collection of all formal concepts of formal context (X, Y, I), i.e.

$\beta(X, Y, I) = \left\{(A, B) \in 2^X \times 2^Y \mid A^{\uparrow} = B, B^{\downarrow} = A\right\}$.

$\beta(X, Y, I)$ equipped with the $\leq$ subconcept-superconcept ordering is called a (X, Y, I) concept lattice.

$\beta(X, Y, I)$ is all (potentially interesting) clusters "hidden" in (X, Y, I) information.

According to Main theorem of concept lattices [16], $(\beta(X, Y, I), \leq)$ is a complete lattice.

Denote the extent of concepts:

$\mathrm{Ext}(X, Y, I) = \left\{A \in 2^X \mid (A, B) \in \beta(X, Y, I) \text{ for some } B\right\}$

and intents of concepts:

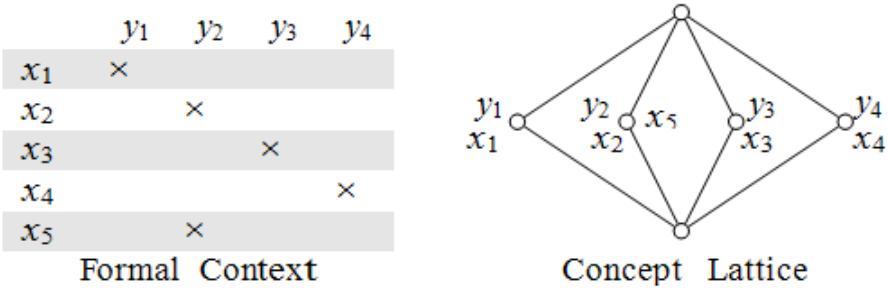


Figure 2: The concept lattice corresponding to the cross-table (formal context)

$\mathrm{Int}(X, Y, I) = \{B \in 2^y \mid (A, B) \in \beta(X, Y, I) \text{ for some } A\}$.

In Fig. 1 is showed the concept lattice for a formal context represented by the cross-table. On the lattice the next formal concepts are extracted: $(\{x_1\}, \{y_1\})$; $(\{x_2, x_5\}, \{y_2\})$; $(\{x_3\}, \{y_3\})$ and $(\{x_4\}, \{y_4\})$.

Conceptual Scaling

Especially at the data collected from practical applications, the attributes no longer have binary values, of the type *yes / no*. These situations are formalized in the so-called *many-valued context*, in which a particular object has a certain attribute with a certain value.

Many-valued context - A *many-valued context* (G, M, W, I) consists of sets G, M and W and a ternary relation I between G, M and W (i.e., $I \subseteq G \times M \times W$) for which it holds that $(g, m, w) \in I$ and $(g, m, v) \in I$ always implies $w = v$.

The elements of G are called objects, those of M (many-valued) attributes and those of W attribute values. $(g, m, w) \in I$ is read as "the attribute m has the value w for the object g".

The many-valued attributes can be regarded as partial maps from G in W. Therefore, it seems reasonable to write $m(g) = w$ instead of $(g, m, w) \in I$.

The *many-valued context* is transformed into a *one-valued* one, trough process called *conceptual scaling* that is not at all uniquely determined.

In the process of scaling, first of all each attribute of a *many-valued context* is interpreted by means of a context. This context is called *conceptual scale*.

Conceptual scale - A *conceptual scale* is a *scale* for the attribute m of a

	$\leqslant 1$	$\leqslant 2$	$\leqslant 3$	$\leqslant 4$	$\geqslant 1$	$\geqslant 2$	$\geqslant 3$	$\geqslant 4$
1	×	×	×	×	×			
2		×	×	×	×	×		
3			×	×	×	×	×	
4				×	×	×	×	×

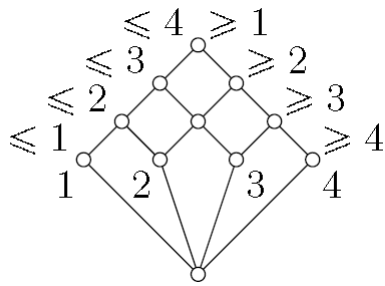


Figure 3: Interordinal scale

many-valued context. Thus, is a *one-valued* context $\mathsf{S}_{\mathfrak{m}} = (\mathsf{G}_{\mathfrak{m}}, \mathsf{M}_{\mathfrak{m}}, \mathrm{I}_{\mathfrak{m}})$ with $\mathfrak{m}(\mathsf{G}) \subseteq \mathsf{G}_{\mathfrak{m}}$. The objects of a scale are called *scale values*, the attributes are called *scale attributes*.

The most commonly used scales also called *elementary scales* are nominal scales, ordinal scales, interordinal scales, biordinal scales and dichotomic scales. An example of interordinal scale is point out in Fig. 3.

ToscanaJ Suite

There is many software for FCA and most of them support the creation of contexts from scratch and the subsequent processing and presentation of the corresponding concept lattices. More than that, Elba and ToscanaJ are a set of mature FCA tools that allow us to query and navigate through data in databases. They are meant to be a *Conceptual Information System (CIS)*.

When implementing a CIS using methods of FCA, the data is modeled mathematically by a *many-valued context* and is transformed via *conceptual scaling* [5]. This means that is defined a formal context called *conceptual scale* for each of the many-valued attributes which has the values of the attribute as objects. Here, a CIS is an FCA-based system used to analyze data from one table.

Creating the *conceptual scale* is realized with a CIS editor (Elba) and usually is a highly iterative task. In the run-time stage, a CIS browser (ToscanaJ) allows us to explore and analyze the real data from the database with the CIS schema.

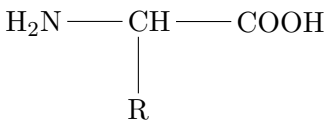
2 Amino acids – structure and properties

The monomers from which the proteins are created are called amino acids (AAs). The proteins are a large family of organic macromolecular compounds.

Only 20 AAs of the hundreds of AAs (which are found in the living organisms) take part in the protein assembly.

Chemically, AAs are organic molecules containing two types of antagonistic functional groups: carboxyl ($-\mathsf{COOH}$) having an acid character; amine ($-\mathsf{NH_2}$) having a base character. They also contains a side chain (R group or AA residue) specific to each amino acid [4].

The molecular formula for AA is:



All AAs can be classified according to different properties: hydrophobicity, R type, etc.

Hydrophobicity is the molecule's physical property to be repelled by water molecules. In fact, there is no repulsive force, but only the absence of attraction. In contrast, the hydrophilic (or polar) molecules are attracted to water molecules. Several hydrophobicity scales have been developed over time. Note that AA's hydrophobicity is crucial for understanding the protein folding. [1, 9, 14].

Table 1 presents the structural and functional aspects of AAs. All 20 AAs are composed of carbon (C), hydrogen (H), nitrogen (N) and oxygen (O) atoms. Moreover, Cysteine and Methionine contain Sulphur (S) in addition to the other AAs. The first column shows the 3-letter name for AAs (biochemistry nomenclature). From two to six column there are carbon, hydrogen, nitrogen, oxygen and sulphur number of atoms, respectively. The last column returns the hydropathy index of AA residues. [9]. Hydropathy index is a measure of relative hydrophobicity. A higher hydropathic index value means more hydrophobic amino acid.

In literature, there are four known hydrophobicity scales taken from [20] (shown in Fig. 4). The most hydrophobic AA residues are at the top of the figure. References for the scales are: (1) Kyte and Doolittle [9]; (2) Rose, et al [13]; (3) Wolfenden, et al.[18]; and (4) Janin [8].

3 Experiments and results

FCA is suitable for binary attributes. In the real data type issue, the situation is slightly different. In our case, each attribute is assigned integer (numbers of atoms) or real values (hydropathic index), not binary values. A method called

AA	**C**	**H**	**N**	**O**	**S**	**Hydropathy index**
Gly	2	5	1	2	0	-0.4
Ala	3	7	1	2	0	1.8
Val	5	11	1	2	0	4.2
Leu	6	13	1	2	0	3.8
Ile	6	13	1	2	0	4.5
Phe	9	11	1	2	0	2.8
Pro	5	9	1	2	0	-1.6
His	6	9	3	2	0	-3.2
Trp	11	12	2	2	0	-0.9
Ser	3	7	1	3	0	-0.8
Thr	4	9	1	3	0	-0.7
Tyr	9	11	1	3	0	-1.3
Cys	3	7	1	2	1	2.5
Met	5	11	1	2	1	1.9
Asp	4	7	1	4	0	-3.5
Asn	4	8	2	3	0	-3.5
Glu	5	9	1	4	0	-3.5
Gln	5	10	2	3	0	-3.5
Lys	6	14	2	2	0	-3.9
Arg	6	14	4	2	0	-4.5

Table 1: List of proteinogenic amino acids

conceptual scaling is used for the FCA application to these data. Conceptual scaling converts the many-valued context into a standard formal context.

An advantage of FCA is that there is no standard attributes interpretation. The field expert chooses a suitable scale for attributes interpretation [3]. In our approach, for another ways of visualization of properties and classification of AAs, we used data relating to the molecular structure of the 20 AA, i.e. atoms number of the molecule. For data representation we used Elba and ToscanaJ tools.

Visualization 1. For transformation from many-valued context to one-valued context interordinal scale is used. In Figs. 5–8 can be visualized a clustering of AAs by the number of atoms from molecules.

In Fig. 5 we can see that have been retrieved 7 formal concepts. Thus, 5% from those 20 AAs (meaning 1 AA) have 11 carbon atoms, 10% (meaning 2 AAs) contains 9 carbon atoms (≥ 9 and < 10), and so forth.

Kyte and Doolittle (1)	Rose, et al (2)	Wolfenden, et al (3)	Janin (1979) (4)
Ile	Cys	Gly,Leu,Ile	Cys
Val		Val,ala	Ile
	Phe,Ile		Val
Leu	Val	Phe	Leu,Phe
	Leu,Met,Trp	Cys	Met
Phe		Met	Ala,Gly,Trp
Cys			
Met,Ala	His	Thr,Ser	
	Tyr	Trp,Tyr	His,Ser
Gly	Ala		Thr
Thr,Ser	Gly		Pro
Trp,Tyr	Thr		Tyr
Pro			Asn
		Asp,Lys,Gln	Asp
His	Ser	Glu,His	Gln,Glu
Asn,Gln	Pro,Arg	Asp	
Asp,Glu	Asn		
Lys	Gln,Asp,Glu		
			Arg
Arg	Lys	Arg	Lys

Figure 4: A comparison of four distinct scales for the hydrophobicity

In Fig. 7 notice 3 formal concepts: 65% monocarboxylic AAs (2 oxygen atoms), 10% AAs with 4 oxygen atoms, and 25% AAs have 3 oxygen atoms.

Similar information can also be extracted from Figs. 6 and 8. This is a faster way to visualize complex information than in the tables.

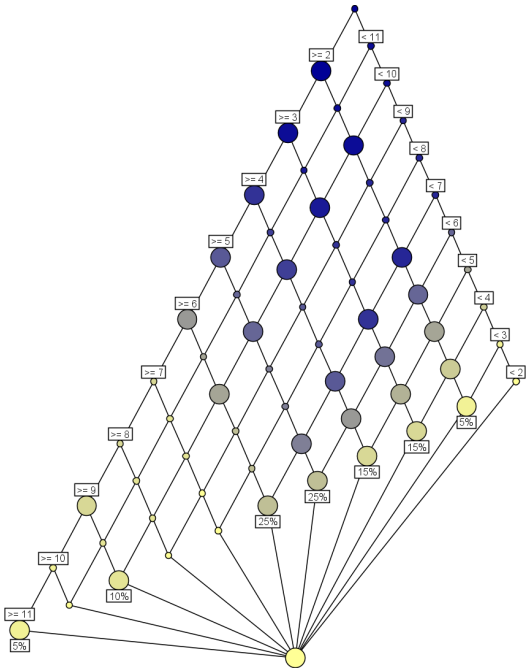


Figure 5: Diagram for number of Carbon atoms

Visualization 2. Subsequent, we used two attributes: the number of carbon atoms and the number of oxygen atoms from AA molecules and created a new scale based on these. The scaling was done as follows. AAs that have more than 3 atoms of the oxygen we called *AA Dicarboxylic*. The according to the number of carbon atoms distinguish: 1). *AAs Low* - AAs with less than 4 carbon atoms; 2). *AAs High* - AAs with more than 5 carbon atoms; and 3). *AAs Medium* - the others;

In Fig. 9 the concept lattice shows number of AAs belong to the above-mentioned levels. For instance, we can read from the lattice that there are two AA Dicarboxylic, and these belong *AA Medium* group. This is one of the main distinguishing characteristics of using concept lattices to visualize information.

Visualization 3. A strong method of FCA is to "mix" many lattices together to provide a combined perspective of several lattices, called a nested line diagram. Fig. 10 displays a mix of diagrams from Fig. 11 and Fig. 9.

Hydrophobic-Polar Classification In addition, for classification, we have used data relating to the hydropathy index (*hi*). AAs classification in hydrophobic (H) and polar (P) is important for some protein folding models.

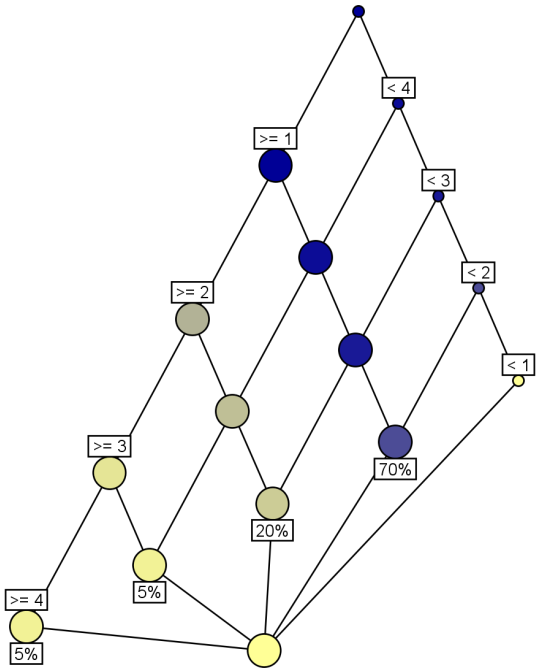


Figure 6: Diagram for number of Nitrogen atoms

Unfortunately, there is no single classification. For example, we have the ones four scale from Fig. 4.

Clustering algorithms most frequently used are: hierarchical, k-means, self-organizing maps, fuzzy k-means [12]. An alternative approach to grouping AAs can be FCA. The lattice concepts of AAs are assumed to show new or old biological relationships.

Relating to hydropathy index (*hi*) we have defined four hydrophobicity levels of AAs: i) if $\mathtt{hi} < -3$: *Polar with electrically charged propensity*; ii) if $\mathtt{hi} < -1.5$: *Polar*; iii) if $\mathtt{hi} < 0$: *Uncertain*; iv) if $\mathtt{hi} \leq 4.5$: *Hydrophobic*. Fig. 11 shows the concept lattice of this scale.

Our classification are presented in Table 2, column 4, compared to the classification taken from [4], page 26 (column 2) and Rosalind [19] (column 3).

Our classification:

Hydrophobic AAs: Ala, Val, Leu, Ile, Phe, Cys, Met.

Polar AAs: Pro, Asp, Glu, Arg, Lys, His, Asn, Gln.

The other five AAs (Gly, Trp, Ser, Thr and Tyr) it remains to be classified according to other criteria.

AA	**Dinu**	**Rosalind**	**FCA**
H	Ala, Val, Leu, Ile, Pro Phe, Trp, Met	Ala, Val, Leu, Ile, Phe Trp, Met, Tyr	Ala, Val, Leu, Ile, Phe Cys, Met
P neutral	Ser, Thr, Tyr, Cys, Asn Gln, Gly	Ser, Thr, Asn, Gln	Pro
P charged	Asp, Glu, Arg, Lys, His	Asp, Glu, Arg, Lys, His	Asp, Glu, Arg, Lys, His, Asn, Gln
Uncertain	-	Gly, Cys, Pro	Gly, Trp, Ser, Thr, Tyr

Table 2: HP AAs classification

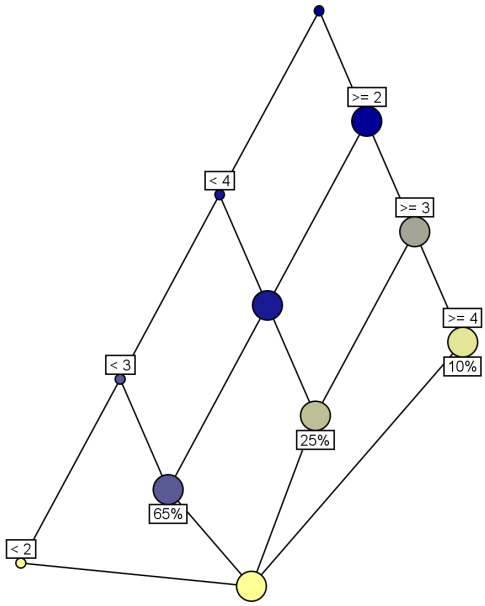


Figure 7: Diagram for number of Oxygen atoms

It can be notice that the classes of AAs identified by applyed FCA, taking into account that hydropathy index are similar to those in the classification taken from Dinu and Rosalind respectively.

Of those seven "Hydrophobic" AAs found, six are found in the same class and in the other two classifications. The seventh AA, Cysteine (Cys), is considered "Polar neutral" AA in Dinu, and in Rosalind it is considered a special case.

Through FCA, we find a single "Polar neutral" AA: Proline (Pro). In Rosalind, this AA is classified in special cases and is considered a hydrophobic AA by Dinu.

In the "Polar charged" class were found the five AAs from the Dinu and Rosalind classifications. Additionally, by our method, we found two new AAs in this class: Asn and Gln. Both AAs are classified as "Polar neutral" AAs in both Dinu and Rosalind.

In contrast, in our classification, there are 5 AAs difficult to classify, which we called "Uncertain AAs". The characteristic of the classification using FCA is that it is sensitive to defining hydrophobicity levels.

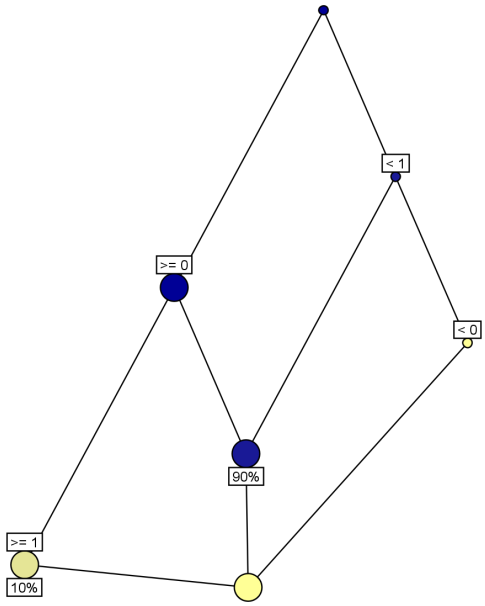


Figure 8: Diagram for number of Sulphur atoms

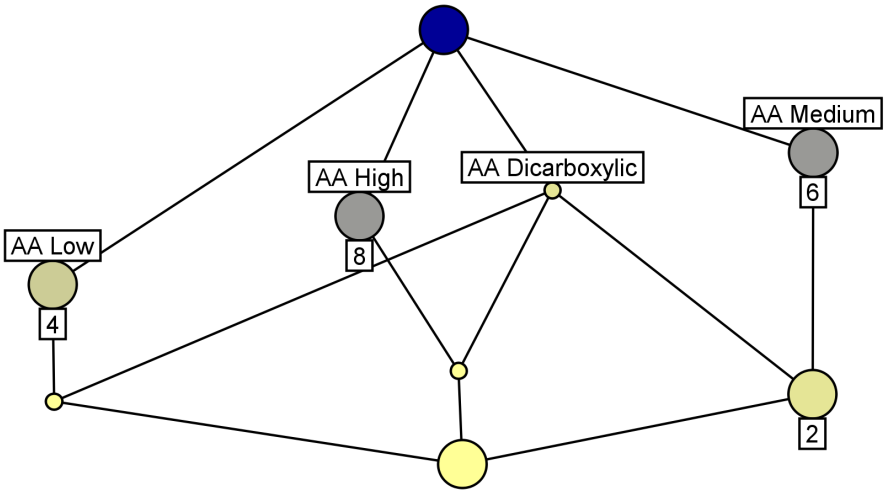


Figure 9: Diagram of the scale based on number of carbon and oxygen atoms from AAs

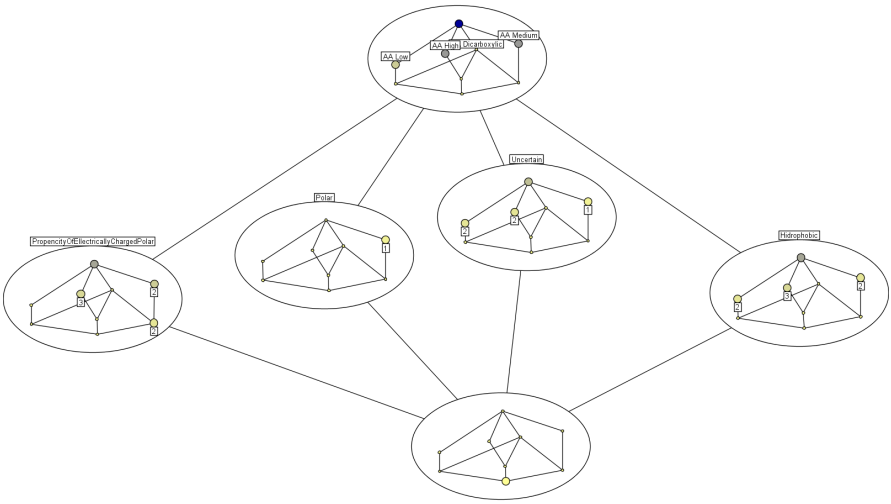


Figure 10: Nested diagram

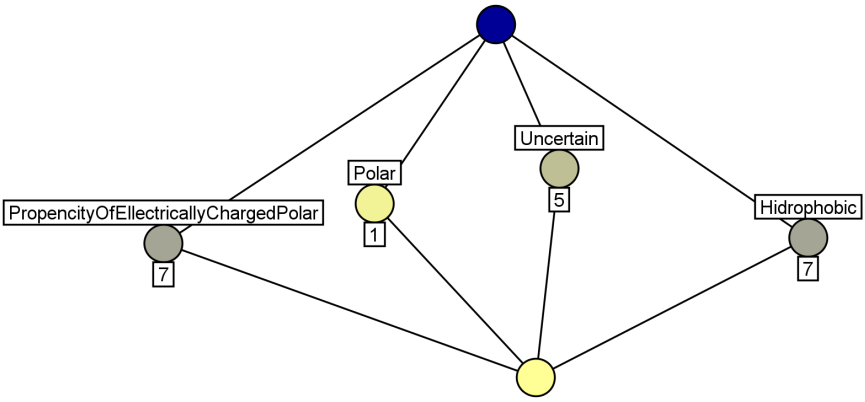


Figure 11: Hydrophobicity scale

Finally, it is difficult to say whether this method adds to the other types of classifications because the hydrophobicity depends on the physical and chemical conditions in which the measurement was taken.

4 Conclusion

In the presented paper, we relied on lattice theory commonly used to analyze and visualize data with formal concept analysis (FCA).

The purpose of this work was to realize a new Hydrophobic-Polar classification and to visualize structure information of AAs in perspective of the Hasse diagrams. Elba and ToscanaJ tools were used to process and explore the dataset.

We defined the hydrophobicity index based on the Kyte and Doolittle scale. In the future, AAs can be classified considering others three known scales of hydrophobicity: Rose, Wolfenden and Janin.

Acknowledgements

The authors would like to thank Prof. Univ. Dr. Bazil Pârv, our research project supervisor, for his professional guidance and valuable support.

References

[1] D. Bandyopadhyay, E. L. Mehler. Quantitative expression of protein heterogeneity: Response of amino acid side chains to their local environment. *Proteins: Structure, Function, and Bioinformatics*, **72,** 2 (2008) 646–659. ⇒28

[2] F. Coste, G. Garet, A. Groisillier, J. Nicolas, T. Tonon, Automated enzyme classification by formal concept analysis, In C. V. Glodeanu, M. Kaytoue, and C. Sacarea, editors, *Formal Concept Analysis*, **8478**, pp. 235–250, Cham, 2014. Springer International Publishing. ⇒23

[3] F. Dau, B. Sertkaya, Formal concept analysis for qualitative data analysis over triple stores, In O. De Troyer, C. Bauzer Medeiros, R. Billen, P. Hallot, A. Simitsis, H. Van Mingroot, editors, *Advances in Conceptual Modeling. Recent Developments and New Directions. ER 2011. Lecture Notes in Computer Science*, **6999**, pp. 45–54, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg. ⇒ 22, 23, 29

[4] V. Dinu, E. Trutia, E. Popa-Cristea, A. Popescu, *Medical Biochemistry – small treated (in Romanian).* Ed. Medicală, București, 2006. ⇒28, 32

[5] B. Ganter, R. Wille, Conceptual scaling. In F. Roberts, editor, *Applications of Combinatorics and Graph Theory to the Biological and Social Sciences*, pp. 139–167, Berlin–Heidelberg–New York, 1989. Springer. ⇒27

[6] R. Godin, P. Valtchev, Formal concept analysis-based class hierarchy design in object-oriented software development, In B. Ganter, G. Stumme, and W. R., editors, *Formal Concept Analysis. Lecture Notes in Computer Science*, **3626**, pp. 304–323. Springer Berlin Heidelberg, 2005. ⇒23

[7] W. Hesse, T. Tilley, Formal concept analysis used for software analysis and modelling, **3626**, In *Formal Concept Analysis Used for Software Analysis and Modelling*, pp. 288–303. Springer-Verlag, Berlin, Heidelberg, 2005. ⇒23

[8] J. Janin, Surface and inside volumes in globular proteins, *Nature*, **277** (1979) 491–492. ⇒28

[9] J. Kyte, R. F. Doolittle, A simple method for displaying the hydropathic character of a protein, *Journal of Molecular Biology*, **157,** 1 (1982) 105–132. ⇒28

[10] J. Macko, Formal concept analysis as a framework for business intelligence technologies II, In *CUBIST Workshop*, 2013. ⇒23

[11] A. Molnar, V. Varga, C. Săcărea, D. Cîmpan, B. Mocian, Conceptual graph driven modeling and querying methods for RDBMS and XML databases, In *2017 13th IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*, pp. 55–62, 09 2017. ⇒23

[12] K. Raza. Formal concept analysis for knowledge discovery from biological data, *International Journal of Data Mining and Bioinformatics*, **18,** 4 (2017) 281–300. ⇒22, 23, 32

[13] G. D. Rose, A. Geselowitz, G. Lesser, R. Lee, M. Zehfus, Hydrophobicity of amino acid residues in globular proteins, *Science*, **229,** 4716 (1985) 834–838. ⇒28

[14] G. D. Rose. R. Wolfenden, Hydrogen bonding, hydrophobicity, packing, and protein folding. *Annual Review of Biophysics and Biomolecular Structure*, **22,** 1 (1993) 381–415. PMID: 8347995. ⇒28

[15] S. Rudolph, C. Săcărea, D. Troancă, Membership constraints in formal concept analysis, In *Proc. 24th Int. Conf. on Artificial Intelligence*, IJCAI'15, pp. 3186–3192. AAAI Press, 2015. ⇒23

[16] R. Wille, Restructuring lattice theory: An approach based on hierarchies of concepts, In I. Rival (ed.), *Ordered Sets*, pp. 445–470, Dordrecht, 1982. Springer Netherlands. ⇒25

[17] R. Wille, Conceptual knowledge processing in the field of economics, In W. R. Ganter B., Stumme G. (eds.), *Formal Concept Analysis. Lecture Notes in Computer Science*, **3626,** pp. 226–249. Springer Berlin Heidelberg, 2005. ⇒23

[18] R. Wolfenden, L. Andersson, P. Cullis, C. Southgate, Affinities of amino acid side chains for solvent wate, *Biochemistry*, **20** (1981) 849–855. ⇒28

[19] ∗∗∗ Rosalind project, `http://rosalind.info/glossary/amino-acid/`, 2019. ⇒32

[20] ∗∗∗ Wikipedia: Hydrophobicity scales, `https://en.wikipedia.org/wiki/hydrophobicity_scales`, 2019. ⇒28

Received: November 11, 2019 • Revised: January 6, 2020

Acta Univ. Sapientiae, Informatica **12,** 1 (2020) 39–50

DOI: 10.2478/ausi-2020-0003

Mouse dynamics based user recognition using deep learning

Margit ANTAL
Sapientia Hungarian University of Transylvania
Department of Mathematics–Informatics
Tirgu Mures
email: manyi@ms.sapientia.ro

Norbert FEJÉR
Sapientia Hungarian University of Transylvania
Department of Electrical Engineering
Tirgu Mures
email:
fejer.norbert@student.ms.sapientia.ro

Abstract. Behavioural biometrics provides an extra layer of security for user authentication mechanisms. Among behavioural biometrics, mouse dynamics provides a non-intrusive layer of security. In this paper we propose a novel convolutional neural network for extracting the features from the time series of users' mouse movements. The effect of two preprocessing methods on the performance of the proposed architecture were evaluated. Different training types of the model, namely transfer learning and training from scratch, were investigated. Results for both authentication and identification systems are reported. The Balabit public data set was used for performance evaluation, however for transfer learning we used the DFL data set. Comprehensive experimental evaluations suggest that our model performed better than other deep learning models. In addition, transfer learning contributed to the better performance of both identification and authentication systems.

Computing Classification System 1998: I.2.1
Mathematics Subject Classification 2010: 68T10
Key words and phrases: behavioural biometrics, mouse dynamics, deep learning, convolutional neural networks

1 Introduction

Behavioural biometrics provide an invisible layer of security for applications, and continuously authenticates users by analyzing the user's unique interactions with their devices. Mouse dynamics is a kind of behavioural biometrics which analyzes the users' mouse movements and detects intruders.

Most of the previous studies in mouse dynamics used machine learning methods with handcrafted features. In this study we propose deep neural networks that use raw mouse data, thus avoiding the typical feature extraction process.

Mouse data sets usually contain the following data about the mouse pointer: time, (x, y) coordinates and other auxiliary information about the buttons and the type of mouse event. When using handcrafted features in the feature extraction process, one has to use the auxiliary information in order to segment the raw data into meaningful mouse actions such as mouse movements or drag and drop operations. In contrast, our proposed architecture uses the raw data segmented into fixed-size units. Then, we used convolutional filters for extracting relevant features from the raw data. Instead of using the raw coordinates, we used directional velocities $(dx/dt, dy/dt)$, which are not only translation invariant, but produce significantly improved results.

Our contribution can be summarized as follows: (i) We proposed a new one-dimensional convolutional network architecture. (ii) We evaluated the impact on performance of two preprocessing methods for handling short mouse movement sequences. (iii) We evaluated the impact of different model training types. We compared transfer learning to training from scratch. These were performed for biometric identification as well as for biometric authentication. In addition, our research is reproducible: the data sets are publicly available and the results can be replicated with the software available on GitHub[1].

Following this section the most important research results in the field of mouse dynamics biometric are summarized. The third section presents our methods: data preprocessing, the architecture of our convolutional neural network, and the ways in which transfer learning were applied in this study. This is followed by a new section presenting the data sets, performance metrics, measurement protocol, as well as the identification and authentication results. The last section concludes the paper.

[1] `https://github.com/norbertFejer/AFE_Project`

2 Related works

Several behavioural biometrics are already implemented in operational authentication systems. These methods are most often used to continuously verify the user's identity. On-line courses use keystroke dynamics to continuously verify the identity of the registered users. While keystroke data may contain sensitive personal information, such as names or passwords, mouse dynamics do not contain sensitive data at all. In contrast to physiological biometrics which require the usage of a special sensor by the user, usually behavioural biometric data can be collected without the consent of the user.

One of the first studies regarding the performance of mouse dynamics authentication was written by Gamboa and Fred [8]. They implemented a memory game as a web application and collected the mouse interactions of the game users. Mouse interactions were segmented into so called mouse strokes defined as mouse movements performed between successive clicks. A set of 63 handcrafted features were extracted from these strokes. The feature extraction phase was followed by the learning phase which consisted of the estimation of the probability density functions of each user interaction. The system performance based on a sequence of 10 strokes was 11.8% EER (Equal Error Rate). Unfortunately, this data set is not publicly available.

The first publicly available mouse data set was published in 2007 by Ahmed and Traore [1], although this data set does not include raw data, but segmented and processed data. The data set contains general computer usage mouse data of 22 users, that is, users performed their daily work on their computers. Raw mouse data was segmented into three types of action: PC - point and click: mouse movement ending in a mouse click; MM - general mouse movement; DD - drag and drop. Histogram-based features were extracted from sequences of consecutive mouse actions. They reported on their data set of 22 users 2.46% EER using 2000 mouse actions for user authentication. The authors extended their data set to 48 users and published a new study on continuous authentication based on this extended data set [2].

Shen et al. published three papers in the topic of user authentication based on mouse dynamics [10], [11], [12]. Two data sets were also collected, one for static (57 subjects) and one for continuous user authentication (28 subjects) through mouse dynamics. Several machine learning and anomaly detectors were tested. Authentication performance having low equal error rates (below 1% EER) were obtained by using a large amount of mouse movement data (e.g. 30 minutes).

Zheng et al. also investigated the user authentication problem in their studies [13], [14]. They proposed some novel features such as angle based metrics. They obtained 1.3% EER using a sequence of 20 mouse actions. Unfortunately, their data sets containing general mouse usage data are also private.

Another study was conducted by Feher et al. [6]. They also collected their own dataset containing data from 25 subjects. Their best performance was 8.53% EER using a sequence of 30 mouse actions. All these studies were based on classical machine learning algorithms using some handcrafted feature sets.

The first study to use deep neural networks for mouse dynamics was published by Chong et al. [5]. They investigated one and two-dimensional convolutional neural networks (CNN) for mouse dynamics. While 1D-CNN network was trained by using the mouse movement trajectory's time series, the 2D-CNN network was trained using images of mouse movement trajectories. Despite the loss of time information in the case of 2D-CNN, this model outperformed both 1D-CNN and SVM models using handcrafted features. They extended their study [4] by considering Long Short-Term Memory (LSTM) and hybrid CNN-LSTM networks as well. Among these models the 2D-CNN model performed best resulting in a 0.96 average AUC (Area Under the Curve) for the Balabit data set.

3 Methods

3.1 Data preprocessing

A mouse dynamics data set consists of several log files containing mouse events with the following information: the x and y coordinates, the timestamp and the type of event. Based on the type of event we distinguish mouse move, mouse click, drag and drop and scroll actions. Usually a sequence of mouse movement events is ended in a mouse click, but there are mouse movement sequences without the ending click. A drag and drop operation performed by a user results in a sequence of drag mouse events. All mouse events contain the x and y coordinates of the mouse pointer with the exception of the mouse scroll event. Therefore, scroll events were not considered.

Mouse events were segmented into sequences. A sequence was ended when the time difference between two consecutive mouse events exceeded a threshold. These sequences were segmented into fixed sized blocks. When the length of the sequence is not a multiple of the block size we end up in a few shorter sequences. These shorter sequences can be dropped or can be concatenated to obtain full length blocks. Both cases were evaluated in our measurements.

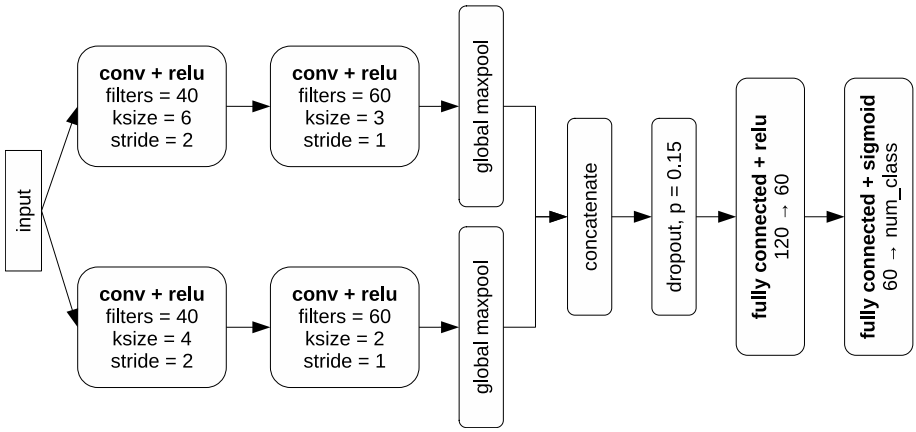


Figure 1: 1D-CNN architecture.

In order to obtain translation invariant mouse position sequences we decided to use speed values $(dx/dt, dy/dt)$ instead of absolute position coordinates (x, y).

3.2 1D-CNN

One dimensional convolutional neural networks (1D-CNNs) are used for time series modeling. As mouse movement sequences $x(t), y(t)$ are one dimensional time series, 1D-CNN models are well suited for modeling this type of signal. Our 1D-CNN architecture can be seen in Figure 1.

A tower model was used with different kernel sizes, which helped the network to learn input sequences on different time scales. We used the sigmoid activation function and a dropout layer with 0.15 probability to avoid overfitting. The network was trained in Keras [9] using the Adam optimizer (learning rate: 0.002, decay: 0.0001, loss function: binary cross-entropy). 16 epochs were used for training and a batch size of 32.

3.3 Transfer learning

Transfer learning is defined as reusing knowledge from previously learned tasks for the learning of a new task. This method is very popular in computer vision because it allows us to build accurate machine learning models faster. One may use a pre-trained model (a model trained on a large benchmark data set) instead of starting the learning process from scratch. In computer vision it is a

common practice to use well-proven models from the published literature. This means that both the architecture and the parameters of the model are reused. In this study we used transfer learning in a slightly different way. As a first step we developed our own model architecture. Thereafter we trained our model on a large data set and saved the model. This pre-trained model was reused for all the measurements performed on another data set. In conclusion, we transferred only the representation learning that is the knowledge of extracting the features.

4 Experiments

4.1 Data sets

In this study we used two public data sets: the Balabit Mouse Challenge data set [7] and the DFL data set [3].

The Balabit Mouse Dynamics Challenge data set contains timing and positioning information of mouse pointers. As the authors of the data set state, it can be used for evaluating the performance of user authentication and identification systems based on mouse dynamics. The data set contains mouse dynamics data of 10 users, and is divided into training and test sessions where the training sessions are much longer than the test sessions.

The DFL data set contains mouse dynamics data of 21 users (15 male and 6 female). The raw data format is similar to the Balabit data set therefore it contains timing and positioning information of mouse pointers. A data collector application was installed on the users' computers which logged their mouse dynamics data, therefore the acquisition of the data was uncontrolled. The sessions of this data set are not divided into training and test sessions. The details of the data set are available at: `https://ms.sapientia.ro/~manyi/DFL.html`.

Table 1 shows the quantity of data available for training using the two types of settings presented in the 3.1 section. The second column of the table shows the number of blocks available for each user of the data set when we drop the short sequences, and the third column contains the number of blocks in the case of concatenating the shorter sequences into full-size blocks.

4.2 Performance metrics

Accuracy is defined as the proportion of correctly predicted labels among the total number of testing samples. Although this is the most intuitive metric

User	Drop	Concatenate
7	2457	3119
9	2408	3081
12	459	1800
15	385	1098
16	871	1716
20	1269	1928
21	449	894
23	345	889
29	324	933
35	217	695

Table 1: Number of blocks for each user of the Balabit data set. Each block contains 128 mouse events.

when measuring the performance of a classifier, it is not always the best choice, e.g. when the data set is highly imbalanced. A commonly used metric when measuring the performance of biometric systems is the Receiver Operating Characteristics (ROC) curve. This curve plots the true positive ratio (TPR) against the false positive ratio (FPR), and the area under the curve (ROC AUC) is often used to compare the performances of different biometric systems.

4.3 Measurement protocol

As the acquisition of the DFL data set was uncontrolled, we decided to use this data set only for representation learning, which means that this data set was used to initialize the weights of our models (e.g. convolution kernels).

We evaluated both identification and authentication biometric systems. While the identification is a multi-class classification problem, authentication is a binary classification problem.

As described in section 3.1 mouse dynamics data was segmented into fixed sized blocks. There are big differences between users in terms of data volume. The user having the most data has ten times as much data as the user with the least data. Based on the amount of data used for the measurement, two types of measurements were made: (i) measurement using 300 blocks from each user – 300; (ii) measurement using all blocks of data form each user – ALL. While the first type is a class-balanced measurement, the second is a class-imbalanced measurement.

From the point of view of training the models, we distinguish three cases: (i) models trained from scratch using the training data from the Balabit data set – PLAIN models; (ii) models using the transfer learning - the models were pre-trained on the DFL data set – TRANSFER1 models; (iii) models initialised with transfer learning, then updating the weights using the training data from the Balabit data set –TRANSFER2. This case is similar to the PLAIN one. While in the first case we start with random weights, here we adjust the weights obtained from the TRANSFER1 model.

In the case of the identification measurements, we trained a single classifier using the training data (balanced - using the same number of blocks from each user or imbalanced using all the available data from each user), then we used the same number of test data from each user for computing the evaluation metrics.

In the case of the authentication measurements, we trained a separate model to each user using the same number of positive and negative data. In the first case (300), we took 300 positive blocks of data from a given user, then the same number of negative data was selected from the remaining users. The only user not having 300 blocks of data is user35 (see Table 1). In order to increase the number of training examples we used data augmentation. We added a random noise drawn from a uniform distribution in the range $[-\epsilon, \epsilon]$ to each signal (we used $\epsilon = 0.2$). Data augmentation was performed independently on $x(t)$ and $y(t)$ signals. In the second case (ALL), we considered all the positive data available from a given user, then the same number of negative data was selected from the remaining users.

Regardless of the measurement type we always separated 70 blocks of data from each user for evaluating the model. Therefore, all types of training were evaluated using the same amount of test data.

We used a single pre-trained model for transfer learning. This model was trained on the DFL data set. Therefore, we transferred the learned data representation from one data set to another.

All the evaluations were performed in Python 3.6.8 (Anaconda distribution) using Keras [9].

4.4 Results

4.4.1 Biometric identification

The effect of using a class-balanced subset (300 blocks/class) for evaluation compared to using all the available data is shown in Table 2. We evaluated

three types of models: PLAIN, TRANSFER1 and TRANSFER2. First of all, it can be seen that using all data resulted in lower performances than using a class-balanced subset of the available data. Secondly, we see that using transfer learning with frozen weights (TRANSFER1 - data representation was learned using another data set), resulted in much poorer identification rate than training the model from scratch. Thirdly, as we expected, the pre-trained model with updated weights (TRANSFER2) resulted in the best identification accuracies.

Number of blocks	PLAIN	TRANSFER1	TRANSFER2
300	0.63	0.50	0.66
ALL	0.55	0.34	0.62

Table 2: Identification results in terms of accuracy. Class-balanced subset vs. all data.

The results shown in the Table 2 were obtained using full sized mouse events blocks by dropping the shorter mouse event sequences (see subsection 3.1). The measurements were repeated for the other case where the training data included concatenations of shorter series. Table 3 shows the comparative results for the two cases.

Preprocessing	PLAIN	TRANSFER1	TRANSFER2
Drop	0.55	0.34	0.62
Concatenate	0.57	0.37	0.61

Table 3: Identification results in terms of accuracy using all data. Preprocessing type: concatenate vs. drop.

4.4.2 Biometric authentication

Tables 4 and 5 show the results of different authentication measurements in terms of accuracy and AUC respectively. Each performance is reported using the average performance value and in parenthesis the standard deviation. We can observe that there is no significant difference between PLAIN and TRANSFER2 results. This suggests that transfer learning does not significantly improve system performance. We can also notice that using a pre-trained model without updating the weights for the new data set (TRANSFER1) results in lower performance than training the model from scratch (PLAIN).

We should also notice that using all the available positive data for training (ALL) the models resulted in better performances for all types of training (see Figure 2). Not only are the average AUC values higher but the standard deviations are much more lower. This means that there are negligible differences in performance between users.

Number of blocks	PLAIN	TRANSFER1	TRANSFER2
300	0.86 (0.10)	0.80 (0.11)	0.87 (0.10)
ALL	0.93 (0.04)	0.79 (0.10)	0.93 (0.04)

Table 4: Authentication results in terms of accuracy. 300 vs. all data.

Number of blocks	PLAIN	TRANSFER1	TRANSFER2
300	0.92 (0.09)	0.86 (0.10)	0.93 (0.09)
ALL	0.98 (0.02)	0.87 (0.11)	0.98 (0.01)

Table 5: Authentication results in terms of AUC. 300 vs. all data.

We compared our best results with other results obtained on the Balabit data set using approximately the same size of mouse sequences for predicting the authenticity of the users. The comparison is shown in Table 6. It can be seen that our model has brought a significant improvement compared to Chong et al.'s [4] 1D-CNN model, moreover it is better than their optimized 2D-CNN model performance.

Paper	Model type	Average AUC
Chong, 2018 [5]	SVM	0.87
Chong, 2019 [4]	1D-CNN	0.90
Chong, 2019 [4]	2D-CNN	0.96
This	1D-CNN	0.98

Table 6: Comparison of authentication systems' performances on the Balabit data set.

5 Conclusions

In this study we proposed a novel 1D-CNN model for user authentication based on mouse dynamics. The advantage of our model over the classical machine learning model is that there is no longer need for ad-hoc features; the model is

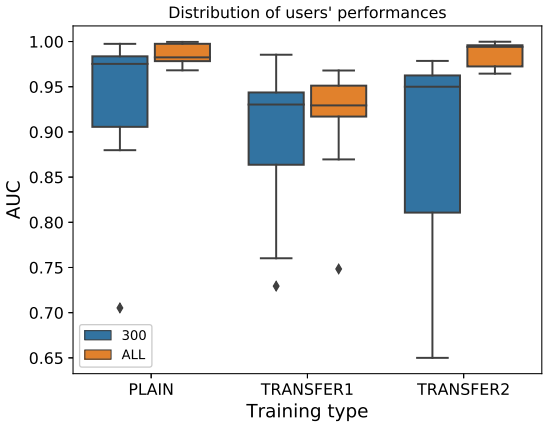


Figure 2: Authentication results for the Balabit dataset. Training data: 300 vs. all. Training methods: PLAIN, TRANSFER1, TRANSFER2. Each box shows the distribution of users's performances (AUC) using the given training data and method.

able to learn the features from raw data. However, we also demonstrated that transfer learning or learning the data representation on an independent large data set could improve the performance of the authentication system. The results show that our 1D-CNN model performs better than the other CNN models proposed for the same task.

Acknowledgements

The work of Norbert Fejér was supported by Accenture Industrial Solutions.

References

[1] A. A. E. Ahmed, I. Traore, A new biometric technology based on mouse dynamics, *IEEE Transactions on Dependable and Secure Computing* **4,** 3 (2007) 165–179. ⇒41

[2] A. A. E. Ahmed, I. Traore, Dynamic sample size detection in continuous authentication using sequential sampling, In *Proceedings of the 27th Annual Computer Security Applications Conference* ACSAC '11, pp. 169–176, New York, NY, USA, 2011. ACM. ⇒41

[3] M. Antal, L. Dénes-Fazakas, User verification based on mouse dynamics: a comparison of public data sets, In *2019 23th International Symposium on Applied Computational Intelligence and Informatics*, pp. 143–147, May 2019. ⇒44

[4] P. Chong, Y. Elovici, A. Binder, User authentication based on mouse dynamics using deep neural networks: A comprehensive study, *IEEE Transactions on Information Forensics and Security*, **15** (2020) 1086–1101. ⇒42, 48

[5] P. Chong, Y. X. M. Tan, J. Guarnizo, Y. Elovici, A. Binder, Mouse authentication without the temporal aspect – what does a 2d-cnn learn? In *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 15–21, May 2018. ⇒42, 48

[6] C. Feher, Y. Elovici, R. Moskovitch, L. Rokach, A. Schclar. User identity verification via mouse dynamics. *Inf. Sci.* **201** (2012) 19–362. ⇒42

[7] Á. Fülöp, L. Kovács, T. Kurics, E. Windhager-Pokol, Balabit mouse dynamics challenge data set, 2016. ⇒44

[8] H. Gamboa, A. Fred. A behavioral biometric system based on human-computer interaction. In *Proc. SPIE 5404, Biometric Technology for Human Identification, (25 August 2004)*, **5404**, pp. 381–392, 2004. ⇒41

[9] KERAS. Keras, 2016. ⇒43, 46

[10] C. Shen, Z. Cai, X. Guan, Continuous authentication for mouse dynamics: A pattern-growth approach, In *Proceedings of the 2012 42Nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, DSN '12, pp. 1–12, Washington, DC, USA, 2012. IEEE Computer Society. ⇒41

[11] C. Shen, Z. Cai, X. Guan, Y. Du, R. A. Maxion, User authentication through mouse dynamics, *IEEE Transactions on Information Forensics and Security*, **8,** 1 (2013) 16–30. ⇒41

[12] C. Shen, Z. Cai, X. Guan, R. A. Maxion, Performance evaluation of anomaly-detection algorithms for mouse dynamics, *Computers & Security* **45** (2014) 156–171. ⇒41

[13] N. Zheng, A. Paloski, H. Wang, An efficient user verification system via mouse movements, In *Proceedings of the 18th ACM Conference on Computer and Communications Security*, CCS '11, pp. 139–150, New York, NY, USA, 2011. ACM. ⇒42

[14] N. Zheng, A. Paloski, H. Wang, An efficient user verification system using angle-based mouse movement biometrics, *ACM Trans. Inf. Syst. Secur.*, **18,** 3 (2016) 11:1–11:27. ⇒42

Received: January 25, 2020 • Revised: February 16, 2020

Acta Univ. Sapientiae, Informatica **12,** 1 (2020) 51–69

DOI: 10.2478/ausi-2020-0004

Word pattern prediction using Big Data frameworks

Bence SZABARI
Eötvös Loránd University
Budapest, Hungary
email: n0qsdc@inf.elte.hu

Attila KISS
J. Selye University
Komárno, Slovakia
email: kissae@ujs.sk

Abstract. Using software applications or services, which provide word or even word pattern recommendation service has become part of our lives. Those services appear in many form in our daily basis, just think of our smartphones keyboard, or Google search suggestions and this list can be continued. With the help of these tools, we can not only find the suitable word that fits into our sentence, but we can also express ourselves in a much more nuanced, diverse way. To achieve this kind of recommendation service, we use an algorithm which is capable to recommend word by word pattern queries. Word pattern queries, can be expressed as a combination of words, part-of-speech (POS) tags and wild card words. Since there are a lot of possible patterns and sentences, we use Big Data frameworks to handle this large amount of data. In this paper, we compared two popular framework Hadoop and Spark with the proposed algorithm and recommend some enhancement to gain faster word pattern generation.

1 Introduction

Expressing ourselves in writing can be a challenging task, especially if we are not a native speaker of the target language. Fortunately, nowadays a lot of

Computing Classification System 1998: H.2, C.2
Mathematics Subject Classification 2010: 68U15
Key words and phrases: word-pattern, word-prediction, big data, hadoop, spark, nlp, map reduce, snappy, lz4, data compression

utilities can help us to express ourselves in a very diverse way, for example smart phone keyboards with word recommendation, online synonym dictionaries, or even the Google Search engine has the functionality to recommend the proper topic.

In this article, we will focus on word generation or to be more precise word pattern generation, which means that users can express their thoughts and ideas with the combination of word(s), part-of-speech (POS) tags [11] and with any arbitrary word (wild card word). Those kind of queries are called word pattern queries. For example:

$$VB * love$$

is a word pattern query that needs to satisfy the following requirements: a verb at the VB position, any arbitrary word at * position then word 'love'. The proper answer to that query is consists of the matched word list to the given pattern along with their relative frequencies of appearance in a large corpora.

In general, construction of word patterns can take a considerable amount of time, since there are a huge number of potential word patterns can be created based on a text, that contains just a few or even millions of rows.

Based on Erin Gilheany's comparison idea [1] and Kritwara Rattanaopas' data compression improvement [2], we were also interested in the efficiency of generating word patterns, therefore we made different experiments with Apache Hadoop [9] and Apache Spark [10] since both of them allows distributed processing of large data sets. We also made suggestions, to make the pattern generation faster in case of Hadoop. For this, we applied the Hadoop Native Library that can use Snappy and LZ4 data compression algorithms to compress the intermediate output of the mapper task. With the help of those compression codecs, we were able to achieve faster pattern generation.

2 Big data frameworks and paradigms

As we mentioned above, we have to process large amount of text files with thousand and millions of lines therefore we use Big Data frameworks like Apache Hadoop and Apache Spark.

2.1 Hadoop

Apache Hadoop [9] is a bundle of open-source software utilities that can solve problems involving large amounts of data and computation using a net-

work of many computers which are built from commodity hardware. It provides a framework for distributed storage and processing of big data using the MapReduce programming model.

The main parts of Hadoop are the distributed storage system called Hadoop Distributed File System (HDFS in short), and the processing part which is using MapReduce programming model, and YARN (Yet Another Resource Negotiator). Hadoop splits files into large block and distributes them across nodes in the cluster, then it sends the runnable code to the nodes to process data in parallel.

HDFS [13] is a distributed file system designed to run on commodity hardware. It has many similarities with existing distributed file systems, however it has also some key differences to the others. HDFS is highly fault-tolerant and is designed to be deployed on low-cost hardware which makes it economical. It also provides high throughput access to application data and is suitable for applications that have large data sets.

YARN (Yet Another Resource Negotiator) [12] is a cluster management system. It has been part of Apache Hadoop since v2.0. With the help of YARN arbitrary applications can be executed on a Hadoop cluster. Therefore, the application has to consist of one application master and an arbitrary number of containers. Latter are responsible for the execution of the application whereas the application master requests container and monitors their progress and status.

In order to execute these applications, YARN consists of two component types:

1. The **ResourceManager** is unique for a complete cluster. Its main task is granting the requested resources and balancing the load of the cluster. Furthermore, it starts the application master initially and restarts it in case of a failure.
2. On each computing node, one **NodeManager** is executed. It starts and monitors the containers assigned to it as well as the usage of its resources, i.e., CPU usage and memory consumption

2.2 MapReduce

MapReduce [8] is a programming model and an associated implementation for processing and generating big data sets with a parallel, distributed algorithm on a cluster.

A MapReduce program is composed of a map procedure (or method), which performs filtering and sorting, and a reduce method, which performs a sum-

mary operation. The "MapReduce System" (also called "framework") orchestrates the processing by marshalling the distributed servers, running the various tasks in parallel, managing all communications and data transfers between the various parts of the system, and providing for redundancy and fault tolerance.

A MapReduce framework is usually composed of three operations:

1. **Map** each worker node applies the map function to the local data, and writes the output to a temporary storage. A master node ensures that only one copy of the redundant input data is processed.
2. **Shuffle** worker nodes redistribute data based on the output keys (produced by the map function), such that all data belonging to one key is located on the same worker node.
3. **Reduce** worker nodes now process each group of output data, per key, in parallel.

MapReduce allows for the distributed processing of the map and reduction operations. Maps can be performed in parallel, provided that each mapping operation is independent of the others; in practice, this is limited by the number of independent data sources and/or the number of CPUs near each source.

The Map and Reduce functions of MapReduce are both defined with respect to data structured in (key, value) pairs. Map takes one pair of data with a type in one data domain, and returns a list of pairs in a different domain:

$$\mathtt{map}(\mathtt{k1}, \mathtt{v1}) \rightarrow \mathtt{list}(\mathtt{k2}, \mathtt{v2})$$

The Map function is applied in parallel to every pair (keyed by k1) in the input dataset. This produces a list of pairs (keyed by k2) for each call. After that, the MapReduce framework collects all pairs with the same key (k2) from all lists and groups them together, creating one group for each key.

The Reduce function is then applied in parallel to each group, which in turn produces a collection of values in the same domain:

$$\mathtt{reduce}(\mathtt{k2}, \mathtt{list}(\mathtt{v2})) \rightarrow \mathtt{list}((\mathtt{k3}, \mathtt{v3}))$$

Thus the MapReduce framework transforms a list of (key, value) pairs into another list of (key, value) pairs. This behavior is different from the typical functional programming map and reduce combination, which accepts a list of arbitrary values and returns one single value that combines all the values returned by map.

2.3 Spark

Apache Spark [10] is an open-source distributed general-purpose cluster-computing framework that can do in-memory data processing, while providing the ability to develop applications in Java, Scala, Python or even in R. Spark provides four main submodules which are SQL, MLib for machine learning, GraphX and Streaming.

In this paper, we focus on the SQL module, especially on the new data structure called DataFrame [16] which has been added in Spark 1.6. Dataframe was built on top of the previously used **RDDs** (Resilient Distributed Dataset) therefore, it is combining the benefits of RDDs (strong typing, ability to use lambda functions) and the benefits of Spark SQL's optimized execution engine.

It is also possible to use functional transformation such as map, flatMap, filter, etc. to manipulate these kind of datasets. Using those mapper functions are essentials during the word pattern generations.

2.4 Hadoop vs Spark

Spark is developed to run on top of Hadoop and this is an alternative to the traditional MapReduce model. The key differences are the following:

- Spark stores, a process data in-memory while Hadoop using the disk
- Hadoop uses the MapReduce paradigm and Spark uses the distributed data structure called datasets which are built on RDDs
- Hadoop merges and partitions shuffle spill files inte one big files, while Spark doesn't
- The MapReduce can be inefficient when the job has to reuse the same dataset, while Spark can hold the data in memory for efficient reuse

3 Word recommendation

3.1 Word recommendation problem

The applied method can help to choose the proper words in a certain context, by recommending a list of suitable words that can fit in the given words. A word pattern query is an ordered sequence of specific word(s), POS tag(s), and wild card word(s). For example:

$$VB * love$$

For each word pattern query, we want to get a list of frequently used words that match the POS tags. Furthermore, if we include the relative frequencies of the matched words that can help you to decide the right phrase for you.

The conventional language models can suggest the most appropriate words that can appear at a specific position of a sentence or phrase. However, those models are usually inappropriate for a word pattern query service. The problem with those models:

- they are not built to estimate the probability distributions for more than one words
- they are not able to understand the usage of POS tags

Therefore, they can not have any relationships between POS tags and words. In this case, we would like to model the following probability distribution $p(W_t|W_c)$ for matching word list W_t given context words W_c:

$$p(W_t|W_c) = \frac{C(W_t, W_c)}{C(W_c)}$$

where W_t denotes a list of words corresponding to POS tags in a word pattern query, $C(W_t, W_c)$ indicates the number of sentences that match the word pattern query with the words of Wt in the positions of POS tags, and $C(W_c)$ indicates the total number of sentences that match with the word pattern query. W_t part can be greater than 1, i.e., $|W_t| > 1$.

3.2 Used method for word recommendation

To create word pattern queries we used the presented algorithm in [3], which generates word patterns along with their associated information earlier on.

Word patterns can contain words, POS tags and wild card words denoted by a symbol (*). The wild card word can be any English word, so it can be useful when we are only focus on the other words within the word pattern, as they are just placeholders.

3.3 Workflow

As an overview, to construct word patterns the used method consists of the following steps:

Figure 1: Word pattern generation process

3.4 Preprocessing and tagging

To be able to create word patterns, first we have to transform the input sample texts. It may occur the sample texts contain non-English words or characters in this case they need to be removed. Numbers and punctuation marks can remain in the text.

After the transform phase has been finished, we are finally able to create POS Tagged sentences, using the Stanford POS Tagger [17]. This tool can read text and assign parts-of-speech to each word or token, e.g. noun, verb, preposition and so on. The possible POS Tags which is supported by tagger are listed in Table 2.

For example, if we have the following sentence part, after the preprocessing:

"mathematical notation widely used in physics and other sciences avoids many ambiguities compared to expression in natural language however for various reasons several lexical syntactic and semantic ambiguities remain"

Stanford POS tagger will create the output as shown below:

"mathematical/JJ notation/NN widely/RB used/VBN in/IN physics/NN and/CC other/JJ sciences/NNS avoids/VBZ many/JJ ambiguities/NNS compared/VBN to/TO expression/NN in/IN natural/JJ language/NN however/RB for/IN various/JJ reasons/NNS several/JJ lexical/JJ syntactic/NN and/CC semantic/JJ ambiguities/NNS remain/VBP"

3.5 Generating word patterns

After we got the tagged sentences, the used method creates the word patterns which consist of words, POS tags, and the wild card word symbol (*). It generates all word patterns for each n-gram of the POS-tagged sentences, and then aggregates them to get the information for word patterns.

The possible word patterns which can be generated from n-grams are listed in Table 3. To create the possible word patterns, we have to define constraints to reduce the amount of patterns to get a manageable amount of them. Therefore, we apply the following rules: there is at least one word in a pattern, if there is a wild card within the pattern it does not appear in the first or the last position in a pattern. Also, we reduce the number of possible n-grams to 5 for the reasons mentioned above.

After all word patterns of k-grams

$$k \in [2, 5]$$

are created for the provided input text, the used method clusters them into groups to have the same word pattern in each group. For each group the sentence Ids which have the same word lists are grouped together and their count is computed.

We are interested in the most frequent word list for each word patterns to be able to create a word recommendation service, so the used method constructs a word pattern database which maintains the frequent word lists along with their corresponding sentence Ids and their frequency for each word pattern.

3.6 Map reduce solution

The method use two cycles of Map-Reduce tasks. In the first Map-Reduce phase, the mapper receives sentences with their Ids and produces the key-value pairs where the key is made of a word pattern and its corresponding word list while the value is the sentence Id. Futhermore, the reducer of the first cycle aggregates the sentence Ids of word patterns generated by mapper. In the second cycle, the mapper computes the frequencies of the combination of a word pattern and its word lists, and then the reducer aggregates the results and also retains the top k-th word lists with the highest frequencies.

The following procedure is the algorithm for the first mapper in the first phase. It creates the word patterns for 2-grams to 5 grams for each sentence. The cands variable holds the possible word patterns, so $\mathtt{cands}[n][i][j]$ indicates the j-th value for the i-th word list of the n-gram.

In the first mapper-phase, we process POS tagged lines (assuming that we have the sentence Ids) and as an output the algorithm will create a pairs of

$$([\mathtt{word\ pattern}, \mathtt{word\ list}], \mathtt{sentence\ id})$$

where the word pattern is generated based on the possible word patterns described in Table 3, while word-list contains the matched words corresponding to the pattern and the value will be the sentence id.

Algorithm 1: Mapper phase 1

```
Input  :  (sentence id, sentence with POS tags)
Output:   ([word-pattern, word-list], sentence id)
value ← sentenceId;
tokenize the POS tagged sentence, into a collection of tokens;
tokens ← t1, t2, ... t3;
m ← tokens.size;
for n ← 2 to 5 do
    for s ← 0 to m−n do
        for i ← 0 to cands[n].size − 1 do
            pattern ← [];
            metWords ← [];
            for j ← 0 to n − 1 do
                word ← word at (s+j) th position of tokens;
                pos ← POS Tag at (s+j) th position of tokens;
                // j-th position is a word Tag in the pattern
                if cands[n][i][j] = 'w' then
                    pattern.add(word);
                // j-th position is a POS Tag in the pattern
                else if cands[n][i][j] = 'p' then
                    pattern.add(pos);
                // j-th position is a wildcard in the pattern
                else
                    pattern.add(_WC_);
                end
                if cands[n][i][j] > '0' and pos is a legal tag then
                    metWords.add(word);
            end
            if metWords.isEmpty() then
                metWords.add(_NONE_);
            key ←pattern + ";" + metWords;
            emit(key, value)
        end
    end
end
```

3.7 Example

Let's take an example, in that case when we have the following short sentence: *lincoln/NN practiced/VBD law/NN*. On the next page, we present some sample output. For the sake of simplicity and transparency, we categorized the output by n-grams and omitted the sentence ids.

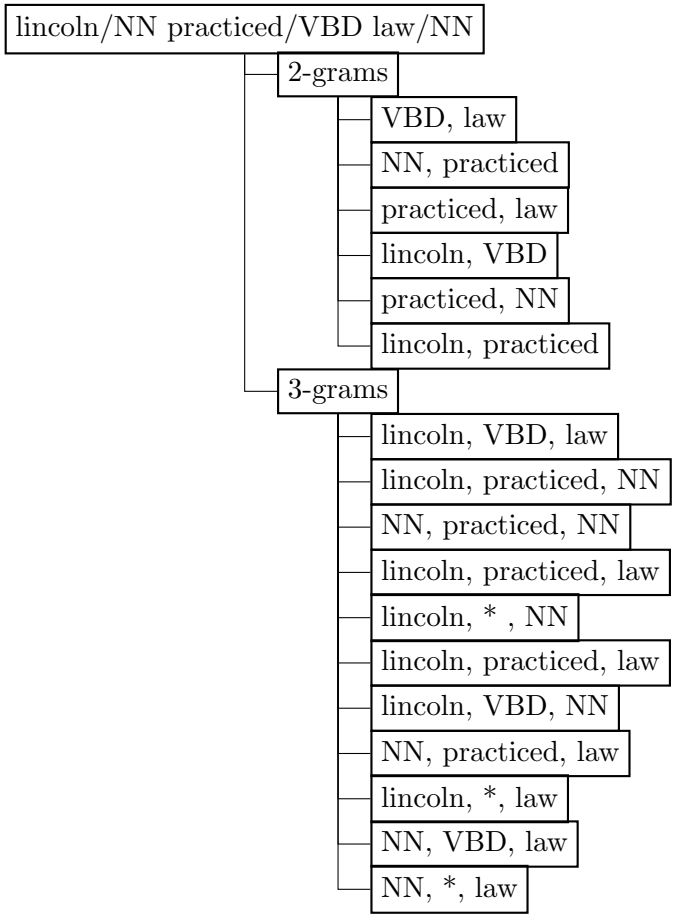


Figure 2: Word patterns example

3.8 Reducer phase 1

The first reducer phase aggregates the sentence ids according to the values of (pattern, word-list). It also removes the duplicated occurences of the same id for the same key.

3.9 Mapper phase 2

The second mapper calculates the relative frequencies of the different matched word list and also extract the pattern from the old key to use as a new key.

Algorithm 2: Reducer phase 1

Input : pairs of ((pattern, word list), sentence id)
Output: ([pattern, word-list]), sentence-id-list)
$key \leftarrow [pattern, word\ list]$;
$value \leftarrow sentence\ id\ list$;
emit($key, value$)

Therefore, we will get pairs like this:

$$word\ pattern \rightarrow (f_1, w_1, sid_1),\ \ldots\ , (f_n, w_n, sid_n)$$

where f_i denotes the ith relative frequency, w_i the ith matched word list, and sid_i is the ith the sentence id.

Algorithm 3: Mapper phase 2

Input : ([pattern, word-list]), sentence-id-list)
Output: pattern, [frequency, word list, sentence id list])
if *word-list.isEmpty()* **then**
  $word\ list$.add(_NONE_);
$key \leftarrow$pattern;
$value \leftarrow [frequency, word\ list, sentence\ id\ list]$;
emit($key, value$)

3.10 Reducer phase 2

In the last reducer phase we remain the top k-th (in our example 10) matched word list with the highest relative frequency. The received result can be the basis of a word pattern database.

4 Hadoop native library

Hadoop can support native libraries [14] out of the box, which includes components like compression codecs (e.g.: bzip2, lz4, snappy and zlib), native io utilities for Centralized Cache Management in HDFS or even CRC32 checksum implementation.

From this library, we mainly focus on the compression codecs that we can use to compress the Mapper job's intermediate output in a Hadoop MapReduce.

Algorithm 4: Reducer phase 2

Input : pattern, [[frequency, word-list, sentence-id-list] . . .]
Output: pattern, [[frequency, word-list, sentence-id-list] . . .] (top k-th)
$key \leftarrow$pattern;
$value \leftarrow$ [];
$list \leftarrow [[frequency, word-list, sentence-id-list] \ldots]$;
sort the $list$ by frequency, in decreasing order;
put the first k-th element from $list$ to $value$;
emit($key, value$)

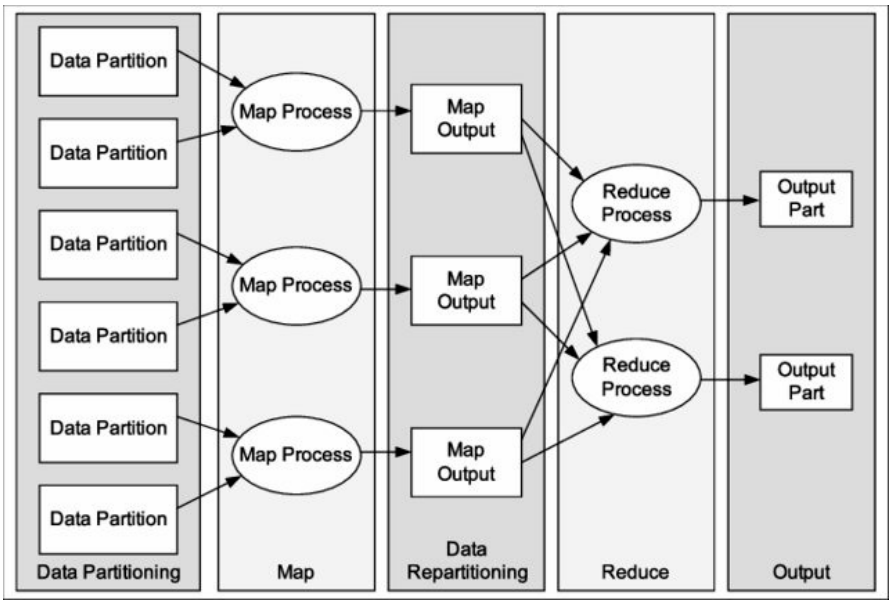


Figure 3: Computing Stages of the MapReduce model [4]

4.1 Compression codecs

4.2 Snappy

Snappy (previously known as Zippy) [7] is a fast data compression and decompression library written in C++ by Google based on ideas from LZ77 and open-sourced in 2011. It does not aim for maximum compression, or compatibility with any other compression library; instead, it aims for very high speeds

and reasonable compression. The compression ratio is 20–100% lower than gzip.

4.3 Lz4

LZ4 [5] is a lossless data compression algorithm that is focused on compression and decompression speed. It belongs to the LZ77 family of byte-oriented compression schemes.

The LZ4 algorithm represents the data as a series of sequences. Each sequence begins with a one-byte token that is broken into two 4-bit fields. The first field represents the number of literal bytes that are to be copied to the output. The second field represents the number of bytes to copy from the already decoded output buffer (with 0 representing the minimum match length of 4 bytes). A value of 15 in either of the bitfields indicates that the length is larger and there is an extra byte of data that is to be added to the length. A value of 255 in these extra bytes indicates that yet another byte to be added. Hence arbitrary lengths are represented by a series of extra bytes containing the value 255. The string of literals comes after the token and any extra bytes needed to indicate string length. This is followed by an offset that indicates how far back in the output buffer to begin copying. The extra bytes (if any) of the match-length come at the end of the sequence.

5 Experiments

As we introduced, we used Apache Hadoop and Apache Spark to measure execution times generating word patterns. During these experiments we also applied several data compression libraries to achieve faster pattern generation. To measure those generation times and see how efficient can be a selected profile we used Monte Carlo method. [6] In our experiments, the following scenarios were compared: standard Hadoop Mapreduce job without any compression codecs, Hadoop Mapreduce job with Snappy compression codec, Hadoop Mapreduce job with LZ4 compression codec, and standard Spark job using dataframes. Furthermore, we used Wikipedia dumps to provide a suitable input for our measurements. [18]

5.1 Experiment settings

The experiments were run on a 21 node cluster with the following configuration: master node has 32 Gb RAM, 12 vCPU while slaves have 15 Gb ram, 8

vCPU. In both cases, Yarn was used as a resource manager and its configuration Node Managers were set to allocate **13** Gb for Containers. The Mapper Container's and the Reducer Container's memory size size were set to **4** Gb. Additional configuration and settings, and even more technical detail can be found at the project git repository [15]

5.2 Results

In the first measurements we compared the standard Hadoop MapReduce jobs against MapReduce jobs with compression codecs.

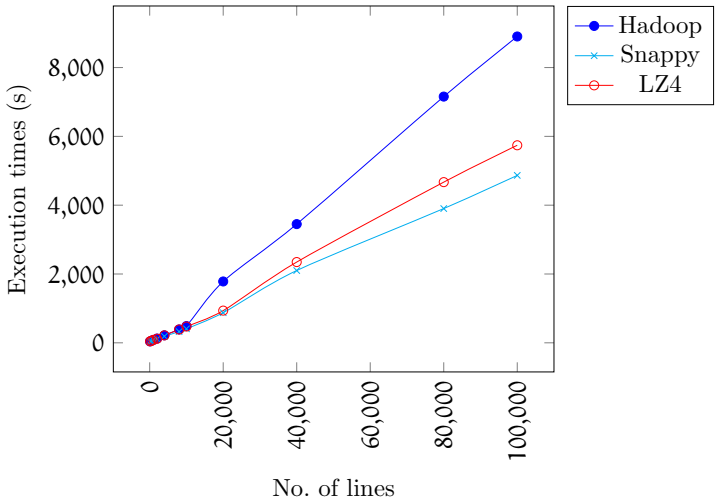


Figure 4: Hadoop Word pattern generation

Either using LZ4 or Snappy, both gave us significantly better results, but Snappy was the most spectacular codec for improvement. Therefore we conclude that, using those compression libraries can reduce the overhead during the word pattern generation and we can get better execution times. In the next phase, we used the Spark with Scala implementation of the word pattern generation and compared against a standard Hadoop job.

As we can see there is a significant difference between the two run times. In the last measurement, we compared all the previously used profile.

In summary, Snappy and the LZ4 compression codecs brought a convincing improvement during the pattern generation. Also, using the Spark framework was able to beat his rival, however it is still possible to have different SparkSQL query optimization so we can reduce word pattern generation time.

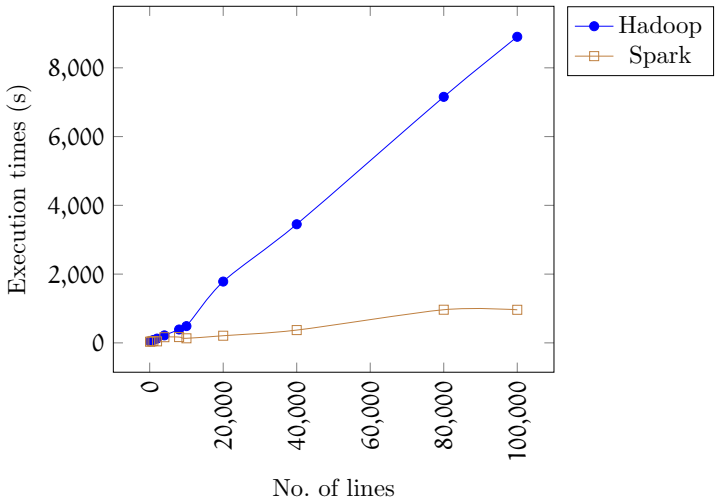


Figure 5: Hadoop compared to Spark

No. of lines	No. of words	No. of unique words	Hadoop	Hadoop with Snappy	Hadoop with LZ4	Spark
100	7045	2578	41 s	40 s	42 s	33 s
500	34 206	9351	58 s	53 s	59 s	44 s
1000	73 963	17 241	84 s	75 s	85 s	56 s
2000	143 092	28 502	124 s	108 s	119 s	44 s
4000	305 789	50 010	218 s	189 s	216 s	165 s
8000	588 912	79 933	389 s	326 s	386 s	169 s
10 000	738 533	93 828	487 s	407 s	473 s	133 s
20 000	1 497 350	151 253	1783 s	876 s	936 s	208 s
40 000	2 912 093	237 531	3450 s	2101 s	2347 s	372 s
80 000	5 858 805	374 986	7156 s	3902 s	4670 s	964 s
100 000	7 192 730	438 023	8903 s	4866 s	5741 s	962 s

Table 1: Experiment results

6 Conclusion

This paper introduced a word recommendation problem [3] where the user can express themselves using word pattern queries that contain words, POS tags and wild cards and as a result they get back matched word lists along with their relative frequencies.

For this problem, we proposed several fine tuning to reduce the overhead during the word pattern generation. To achieve this, we first introduced the Hadoop Native Library which contains compression codecs like Snappy, LZ4,

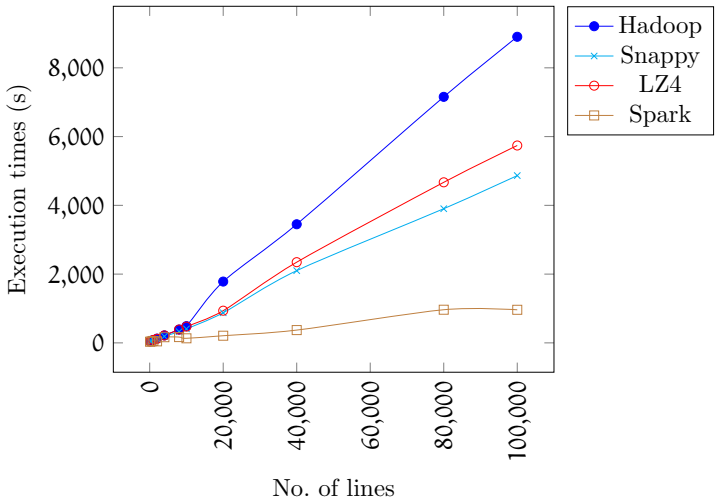


Figure 6: Word pattern generation summary

ZLib then we chose the first two of them, to see how they act during pattern generation, and then we compared the Spark framework with Hadoop.

7 Future works

As we pointed out in case of Spark there is still options to get better execution times than Hadoop. For example, fine tuning the built-in Spark Catalyst query optimizer. From the perspective of Hadoop, there are several compression codecs that we have not tested, and we used the default configuration, therefore a more advanced configuration / profile can be a good starting point for further examinations.

8 Acknowledgements

The project has been supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.3-VEKOP-16-2017-00002)

This publication is the partial result of the Research & Development Operational Programme for the project "Modernisation and Improvement of Technical Infrastructure for Research and Development of J. Selye University in the Fields of Nanotechnology and Intelligent Space", ITMS 26210120042, co-funded by the European Regional Development Fund.

References

[1] G. Erin. *Processing time of TFIDF and Naive Bayes on Spark 2.0, Hadoop 2.6 and Hadoop 2.7: Which Tool Is More Efficient?*, Msc Thesis, National College of Ireland Dublin, 2016. ⇒52

[2] K. Rattanaopas, S. Kaewkeeree. Improving Hadoop MapReduce performance with data compression: A study using wordcount job, *2017 14th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology* (ECTICON). IEEE, 2017. p. 564-567 ⇒52

[3] KM. Lee, CS. Han, KI. Kim, SH. Lee, Word recommendation for English composition using big corpus data processing, Cluster Computing, (2019), 1911-1924. ⇒56, 65

[4] M. Kontagora, H. Gonzalez-Velez, Benchmarking a MapReduce Environment on a Full Virtualisation Platform, *The 4th International Conference on Complex, Intelligent and Software Intensive Systems*, 433-438. 10.1109/CISIS.2010.45. ⇒62

[5] M. Bartík, S. Ulbik, P. Kubalik Matěj. LZ4 compression algorithm on FPGA, *2015 IEEE International Conference on Electronics, Circuits, and Systems (ICECS). IEEE, 2015* ⇒63

[6] RY Rubinstein, DP. Kroese, Simulation and the Monte Carlo method. Vol. 10. John Wiley & Sons, 2016. ⇒63

[7] R Lenhardt,J Alakuijala, Gipfeli-high speed compression algorithm. *2012 Data Compression Conference (pp. 109-118). IEEE* ⇒62

[8] H. Karloff, S. Suri, S. Vassilvitskii, A model of computation for MapReduce. *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms. Society for Industrial and Applied Mathematics, 2010.* ⇒53

[9] Apache Hadoop, Apache, `https://hadoop.apache.org/` ⇒52

[10] Apache Spark, Apache, `https://spark.apache.org/` ⇒52, 55

[11] E. Brill, A simple rule-based part of speech tagger, *Proceedings of the third conference on Applied natural language processing. Association for Computational Linguistics, 1992.* ⇒52

[12] Apache Yarn, Apache, `https://hadoop.apache.org/docs/current/hadoop-yarn/hadoop-yarn-site/YARN.html` ⇒53

[13] Apache HDFS docs, `https://hadoop.apache.org/docs/r1.2.1/` ⇒53

[14] Hadoop Native Library, `https://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/NativeLibraries.html` ⇒61

[15] Project repository, `https://gitlab.com/thelfter/word-prediction` ⇒64

[16] Spark Sql, `https://spark.apache.org/docs/latest/sql-programming-guide.html` ⇒55

[17] Stanford part-of-speecg tagger, `https://nlp.stanford.edu/software/tagger.html` ⇒57

[18] Wikipedia dumps, `https://dumps.wikimedia.org/` ⇒63

9 Appendix

Table 2: Part-of-speech (POS Tags)

CC coordinating conjunction	POS possessive ending
CD cardinal number	PRP personal pronoun
DT determiner	PRP$ possessive pronoun
EX existential there	RB adverb
FW foreign word	RBR adverb, comparative
IN preposition or subordinating conjunction	RBS adverb, superlative
JJ adjective	RP particle
JJR adjective, comparative	SYM symbol
JJS adjective, superlative	TO to
LS list item marker	UH interjection
MD modal	VB verb, base form
NN noun, singular or mass	VBD verb, past tense
NNS noun, plural	VBG verb, gerund or present participle
NNP proper noun, singular	VBN verb, past participle
NNPS proper noun, plural	VBP verb, non-3rd person singular present
PDT predeterminer	VBZ verb, 3rd person singular present
WDT Wh-determiner	WP$ possessive wh-pronoun
WP Wh-pronoun	WRB Wh-adverb

Table 3: Possible word pattern: w word, p POS tag, * wild card

2-gram (3 cases)	w w	w p	p w
3-gram (10 cases)	w w w	w p p	
	p w w	p * w	w w p
	w * w	p w p	w p w
	w * p	p p w	
4-gram (32 cases)	w w w w	w w w p	w w p w
	w w p p	w w * w	w w * p
	w p w w	w p w p	w p p w
	w p p p	w p * w	w p * p
	w * w w	w * w p	w * p w
	w * p p	w * * w	w * * p
	p w w w	p w w p	p w p w
	p w p p	p w * w	p w * p
	p p w w	p p w p	p p p w
	p p * w	p * w w	p * w p
	p * p w	p * * w	

5-gram (100 cases)	w w w w w	w w w w p	w w w p w
	w w w p p	w w w * w	w w w * p
	w w p w w	w w p w p	w w p p w
	w w p p p	w w p * w	w w p * p
	w w * w w	w w * w p	w w * p w
	w w * p p	w w * * w	w w * * p
	w p w w w	w p w w p	w p w p w
	w p w p p	w p w * w	w p w * p
	w p p w w	w p p w p	w p p p w
	w p p p p	w p p * w	w p p * p
	w p * w w	w p * w p	w p * p w
	w p * p p	w p * * w	w p * * p
	w * w w w	w * w w p	w * w p w
	w * w p p	w * w * w	w * w * p
	w * p w w	w * p w p	w * p p w
	w * p p p	w * p * w	w * p * p
	w * * w w	w * * w p	w * * p w
	w * * p p	w * * * w	w * * * p
	p w w w w	p w w w p	p w w p w
	p w w p p	p w w * w	p w w * p
	p w p w w	p w p w p	p w p p w
	p w p p p	p w p * w	p w p * p
	p w * w w	p w * w p	p w * p w
	p w * p p	p w * * w	p w * * p
	p p w w w	p p w w p	p p w p w
	p p w p p	p p w * w	p p w * p
	p p p w w	p p p w p	p p p p w
	p p p * w	p p * w w	p p * w p
	p p * p w	p p * * w	p * w w w
	p * w w p	p * w p w	p * w p p
	p * w * w	p * w * p	p * p w w
	p * p w p	p * p p w	p * p * w
	p * * w w	p * * w p	p * * p w
	p * * * w		

Received: January 31, 2020 • Revised: February 28, 2020

Acta Univ. Sapientiae, Informatica **12,** 1 (2020) 70–83

DOI: 10.2478/ausi-2020-0005

Opportunity activity sequence investigations in B2B CRM systems

Doru ROTOVEI
University of West
Timisoara, Romania
email: doru.rotovei80@e-uvt.ro

Abstract. Closing a deal in a business to business environment implies a series of orchestrated actions that the sales representatives are taking to take a prospective buyer from first contact to a closed sale. The actions, such as meetings, emails, phone calls happen in succession and in different points in time relative to the first interaction.

Time-series are ordered sequences of discrete-time data. In this work, we are examining the relationship between the actions as time series and the final win outcome for each deal. To assess whether the behavior of the salespeople have a direct influence on the final outcome of the current deal, we used histogram analysis, dynamic time warping and string edit distance on a real-world Customer Relationship Management System data set. The results are discussed and included in this paper.

1 Introduction

The sales process in the Business to Business (B2B) environment consists of a series of steps and actions intended to take a prospective buyer from initial interest to a closed sale. The series of actions the salespeople take can damage or improve the odds of successfully closing the deal. Therefore analyzing these

Computing Classification System 1998: G.3
Mathematics Subject Classification 2010: 68R15
Key words and phrases: time series; CRM; B2B; machine learning; string edit distance, dynamic time warping;

series of actions can reveal insights into how to successfully close a deal that can be taught and shared with the entire sales team.

Closing a deal in B2B environment is different than selling Business to Consumer (B2C). In B2C, the target is presumably millions of people versus B2B where the potential customers are considerably few. In both B2C and B2B product knowledge is needed however in B2B the product knowledge of the sales representative has to go deeper into design details, advantages, disadvantages and competitors' knowledge as the buyers understand the complexity of the products and their sophistication. In B2C usually one decision maker is involved, in B2B orders are considerably higher in monetary value, multiple decision-makers need to be convinced which leads to a longer time period to close the deal [3, 21]. Therefore a system is needed to keep track of all these activities. The tracking of all the information, requirements, products of interest, notes and, in general, entire interaction with the potential customer is tracked using an opportunity entity inside a Customer Relationship Management (CRM) System [8].

Each activity (such as meetings, emails, phone calls) that a sales representative is performing for an opportunity is usually recorded with their associated notes for historical and recollection reasons. We can note that each activity is performed in a precise point in time relative to the creation of the CRM opportunity. Furthermore, the time gap between activities and the type of activity (email, phone call) could play a role in the final outcome of the deal.

How much information is recorded in the series of activities for each opportunity? Is there a pattern of a successful sale in the way a deal is won? Can we distinguish the series of activities that lead to lost sales versus won sale? We set ourselves to find out the answer to these questions by doing a time series analysis of the activities performed for each won and lost opportunity.

Our hypothesis is that won deals and lost deals can be clustered together, in other words, there is a distinctive pattern of activities performed for won deals and for lost deals. This hypothesis, if true, could help us improve the B2B sales prediction models that we previously studied in [17, 16].

This paper is organized as follows: section 2 describes related work on B2B time series analysis, section 3 introduces our methodology and the problem domain while sections 4 and 5 presents our results, conclusion and future work.

2 Literature review

In the last few years, researchers have become increasingly interested in improving B2B selling using CRM data analytics [14]. Therefore in this section, we will look first at the B2B selling and the influencers of a successfully closed

deal followed by sales forecasting and the methods used to predict future business income.

2.1 B2B selling

B2B sales are driven by the relationship that is established between the salesperson and the prospect. One study found that the relationship is stronger and the actions more impactful when the product or service is more critical to the customer and when the sales person is directly interacting with an individual decision maker instead of the firm [13]. In another study, the performance of 816 salespeople in 30 sales organizations has been conducted to research the influence of the company strategy towards the final sales outcome. It has been found that the company strategy influences considerably the individual salespeople performance especially related to the prioritization, customer orientation and value-based selling [19].

The clarity of the sales role, no ambiguity and lack of conflict, has a direct positive influence on the job performance which in turn influences the quota attainment and the individual opportunity success rate [9].

The ethical behavior of salespeople has been studied in [20] where it has been found that ethical behavior is linked to increase sales and customer satisfaction as well as product or service quality. Likewise, the sales person adaptability during the sale has a strong correlation to the success of the sale [2]. Adaptability requires the examination of the internal and the external activity that trigger changes in behavior to successfully close a sale.

There is also momentum in a closing deal. After each stage and after each customer interaction a successful sales representative will ask for an agreement for the next step. If the potential customer is moving along and the set milestones are hit, more than likely the successful sale will happen [4].

In this study, we are looking at the behavior of salespeople as recorded in the series of activities performed on each opportunity and their relationship to the final outcome. The behavior, even influenced by the company strategy or the adaptability of the sales person, if consistent can shed light on the steps necessary to successfully close a deal.

2.2 Sales forecasting

Forecasting is the process of predicting future events based on the information and events that already occurred [1]. Sales forecasting is a prediction of future sales performance using the information known today from marketing conditions to sales pipeline analysis.

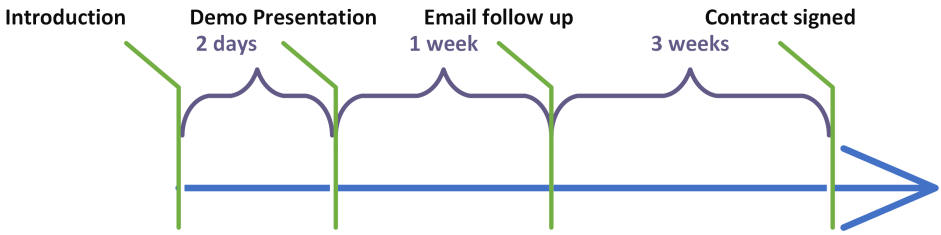


Figure 1: An example of activities succession to close an opportunity

As sales are the lifeline of any business, a significant amount of research has been done to predict future sales using time series analysis in different industries such as fashion, e-commerce, appliances to name a few [12, 11, 22]. Even more, combining forecasting from different models has shown to perform better than a single model forecast for most time series [6].

However, to best of our knowledge, no study has looked specifically at the impact of the individual activities for each opportunity and how the implied behavior might affect the final lost/win outcome.

3 Problem domain and methodology

B2B sales take time to close. To keep track of the progress on each opportunity, sales representatives are recording notes along with their activities. The customer interaction history might include quotes and proposals, product codes and volumes needed, delivery requirements or product configuration information. All in all, the opportunity entity contains the stages, milestones and key activities performed for each opportunity.

Sales representatives, over time, develop a set of steps and activities that they execute to close a deal that could be as personal as a fingerprint. For example one sales representative might use a demo meeting, following with an email, waiting for a week and making a phone call, whereas another sales representative might be more aggressive and after the demo follow up with a phone call and two days later with another voice mail. Part of the behavior of the sales representative is recorded in the succession of the activities performed and also in the length of time between each activity until the opportunity is closed (see Fig. 1).

Our study builds up by starting first with histogram analysis of the lost/won deals, subsequently by time series analysis using dynamic time warping and wraps up with string edit distance.

3.1 Histogram analysis

A histogram shows the frequency distribution of a continuous data set. Using the shape bar bins which represent the intervals of the continuous data, the data is plotted bidimensionally against the frequency [7].

In the first part, we created histograms of the won deals and of the lost deals. The histograms, if different, can show us these differences between the activity patterns for lost deals compared to the activity patterns for the won deals.

3.2 Dynamic time warping

Our subsequent approach to discover similarities between won activities and lost activities was to use dynamic time warping (DTW) which is an algorithm used for comparison of time series. In essence, given two time series of activities the algorithm stretches or compresses the series along the time axis in order to resemble each other as much as possible while simultaneously measuring the similarities between the two time series.

Formally, given two time series of length n and respectively m:

$$\begin{aligned} X &= x_1, x_2, \ldots, x_n \\ Y &= y_1, y_2, \ldots, y_m \end{aligned} \tag{1}$$

we construct a wrap path W

$$W = w_1, w_2, \ldots, w_K \qquad max(m, n) \leq K << m + n \tag{2}$$

where K is the length of the wrap and the k element of the wrap path is:

$$w_k = (i, j) \qquad i = 1 \ldots n, j = 1 \ldots m \tag{3}$$

the distance of the wrap path W is:

$$Dist(W) = \sum_{k=1}^{K} Dist(w_{ki}, w_{kj}) \qquad i = 1 \ldots n, j = 1 \ldots m \tag{4}$$

The optimal warp path is the wrap path with the minimum distance[18].

3.3 String edit distance

Similar to a DNA sequence encoded with the four letters AGTC that undergoes insertions, deletions substitutions and transpositions, we could encode

all the activities performed on an opportunity as a succession of letters and subsequently for each succession calculate the distance between any other succession using a string edit distance metric. The main idea is to compare two DNA sequences and see how closely they are to each other.

3.3.1 Damerau–Levenshtein string edit distance

Damerau and Levenshtein studied spelling errors and discovered that almost 80% of the spelling errors are at distance one in the metric that carries their names [5]. The metric is a function from an alphabet combination of characters to an integer value [10]. The created metric evaluates how many operations are needed to transform string s1 into string s2.

The distance d(s1,s2) between two strings can be a combination of the following operations:

- insert a character,
- delete a character,
- substitute a character with another from the same alphabet,
- transposition of two adjacent characters.

Although can be a high number of combination of these operations to convert s1 into s2, the metric returns the length of the shortest sequence as the distance between the two strings. For example, in Fig. 2, to transform the text ABACDEAAB into BACDEAEAB we need two operations a delete operation and an insert operation: **A**BACDEAAB $\Longrightarrow$ BACDEAAB $\Longrightarrow$ BACDEA**E**AB. Therefore $\mathsf{d}(\mathsf{ABACDEAAB}, \mathsf{BACDEAEAB}) = 2$.

We used this metric to calculate the distance between the list of activities performed on won deals and the list of activities performed on lost deals in two flavors. First, we did not consider the time elapsed between activities. On the second approach we encoded any week with no activity with a 0. This way we take into account not only the type of activities and their succession, but also the time component that captures periods of inactivity.

4 Results

4.1 The dataset

The data set [15] used for our research represents B2B sales of an ERP software that is sold globally. The data set has 276 deals, 153 lost and 123 won deals in the period 2009-2016 with 19082 activities.

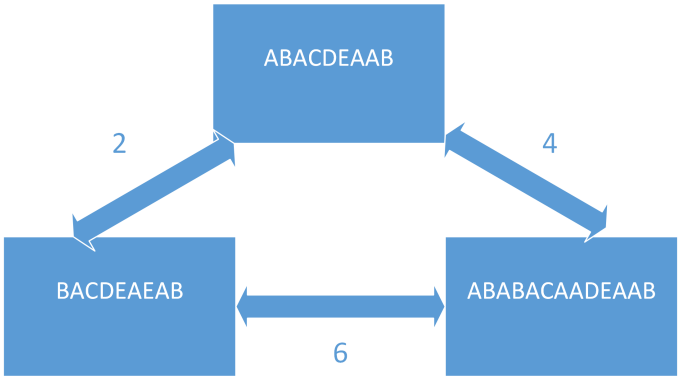


Figure 2: An example of activity succession alignment using Damerau–Levenshtein string edit distance

A summary statistics of the data set is present in Table 1.

Furthermore, the quartiles of the number of days to a closed deal is shown in Table 2.

4.2 Histogram analysis results

We used two approaches to preprocess the opportunities data. The first approach was to extract the minutes between two consecutive notes for each lost and won opportunity.

A second approach for histogram analysis was to use the number of activities the sale representative is performing each week for each opportunity. This approach is meant to reduce the variance due to a too granular look when using the number of minutes between activities. This approach solves also the weekend problem as in general there are no activities in the weekend and national holidays (however, if there are we don't need to treat them specially). Using the week level aggregation could reveal more clear weekly behavioral patterns.

Examining the histogram in Fig. 3 we notice that there are more activities for the lost deals than the won deals (frequency bypasses 5000). However, this is correlated with the fact that there are more lost activities than won activities in the data set. Other than these insights, the histograms show that the wait time between activities is similar for the won deals and lost deals with just a few tiny spikes on the lost deals side. In this context, there are no partic-

Value	Won Deals	Lost Deals
Opportunities	123	153
Activities	12129	6953
Avg(#activities/opportunity)	98	45
Max(#activities/opportunity)	286	153
Min time to a closed deal	4	1
Avg time to a closed deal	47	38
Max time to a closed deal	195	155

Table 1: Data set statistics

Quartile	Won Deals	Lost Deals
0%	4	1
25%	23	23
50%	47	38
75%	98	58
100%	195	155

Table 2: Quartiles of how many days it takes to close a deal

ular idiosyncrasies between the two sets to extract an obvious differentiating behavior. One of the reasons could be that the minutes between activities is a too detailed level of analysis and we might need to zoom out. This prompted us to explore the number of activities per week as well.

The results analysis for the number of activities per week in Fig. 4 shows that in the first week a lot of activity happens for both won and lost deals. However, after the first couple weeks the pattern is slightly different for won and lost deals. The won deals have a constant stream of activities, whereas the lost deals, similar to the minutes between activities histogram, have some spikes.

This prompted us to further examine the patterns of activities for the lost/won deals using dynamic time warping time series analysis and string edit distance to further isolate and measure the possible patterns.

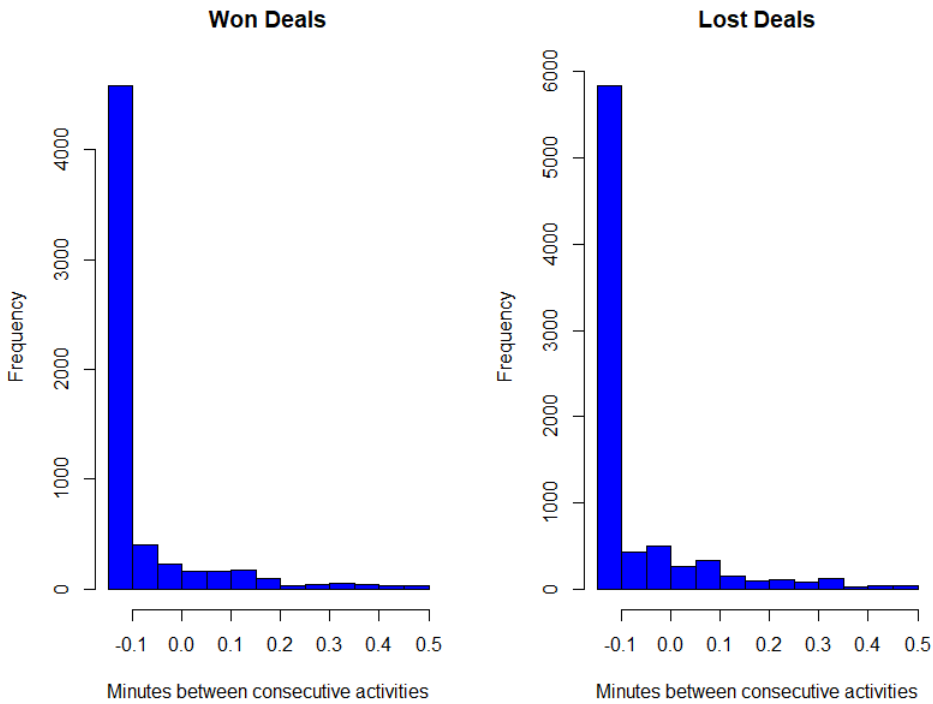


Figure 3: Histogram of minutes between activities for lost/won deals

4.3 Dynamic time warping results

Dynamic time warping is an algorithm used to measure the similarity between two sequences which may vary in speed or time. To prepare the data for DTW exploration we used the same process as above for counting the number of notes per week for each won and lost deal.

We conducted a quantitative analysis by measuring DTW using Euclidean distance and also a variant using standardization. We would have expected the won deals to be close together with a low DTW number and the won versus lost deals to have a higher number. However, as seen in Table 3, the average distance between won deals is 77, the average distance between lost deals is 35 and between lost and won deals is and in between value of 35. This last value we would have expected to be very high which it would prove that we can separate the lost deals from won deals using DTW.

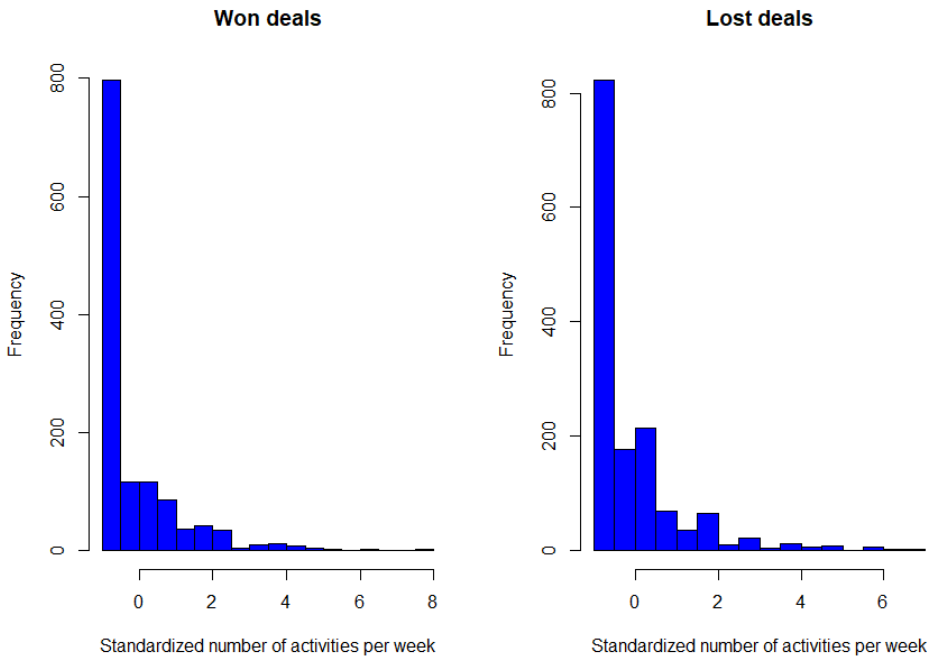


Figure 4: Histogram number of activities per week for won and lost deals

Yet, these findings suggest that our results did not provide convincing evidence towards our hypothesis which means that a pattern cannot be established between the won deals and lost deals. To further prove the new hypothesis, we turned to string edit distance for an additional perspective analysis.

4.4 String edit distance results

To use Damerau–Levenshtein string edit distance we encoded each activity type with a letter from the ASCII code. For our data set we have 28 different activity types. Once we assigned a character to each activity type we create the sequence of activities for all the opportunities lost and won. The process is illustrated in Table 4.

Once all the activity letter sequence was calculated for each opportunity we calculated the distances between won activities, lost activities and between the won and the lost.

Deals	Avg DTW distance	Avg DTW normalized distance
All won deals	77	2.96
All lost deal	35	0.31
All won vs all lost	61	1.74

Table 3: DTW distance between deals

Activity	Encoding	Sequence
Phone call	A	A
Email	B	A**B**
Phone call	A	AB**A**
Customer visit notes	C	ABA**C**
Proposal preparation	D	ABAC**D**
Fax	E	ABACD**E**
Phone call	A	ABACDE**A**
Phone call	A	ABACDEA**A**
Email	B	ABACDEAA**B**

Table 4: Activities encoding sequence

The evidence we were looking for was that the won opportunity activities are clustered together showed by a low distance, the same for lost opportunity activities and in contrast the distance between won opportunity activities and lost opportunity activities is considerably higher. This would mean that we could group the won activities and the lost activities and therefore could establish a pattern for each group of activities.

As we can see from table 5 the average distance between all won deals was 77 and the average distance between all lost deals was 61. These numbers prove that that the succession of activities for the won deals is much more different than for the lost deals. When we calculated the distance between the succession of activities for the won deals against the succession of activities for the lost deals we obtained 42. As can be seen, the numbers match the findings we attained with dynamic time warping: a high distance between the won deals, a low distance for the lost deals and an in-between distance for the won versus lost deals.

The initial approach that we used for string edit distance did not take into account the time element. For example, a phone call followed by an email can

Deals	Avg string edit distance
All won deals	77
All lost deal	61
All won vs all lost	42

Table 5: Damerau–Levenshtein distance between deals

Deals	Avg string edit distance
All won deals	1022
All lost deal	334
All won vs all lost	748

Table 6: Damerau-Levenshtein distance between deals with time series

be done after 1 day or after 1 month. Obviously, it is much better to do it sooner than later. For this reason, we explored a second approach of calculating the string edit distance by encoding weeks with no activities as 0. Therefore a sequence of AB can be A00000B if 5 weeks passed with no activities on the active opportunity. The results of the string edit distance with 0s for weeks with no activity are shown in Table 6.

Although the distances are larger, the results in Table 6 strengthen again the argument that a pattern could not be established. Therefore, contrary to our expectations, our hypothesis that there is a pattern of actions for won deals (and lost deals) does not hold. The general picture emerging from this analysis is that current actions taken cannot predict the future deal outcome.

5 Conclusion and future work

In this paper, we looked into the behavior of the salespeople captured by the successions of activities to determine their influence towards the final outcome. We started with the hypothesis that the pattern of activities for won deals might be different from the patterns of activities for lost deals. However, the results yielded some interesting findings.

The analysis of the succession of activities for lost deals and for the won deals shows that the sales representatives win their deals in very different ways as exhibited by the high distance value for both DTW and Damerau–Levenshtein distance. In contrast, the deals are lost in a more comparable fashion revealed by a low DTW and Damerau–Levenshtein distance.

More surprising however, is the fact that the lost deals patterns are closer to the won deals than the won deals themselves. We can infer from these results that the sales representatives don't behave differently, they behave the same way for the won deals and for the lost deals and their behavior is not correlated with the final outcome i.e. they treat each deal the same putting the same amount of effort to close the deal, not knowing if it will be lost or won which is revealing and encouraging for the sales team.

However, our data set was limited to one B2B CRM instance. Future research will have to clarify whether similar results could be obtained with a different B2B CRM data set. Furthermore, we examined the time series without adding any contextual knowledge (the size of the deal, the period during the quarter, the country of the customer). Adding context might offer additional insights that we plan to address in more detail in a future work.

References

[1] J. S. Armstrong, *Principles of forecasting: a handbook for researchers and practitioners* , Springer Science & Business Media, 2001. ⇒72

[2] C. D. Bodkin, C. P. Schuster, A Preliminary Investigation of the Predilection for Adaptive Behavior and Sales Success, *Proceedings of the 1988 Academy of Marketing Science (AMS) Annual Conference*, 2015, pp. 291–295. ⇒72

[3] E. Bridges, R. Goldsmith, C. F. Hofacker, Attracting and retaining online buyers: comparing B2B and B2C customers, *Advances in electronic marketing* **4772** (2005) 1–27. ⇒71

[4] F. Buttle, *Customer Relationship Management: Concepts and Technology* , Sydney: a Butterworth-Heinemann Title, 2009. ⇒72

[5] F. J. Damerau, A technique for computer detection and correction of spelling errors, *Communications of the ACM* **7,3** (1964) 171–176. ⇒75

[6] M. Gahirwal, Inter time series sales forecasting, *arXiv preprint arXiv:1303.0117* (2013). ⇒73

[7] D. Griffiths , *Head first statistics* , O'Reilly Media, Inc., 2008. ⇒74

[8] K. Jaya, R. Vadlamani, Evolutionary computing applied to customer relationship management: A survey *Engineering Applications of Artificial Intelligence* **56** (2016) 30–59. ⇒71

[9] T. A. Judge, C. J. Thoresen, J. P. Bono, G. K. Patton, The job satisfaction–job performance relationship: A qualitative and quantitative review, *Psychological Bulletin* **127,** 3, (2001), 376–407. ⇒72

[10] V. I. Levenshtein, Binary codes capable of correcting deletions, insertions, and reversals, *Soviet physics doklady* **10,8** (1966) 707–710. ⇒75

[11] N. Liu, T. Choi, C. Hui, S. Ng, S. Ren, Sales forecasting for fashion retailing service industry: a review *Mathematical Problems in Engineering* **2013**. ⇒73

[12] P. Pai, C. Liu, Predicting Vehicle Sales by Sentiment Analysis of Twitter Data and Stock Market Values *IEEE Access* **6** (2018) 57655–57662. ⇒73

[13] R. Palmatier, R. Dant, K. R. Evans, D. Grewal, Factors influencing the effectiveness of relationship marketing: A meta-analysis *Journal of marketing* **70, 4** (2006) 136–153. ⇒72

[14] A. Rainer, P. Thomas, Successful practices in customer relationship management, *Proceedings of the 37th Annual Hawaii International Conference on System Sciences*, Hawaii, USA, 2004. ⇒71

[15] D. Rotovei, B2B ERP sales activities data set, *B2B ERP sales activities data set*. ⇒75

[16] D. Rotovei, V. Negru, *Improving Lost/Won Classification in CRM Systems Using Sentiment Analysis*, *Proc. 19th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, Timisoara, Romania, 2017, pp. 180–187. ⇒71

[17] D. Rotovei, V. Negru, *A methodology for improving complex sales success in CRM Systems*, *INnovations in Intelligent SysTems and Applications (INISTA)*, Gdansk, Poland, 2017, pp. 322–327. ⇒71

[18] S. Salvador, P. Chan, Toward accurate dynamic time warping in linear time and space, *Intelligent Data Analysis* **11,5** (2007) 561–580. ⇒74

[19] H. Terho, A. Eggert, A. Haas, W. Ulaga, How sales strategy translates into performance: The role of salesperson customer orientation and value-based selling *Journal of Industrial Marketing Management* **70, 4** (2006) 136–153. ⇒72

[20] A. S. Tolba, M. Iman, A. M. Hakim, Impact Of Ethical Sales Behavior, Quality and Image on Customer Satisfaction and Loyalty: Evidence From Retail Banking in Egypt, *International Journal of Management and Marketing Research* **8,2** (2015) 1–18. ⇒72

[21] K. Vinod, R. E. Gagandeep, Business to business (b2b) and business to consumer (b2c) management *International Journal of Computers & Technology* **3, 3b** (2012) 447–451. ⇒71

[22] D. Wei, P. Geng, L. Shuaipeng, L. Ying, A prediction study on e-commerce sales based on structure time series model and web search data, *Proceedings of the 126th Chinese Control and Decision Conference (CCDC)*, 2014, pp. 5346–5351. ⇒73

Received: January 3, 2020 • Revised: March 2, 2020

Acta Univ. Sapientiae, Informatica **12,** 1 (2020) 84–101

DOI: 10.2478/ausi-2020-0006

Metric and upper dimension of zero divisor graphs associated to commutative rings

S. PIRZADA
University of Kashmir, Srinagar, India
email:
pirzadasd@kashmiruniversity.ac.in

M. AIJAZ
Department of Mathematics, University of Kashmir, India
email: ahaijaz99@gmail.com

Abstract. Let R be a commutative ring with $Z^*(R)$ as the set of non-zero zero divisors. The zero divisor graph of R, denoted by $\Gamma(R)$, is the graph whose vertex set is $Z^*(R)$, where two distinct vertices x and y are adjacent if and only if $xy = 0$. In this paper, we investigate the metric dimension $\dim(\Gamma(R))$ and upper dimension $\dim^+(\Gamma(R))$ of zero divisor graphs of commutative rings. For zero divisor graphs $\Gamma(R)$ associated to finite commutative rings R with unity $1 \neq 0$, we conjecture that $\dim^+(\Gamma(R)) = \dim(\Gamma(R))$, with one exception that $R \cong \Pi\mathbb{Z}_2^n,\ n \geq 4$. We prove that this conjecture is true for several classes of rings. We also provide combinatorial formulae for computing the metric and upper dimension of zero divisor graphs of certain classes of commutative rings besides giving bounds for the upper dimension of zero divisor graphs of rings.

1 Introduction

Throughout this article, R is assumed to be a commutative ring with unity $1 \neq 0$, unless otherwise stated. Let $Z(R)$ be its set of zero divisors. The zero

Computing Classification System 1998: G.2.2
Mathematics Subject Classification 2010: 13A99, 05C78, 05C12
Key words and phrases: ring, zero divisor, zero divisor graph, metric dimension, upper dimension

divisor graph of R [2], denoted by $\Gamma(R)$, is defined as an undirected graph associated to a commutative ring R having vertex set $V(\Gamma(R)) = Z^*(R) = Z(R) - \{0\}$, where distinct vertices x and y are adjacent if and only if $xy = 0$. The original definition of zero divisor graph was introduced by Beck [6] and in his work he defined $V(\Gamma(R)) = Z(R)$ and two vertices x and y being adjacent if and only if $xy = 0$. This definition of zero divisor graphs was first introduced in [3]. Recently, Kimball and LaGrange [13] generalized the definition to the idempotent divisor graph of a commutative ring. Besides this the zero divisor graph has also been extended to other algebraic structures like semi rings, Abelian groups, vector spaces, modules etc, (e.g., see the articles such as [5, 8, 9, 23] and references therein).

For any set X, let $|X|$ denote the cardinality of X and X^* denote the set of non-zero elements of X. We denote an empty set by φ. An element x in a ring R is called nilpotent if $x^m = 0$ for some positive integer m. A ring R is called reduced if it has no non-zero nilpotent elements. A ring is called local if it has a unique maximal ideal. An element x in a ring R is called a unit if there exists an element y in R such that $xy = 1$, where 1 is a multiplicative identity in R. The set of all units of a ring R is denoted by $U(R)$. We denote a ring of integers by $\mathbb{Z}$, a ring of integer modulo n by $\mathbb{Z}_n$ and a finite field with q elements by $\mathbb{F}_q$.

This article continues the investigation of zero divisor graphs that have same metric and upper dimension that was started in [17]. Section 2 reviews basic definitions and known results concerning the metric and upper dimension of zero divisor graphs of rings, as well as the results obtained for graphs in general. The rest of the paper focuses on zero-divisor graphs of commutative rings. In Section 3, we show that if either $\Gamma(R)$ (or $\overline{\Gamma}(R)$) is a regular graph, then $\dim(\Gamma(R)) = \dim^+(\Gamma(R))$. We also characterize certain families of local and reduced artinian rings and show that their zero divisor graphs have same values for these two parameters. Further, we compute the metric and upper dimension formulae for certain other classes of rings and show that the two values are equal. We obtain a lower bound for the upper dimension of zero divisor graph of a finite Boolean ring.

2 Preliminaries and terminology

A graph G with vertex set $V(G) \neq \emptyset$ and edge set $E(G)$ of unordered pairs of distinct vertices is called a simple graph. A graph G is connected if and only if there is path between any two pair of vertices x and y of G. In a graph G,

the distance between two vertices x and y is the length of the shortest path between x and y. A subset $\mathfrak{B}$ of $V(G)$ is said to resolve a pair of vertices $\{u, v\} \subset V(G)$, if there exists some $b \in \mathfrak{B}$ such that $d(u, b) \neq d(v, b)$ or equivalently if the metric representations of distinct vertices are distinct, where the metric representation for a vertex $v \in V(G)$ with respect to an ordered set $\mathfrak{B} = \{b_1, b_2, \ldots, b_k\}$ of vertices of G is an ordered k-tuple defined as $r(v \mid \mathfrak{B}) = (d(v, b_1), d(v, b_2), \cdots, d(v, b_k))$. If $\mathfrak{B}$ resolves all the vertices of G, we say $\mathfrak{B}$ is a resolving set of G and $\mathfrak{B}$ is said to be a minimal resolving set if no proper subset of $\mathfrak{B}$ resolves all vertices of G. A minimal resolving set $\mathfrak{B}$ with least number of vertices is called a *metric basis* of G and the cardinality of metric basis is called the *metric dimension* of G, denoted by $\dim(G)$. Also, a minimal resolving set containing the maximum number of vertices is called an *upper basis* of G and the cardinality of the upper basis is called the *upper dimension* of G, denoted by $\dim^+(G)$.

The concept of finding the metric dimension of a graph first appeared in 1970's introduced by Slater [24] and independently by Harary and Melter [11] and the concept of upper dimension of graphs was introduced by Chartrand et al. [7], where they defined the upper dimension to be the order of the minimal resolving set that has the maximum cardinality. Recently, these concepts of metric and upper dimension of graphs were extended to zero divisor graphs of rings, see [17, 20, 21].

For other definitions, terminology and notations of ring theory, we refer to [4, 12] and for graph theory, we refer to [14].

Theorem 1 [Theorem 3.5, [10]] *For every pair* a, b *of integers with* $2 \leq a \leq b$*, there exists a connected graph* G *with* $\dim(G) = a$ *and* $\dim^+(G) = b$

Let $u \leftrightarrow v$ denote that u is adjacent to v and $u \nleftrightarrow v$ denote that u is not adjacent to v.

Distance similarity. In a connected graph G, two vertices u and v are said to be *distance similar* if for any vertex $x \in V(G) - \{u, v\}$, $d(u, x) = d(v, x)$. The relation of distance similarity is an equivalence relation. Therefore, it partitions the vertex set of a graph into equivalence classes known as *distance similar equivalence classes.*

Example 2 Let G be a book graph (see Figure 1) with 5 pages with corners of the pages as $p_1, p_2, \ldots, p_5$, q_1, q_2, $\ldots$, q_5, p, q and the adjacencies as follows: $p \leftrightarrow p_i$, $q \leftrightarrow q_i$, for all $1 \leq i \leq 5$, $p \leftrightarrow q$ and $p_i \leftrightarrow q_j$, if and only if, $i = j$, $1 \leq i, j \leq 5$, elsewhere non-adjacencies. Then we have $d(p_i, p_j) = 2 = d(q_i, q_j)$, $d(p_i, q_j) = 3$, whenever $i \neq j$. Choose $\mathfrak{B} = \{p_1, p_2, q_3, q_4\}$ and

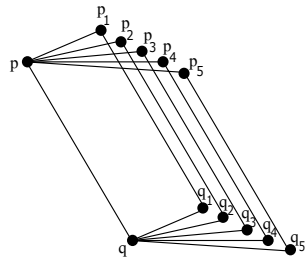


Figure 1: Book graph with 5 pages.

$\mathfrak{B}^* = \{p_1, q_1, q_2, q_3, q_4\}$. Then we see that $\mathfrak{B}$ is a minimal resolving set of minimum order and $\mathfrak{B}^*$ is a minimal resolving set of maximum order and so $\dim(G) = 4$, whereas, $\dim^+(G) = 5$.

Theorem 3 *(i)* [Theorem A, [7]] *Let* G *be a connected graph on* n *vertices. Then* $\dim(G) = 1$ *if and only if* $G \cong P_n$*, where* P_n *denotes the path on* n *vertices.*
(ii) [Lemma 2.3, [18]] *For a connected graph* G *of order* $n \geq 1$*,* $\dim^+(G) = 1$ *if and only if* $G \cong P_2$ *or* P_3 *and for* $n \geq 4$ $\dim^+(P_n) = 2$*, where* P_n *denotes the path on* n *vertices.*

Theorem 4 [Theorem 2.3, [21] and Theorem 2.5, [15]] *Let* G *be a connected graph of order* n*. Then* $\dim(G) = \dim^+(G) = n - 1$ *if and only if* $G \cong K_n$*.*

For integers $k \geq 2$ and $n_1, n_2, \ldots, n_k$ with $2 \leq n_1 \leq n_2 \leq \cdots \leq n_k$, let $G = K_{n_1,n_2,\ldots,n_k}$ be a complete r-partite graph with $V_1, V_2, \ldots, V_k$ as the partite sets. Let u, v be vertices in some partite set, then $d(u, v) = 2$ and for any other vertex $w \in V(G)$, we have either $d(u, w) = 2$ if and only if w is in the same partite set or otherwise $d(u, w) = 1$. Thus, these partite sets partition the vertex set of G into distance similar equivalence classes and therefore we have the following theorem.

Theorem 5 *(i) For integers* $k \geq 2$ *and* $n_1, n_2, \ldots, n_k$ *with* $2 \leq n_1 \leq n_2 \leq \cdots \leq n_k$ *and* $n_1 + n_2 + \cdots + n_k = n$*,*

$$\dim(K_{n_1,n_2,\ldots,n_k}) = \dim^+(K_{n_1,n_2,\ldots,n_k}) = n - k$$

(ii) For all positive integers $n \geq 2$*,* $\dim(K_{1,n}) = \dim^+(K_{1,n}) = n - 1$*.*

The Cartesian product of two graphs G_1 and G_2, denoted by $G = G_1 \times G_2$, is the graph whose vertex set is $V = V(G_1) \times V(G_2)$ and for any two vertices $w_1 = (u_1, v_1)$ and $w_2 = (u_2, v_2)$ in V with $u_1, u_2 \in V(G_1)$ and $v_1, v_2 \in V(G_2)$, there is an edge $w_1w_2 \in E(G)$ if and only if

(a) either $u_1 = u_2$ and $v_1v_2 \in E(G_2)$ or (b) $v_1 = v_2$ and $u_1u_2 \in E(G_1)$,

Theorem 6 *(i)* [Theorem 3.2, [18]] *For* $n \geq 3$,

$$\dim^+(K_{1,n} \times K_2) = \dim^+(K_{1,n}) + 1 = n.$$

(ii) [Corollary 3.3, [18]] *For* $n \geq 5$, $\dim(K_{1,n} \times K_2) = \dim^+(K_{1,n}) = n - 1$.

Theorem 7 [Theorem 2.8, [17]] *Let* R *be a commutative ring with unity. Then* $\dim^+(\Gamma(R))$ *is finite if and only if* R *is finite (and not a domain).*

Recall that the characteristic of a ring R is a smallest positive integer k such that for every $r \in R$, we have $kr = 0$, and if no such integer exists then the ring R is said to have infinite characteristic.

Theorem 8 [Theorem 3.2, [17]] *Let* R *be a finite commutative ring that is not a field such that* R *has odd characteristic. Then* $\dim^+(\Gamma(R)) = \dim(\Gamma(R))$.

Theorem 9 [Theorem 3.3, [17]] *Let* S *be a finite commutative ring of order* $2k$, *where* k *is an odd integer. Then* $\dim^+(\Gamma(S)) = \dim(\Gamma(S))$.

Theorem 10 [Theorem 6-7, [1]] *Let* R *be a finite commutative ring. If all vertices of* $\Gamma(R)$ *(or* $\overline{\Gamma}(R)$*) have the same degrees, then either* $Z(R)^2 = \{0\}$ *or* $R \cong \mathbb{F} \times \mathbb{F}$, *for some finite field* $\mathbb{F}$.

Theorem 11 [Lemma 2.2, [15]] *If* G *is a connected graph and* $D \subseteq V(G)$ *is a subset of the distance similar vertices with* $|D| \geq 2$, *then every resolving set of* G *contains exactly* $|D| - 1$ *vertices of* D.

3 Main results

Theorem 12 *Let* R *be a commutative ring with unity. Then* $\dim(\Gamma(R)) = \dim^+(\Gamma(R)) = 1$ *if and only if* R *is one of the following rings.*

(i) $\dfrac{\mathbb{Z}_3[x]}{(x^2)}$, $\mathbb{Z}_2 \times \mathbb{Z}_2$, $\mathbb{Z}_9$.

(ii) $\mathbb{Z}_6$, $\mathbb{Z}_8$, $\dfrac{\mathbb{Z}_2[x]}{(x^3)}$, $\dfrac{\mathbb{Z}_4[x]}{(2x, x^2 - 2)}$.

Proof. These are the only rings whose zero divisor graph is isomorphic to (i) P_2 or (ii) P_3 and the only connected graphs whose metric and upper dimension is 1 is either P_2 or P_3. Hence the result follows. □

Example 13 *Let* R *be a commutative ring with unity. If* $R \cong \mathbb{Z}_2 \times \mathbb{Z}_4$, $\mathbb{Z}_2 \times \frac{\mathbb{Z}_2[x]}{(x^2)}$, $\mathbb{Z}_2 \times \mathbb{F}_4$, $\frac{\mathbb{Z}_2[x,y]}{(x,y)^2}$, $\frac{\mathbb{Z}_4[x,y]}{(2,x)^2}$, $\frac{\mathbb{F}_4[x]}{(x^2)}$, $\frac{\mathbb{Z}_4[x]}{(x^2+x+1)}$, $\mathbb{Z}_2 \times \mathbb{Z}_2 \times \mathbb{Z}_2$, $\mathbb{Z}_3 \times \mathbb{Z}_3$, *then* $\dim(\Gamma(R)) = \dim^+(\Gamma(R)) = 2$.

If $R \cong \mathbb{Z}_2 \times \mathbb{Z}_4$, then the only basis sets are $\{(0,1),(0,2)\}$, $\{(0,3),(0,2)\}$, $\{(0,1),(1,2)\}$, $\{(0,3),(1,2)\}$ or $\{(0,1),(0,3)\}$. A similar list can be constructed for $\mathbb{Z}_2 \times \frac{\mathbb{Z}_2[x]}{(x^2)}$ because $\Gamma\left(\mathbb{Z}_2 \times \frac{\mathbb{Z}_2[x]}{(x^2)}\right) \cong \Gamma(\mathbb{Z}_2 \times \mathbb{Z}_4)$. If $R \cong \mathbb{Z}_2 \times \mathbb{Z}_2 \times \mathbb{Z}_2$ then any two elements of $S_1 = \{(1,0,0),(0,1,0),(0,0,1)\}$ or $S_2 = \{(1, 1, 0), (1, 0, 1), (0, 1, 1)\}$ forms a basis. If $R \cong \frac{\mathbb{Z}_2[x,y]}{(x,y)^2}$ or $\frac{\mathbb{Z}_4[x,y]}{(2,x)^2}$, or $\frac{\mathbb{F}_4[x]}{(x^2)}$, or $\frac{\mathbb{Z}_4[x]}{(x^2+x+1)}$ then $\Gamma(R) \cong K_3$ and if $R \cong \mathbb{Z}_2 \times \mathbb{F}_4$, then $\Gamma(R) \cong K_{1,3}$.

Theorem 14 [Theorem 2.8, [18]] *Let* G *be a finite connected graph such that every* $v \in V(G)$ *is distance similar to some vertex* $u \neq v$. *Then* $\dim^+(G) = \dim(G)$.

By $G \vee H$, we shall denote the join of two graphs G and H.

Theorem 15 *Let* R *be a commutative ring with unity* $1 \neq 0$, *(not a domain).*

(1) If $|R| = p^2$, *where* p *is prime, then* $\dim(\Gamma(R)) = \dim^+(\Gamma(R))$.
(2) If R *is local with order* p^3, *then* $\dim(\Gamma(R)) = \dim^+(\Gamma(R)$.

Proof. (1). If R is local, then either R is isomorphic to $\mathbb{Z}_{p^2}$ or $\frac{\mathbb{Z}_p[x]}{(x^2)}$ and in either case $\Gamma(R)$ is complete with order $p-1$. If R is reduced, then R is isomorphic to $\mathbb{Z}_p \times \mathbb{Z}_p$, so $\Gamma(R)$ is complete bipartite. Therefore, the result follows.

(2). If R is a local ring of order p^3, then R is isomorphic to one of the following rings; $\frac{\mathbb{F}_p[x,y]}{(x,y)^2}$, $\frac{\mathbb{F}_p[x]}{(x^3)}$, $\frac{\mathbb{Z}_{p^2}[x]}{(px,x^2)}$, or $\frac{\mathbb{Z}_{p^2}[x]}{(px,x^2-\bar{s}p)}$, where $\bar{s}$ is a non-square element in $\mathbb{Z}_p$. If $R \cong \frac{\mathbb{F}_p[x,y]}{(x,y)^2}$, then $Z^*(R) = \{ux\} \cup \{uy\} \cup \{ux + u'y\}$, where $u, u' \in \mathbb{F}_p - \{0\}$. Thus, $\left|\Gamma\left(\frac{\mathbb{F}_p[x,y]}{(x,y)^2}\right)\right| = p^2 - 1$, and for all $u, v \in Z^*(R)$,

we have $uv = 0$. Therefore, $\dim(\Gamma(R)) = \dim^+(\Gamma(R) = p^2 - 2$. Also, if $R \cong \frac{\mathbb{Z}_{p^2}[x]}{(px, x^2)}$, then $Z^*(R) = \{up\} \cup \{ux\}$, where $u \in \mathbb{Z}_p - \{0\}$, so $\Gamma(R) \cong K_{p^2-1}$. Next, if $R \cong \frac{\mathbb{F}_p[x]}{(x^3)}$, then $Z^*(R)$ can be partitioned into two subsets; $Z_1 = \{ux^2 | u \in \mathbb{F}_p - \{0\}\}$ and $Z_2 = \{ax + bx^2 | a \in \mathbb{F}_p - \{0\}, b \in \mathbb{Z}_p\}$. Then Z_1 induces a clique on $p-1$ vertices and Z_2 is an independent subset. Also, for all $z_1 \in Z_1$ and $z_2 \in Z_2$, we have $z_1 z_2 = 0$. Thus, $\Gamma\left(\frac{\mathbb{F}_p[x]}{(x^3)}\right) \cong K_{p-1} \vee T_{p^2-p}$. Let $\{u_1, u_2, \cdots, u_{p-1}\}$ be the set of units of $\mathbb{F}_p$ and let $z_2, z_2' \in Z_2$, then for $1 \leq i \neq j \leq n-1$, we have $d(u_i x, z_2) = d(u_j x, z_2) = 1$ and $d(u_i x, z_1) = d(u_i x, z_2') = 1$, but however $d(z_2, z_2') = 2$, therefore, the sets Z_1 and Z_2 partition the vertex set of $\Gamma\left(\frac{\mathbb{F}_p[x]}{(x^3)}\right)$ into distance similar equivalence classes. Therefore, the result follows by Theorem 14. Finally, if $R \cong \frac{\mathbb{Z}_{p^2}[x]}{(px, x^2 - \bar{s}p)}$, where $\bar{s}$ is a non-square element in $\mathbb{Z}_p$, then we partition the vertex set of $\Gamma\left(\frac{\mathbb{Z}_{p^2}[x]}{(px, x^2 - \bar{s}p)}\right)$ into the subsets $S_1 = \{up \mid u \in \mathbb{Z}_p - \{0\}\}$ and $S_2 = \{ux\} \cup \{up + u'x \mid u, u' \in \mathbb{Z}_p - \{0\}\}$. Then for all $s_1, s_1' \in S_1$ and $s_2, s_2' \in S_2$, we have $s_1 s_1' = 0$, $s_1 s_2 = 0$ and $s_2 s_2' \neq 0$. In fact, the collection $\{S_1, S_2\}$ partitions the vertex set of $\Gamma\left(\frac{\mathbb{Z}_{p^2}[x]}{(px, x^2 - \bar{s}p)}\right)$ into distance similar classes, so the result follows by Theorem 14. □

Corollary 16 *The metric and the upper dimension of zero divisor graph of* $\frac{\mathbb{F}_p[x, y]}{(x, y)^2}$ *and* $\frac{\mathbb{Z}_{p^2}[x]}{(px, x^2)}$ *are equal to* $p^2 - 2$ *and for the zero divisor graph of* $\frac{\mathbb{F}_p[x]}{(x^3)}$ *and* $\frac{\mathbb{Z}_{p^2}[x]}{(px, x^2 - \bar{s}p)}$, *these two values are both equal to* $p^2 - 3$.

Proof. This can be obtained using Theorem 11 in part (2) of Theorem 15 along with Theorem 14. □

In the following theorem, the metric and upper dimension of a class of local rings is characterized.

Theorem 17 [Theorem 2.9 [16]] *Let* R *be a ring (local) isomorphic to* $\mathbb{Z}_{p^n}$, *then* $\dim(\Gamma(R)) = \dim^+(\Gamma(R)) = p^{n-1} - n$.

Theorem 18 *Let* R *be a finite commutative ring. If either* $\Gamma(R)$ *(or* $\overline{\Gamma}(R)$*) is a regular graph, then either* $\dim(\Gamma(R)) = \dim^+(\Gamma(R)) = Z^*(R) - 1$ *or* $\dim(\Gamma(R)) = \dim^+(\Gamma(R)) = Z^*(R) - 2$.

Proof. If either of the graphs $\Gamma(R)$ or its complement is regular, then by Theorem 10, either $Z(R)^2 = 0$ or there is a field $\mathbb{F}$ such that $R \cong \mathbb{F} \times \mathbb{F}$. Therefore, either $\Gamma(R)$ is complete or a complete bipartite graph. Hence the result follows by Theorem 5. □

Resolving sets for zero-divisor graphs have previously been studied in [17, 19] and [21]. In these articles, it was noted that distance similarity was a key factor in determining resolving sets. The following results illustrate this connection between concepts.

Theorem 19 [Theorem 2.1 [19]] *Let G be a connected graph. Suppose G is partitioned into k distinct distance similar classes $V_1, V_2, \ldots, V_k$ (that is, $x, y \in V_i$ if and only if $d(x, a) = d(y, a)$ for all $a \in V(G) - \{x, y\}$).*

(i) Any resolving set W for G contains all but at most one vertex from each V_i.

(ii) Each V_i induces a complete subgraph or a graph with no edges.

(iii) $\dim(G) \geq |V(G)| - k$.

(iv) There exists a minimal resolving set W for G such that if $|V_i| > 1$, at most $|V_i| - 1$ vertices of v_i are elements of W.

(v) If m is the number of distance similar classes that consist of a single vertex, then $|V(G)| - k \leq \dim(G) \leq |V(G)| - k + m$.

Theorem 20 *Let R be a reduced Artinian ring with unity (not a domain) containing no factor isomorphic to $\mathbb{Z}_2$. Then $\dim^+(\Gamma(R)) = \dim(\Gamma(R))$.*

Proof. It is a well known fact that every reduced Artinian ring is a direct product of fields. Therefore, we can write $R \cong R_1 \times R_2 \times \cdots \times R_m$, for some positive integer m, where each R_i, $1 \leq i \leq m$ is a field.

Therefore, $V(\Gamma(R)) = \{(r_1, r_2, \ldots, r_m)\colon r_i \in R_i$, with $r_i \neq 0$ for some i and $r_j = 0$ for some j, $1 \leq i, j \leq m\}$ and two vertices $x = (x_1, x_2, \ldots, x_m)$ and $y = (y_1, y_2, \ldots, y_m)$ are adjacent if and only if either $x_i = 0$ or $y_i = 0$. Assume that $|R_i| > 2$ for each i and choose $x_i \in R_i^*$. Consider the set $E = \{(x_1, 0, \ldots, 0), (x_1, x_2, 0, \ldots, 0), \ldots, (x_1, x_2, \ldots, x_{m-1}, 0)\}$. Then clearly, no two vertices are adjacent in E. Now, for $1 \leq i \leq m - 1$, let E_i denote the collection of those vertices of $\Gamma(R)$ having exactly i non-zero coordinates. In particular, let

$$E_1 = \{(x_1, 0, \ldots, 0), (0, x_2, 0, \ldots, 0), \ldots, (0, 0, \ldots, 0, x_m)\},$$
$$E_2 = \{(x_1, x_2, 0, \ldots, 0), (x_1, 0, x_3, 0, \ldots, 0), \ldots, (x_1, 0, \ldots, 0, x_m), (0, x_2, x_3, 0, \ldots, 0), (0, x_2, 0, x_4, 0 \ldots, 0), \ldots, (0, x_2, 0, \ldots, x_m), (0, \cdots, 0, x_{m-1}, x_m)\},$$
$$\vdots$$
$$E_{m-1} = (x_1, \ldots, x_{m-1}, 0), (x_1, \ldots, x_{m-2}, 0, x_{m-1}), \ldots, (0, x_2, \ldots, x_m).$$

It is easy to see that the collection $E_1, E_2, \cdots, E_{m-1}$ partitions the vertex set of $\Gamma(R)$ and for $x_i \in R_i^*$, each E_i, $1 \leq i \leq m-1$, defines a distance similar equivalence class. Also, as $|R_i| > 2$ for each i, therefore each E_i has at least two vertices. Therefore, the result follows by Theorem 14. □

Remark 21 $\dim^+(\mathbb{Z}_2 \times \mathbb{Z}_2) = \dim(\mathbb{Z}_2 \times \mathbb{Z}_2) = 1$, $\dim^+(\mathbb{Z}_2 \times \mathbb{Z}_2 \times \mathbb{Z}_2) = \dim(\mathbb{Z}_2 \times \mathbb{Z}_2 \times \mathbb{Z}_2) = 2$. If $R \cong \mathbb{Z}_2 \times \mathbb{Z}_2 \times \mathbb{Z}_2 \times \mathbb{Z}_2$, then $\dim(\Gamma(R)) = 3$, with $\mathfrak{B}$ ={(1,1,1,0), (1,1,0,1), (1, 0, 1, 1)} an example of a minimal resolving set, whereas with $\mathfrak{B}'$ ={(1, 0, 0, 0), (0, 1, 0, 0), (0, 0, 1, 0), (0, 0, 0, 1)} as an example of a minimal resolving set, we have $\dim^+(\Gamma(R))$= 4. Notice that a vertex z of $\Pi\mathbb{Z}_2^n$ is a pendent vertex if and only if z has exactly one zero coordinate and for $n = 4$, any three (i.e $n-1$) pendent vertices of $\Pi\mathbb{Z}_2^4$ form a metric basis. However, the same does not follow if $n \geq 5$ as can be seen in Theorem 28 to the end of this section.

Theorem 22 *Let R_1 be a finite commutative ring with unity and R_2 be an integral domain.*

(i) If $R_1 \cong \mathbb{F}_1 \times \mathbb{F}_2 \times \cdots \times \mathbb{F}_k$, where each $\mathbb{F}_i$ is a field and $\mathbb{F}_i \neq \mathbb{Z}_2$ for each i, $1 \leq i \leq k$ and if $R_2 \neq \mathbb{Z}_2$, then $\dim^+(\Gamma(R_1 \times R_2)) = \dim(\Gamma(R_1 \times R_2))$.
(ii) If $R_1 \cong \mathbb{Z}_{p^k}$, where $k \geq 2$,, then $\dim^+(\Gamma(R_1 \times R_2)) = \dim(\Gamma(R_1 \times R_2))$.
(iii) If R_1 is a local ring other than $\mathbb{Z}_{p^k}$, such that $\Gamma(R_1)$ is a complete graph, then $\dim^+(\Gamma(R_1 \times R_2)) = \dim(\Gamma(R_1 \times R_2))$

Proof. (i) Let $R_1 \times R_2 \cong S_1 \times S_2 \times \ldots S_m$, where $|S_i| \neq 2$ for each i, $1 \leq i \leq m$. Then consider the following partition of $V(\Gamma(R_1 \times R_2)$.

$$A_1^{(1)}, A_1^{(2)}, \ldots, A_1^{(m)}, A_2^{(1,2)}, A_2^{(1,3)}, \ldots, A_2^{(1,m)}, A_2^{(2,3)}, A_2^{(2,4)}, \ldots, A_2^{(2,m)},$$
$$\ldots, A_2^{(m-1,m)}, \ldots, A_{m-1}^{(1,2,\ldots,m-1)}, A_{m-1}^{(1,2,\ldots,m-2,m)}, \ldots, A_{m-1}^{(2,3,\ldots,m)},$$

where $A_i^{(s)}$ denotes the subset of $Z^*(R_1 \times R_2)$ having i non-zero coordinates and the non-zero positions are given by the string (s). For example, let $s_i \in S_i^*$, then $A_1^{(2)} = \{(0, s_2, 0. \ldots, 0)\}$, $A_3^{(124)} = \{(s_1, s_2, 0, s_4, 0, \ldots, 0)\}$, etc. The above partition of $V(\Gamma(R_1 \times R_2))$ is obtained by an equivalence relation " $\sim$ " defined in the following way: let $S = (s_1, s_2, \ldots, s_m)$ and $S' = (s'_1, s'_2, \ldots, s'_m)$, then $S \sim S'$ if and only if whenever $s_i = 0$, then $s'_i = 0$. Also, the number of equivalence classes is equal to $2^m - 2$. As $|S_i| > 2$ for each $i, 1 \leq i \leq m$, we have each $A_i^{(s)}$ induces a subgraph of order at least 2 and size 0 and it is not difficult to see that each $A_i^{(s)}$ is a distance similar equivalence class. Therefore, the result follows by Theorem 14 $\big($in fact by Theorem 11, since every basis misses exactly one vertex from each equivalence class, we have $\dim^+(\Gamma(R_1 \times R_2)) = \dim(\Gamma(R_1 \times R_2)) = |Z^*(R_1 \times R_2)| - (2^m - 2)\big)$.

(ii) First assume that $R_1 \cong \mathbb{Z}_4$. Then $R_1 \times R_2 \cong \mathbb{Z}_4 \times \mathbb{F}$, where $\mathbb{F} = \{f_0, f_1, \ldots, f_t\}$ is an integral domain and f_0 is the zero element of $\mathbb{F}$. If $\mathbb{F} = \mathbb{Z}_2$, then $R_1 \times R_2 \cong \mathbb{Z}_4 \times \mathbb{Z}_2$ and the result is true in this case. So assume $|\mathbb{F}| > 2$. Consider the vertex set $Z^*(R_1 \times R_2)$ of $\Gamma(R_1 \times R_2)$ and let $A = \{1, 3\} \times \{0\} = \{(1, 0), (3, 0)\}$, $B = \{0\} \times \{f_1, f_2, \ldots, f_t\}$, $C = \{(2, 0)\}$ and $D = \{2\} \times \{f_1, f_2, \ldots, f_t\}$, where $t \geq 2$. Then the sets A, B, C and D partition the vertex set of $\Gamma(\mathbb{Z}_4 \times \mathbb{F})$ into the distance similar equivalence classes with $|A| > 1, |B| > 1, |D| > 1$ and $|C| = 1$. Thus, every basis contains all elements of A, B and D but one element from each set, by Theorem 11. Without loss of generality, let $\mathfrak{B} = \{(1, 0)\} \cup \big(B - \{(0, f_1)\}\big) \cup \big(D - \{(2, f_1)\}\big)$. Then $r((3, 0)|\mathfrak{B}) = (2, 1, \ldots, 1, 2, \ldots, 2)$, $r((0, f_1)|\mathfrak{B}) = (1, 2, \ldots, 2)$, $r((2, f_1)|\mathfrak{B}) = (2, 2, \ldots, 2)$ and $r((2, 0)|\mathfrak{B}) = (2, 1, \ldots, 1)$. Therefore, $\mathfrak{B}$ is the basis. Consequently, the only element of C does not belong to any basis.

Hence, $\dim^+(\Gamma(\mathbb{Z}_4 \times \mathbb{F})) = \dim(\Gamma(\mathbb{Z}_4 \times \mathbb{F})) = 2|\mathbb{F}| - 3$. Now, assume that $R_1 \cong \mathbb{Z}_{p^n}$, where $p^n \neq 4$ and R_2 is a domain. We partition the vertex set of $\Gamma(\mathbb{Z}_{p^n})$ into $n - 1$ disjoint subsets of the form $V_1, V_2, \ldots, V_{n-1}$, where $V_i = \{k_i p^i : p \nmid k_i\}$, $1 \leq i \leq n-1$. We see that $|V_i| = (p-1)p^{n-i-1}$, $1 \leq i \leq n-1$ and that $|\Gamma(\mathbb{Z}_{p^n})| = p^{n-1} - 1$. In fact the sets $V_1, V_2, \ldots, V_{n-1}$ gives the partition of $V(\Gamma(\mathbb{Z}_{p^n}))$ into distance similar equivalence classes of cardinality at least 2 except for the case that $|V_{n-1}| = 1$, when $p = 2$.

Define the sets $A = U(R_1) \times \{0\}$, $B = \{0\} \times R_2^*$, $C_i = V_i \times \{0\}$ and $D_i = V_i \times R_2^*$, where $1 \leq i \leq n - 1$. The collection $\mathcal{P} = \{A, B, C_1, C_2, \ldots, C_{n-1}, D_1, D_2, \ldots, D_{n-1}\}$ gives the partition of vertex set of $\Gamma(R_1 \times R_2)$ into distance similar equivalence classes. Notice that $|B| = 1$ if and only if $R_2 \cong \mathbb{Z}_2$, $|C_i| = 1$ if and only if $p = 2$ and $i = n - 1$, and $|D_i| = 1$ if and only if $p = 2$, $i = n - 1$ and $R_2 \cong \mathbb{Z}_2$. So first assume that $p > 2$. Then if $R_2 \ncong \mathbb{Z}_2$, then the collection $\mathcal{P}$

gives the partition in which each set has cardinality at least 2. Therefore, the result follows by Theorem 14. Now, if $p > 2$ and $R_2 \cong \mathbb{Z}_2$, then $|R_1 \times R_2| = 2k$, where k is an odd integer. Therefore, the result follows by Theorem 9.

Finally, assume that $R_1 \cong \mathbb{Z}_{2^n}$, $n \geq 3$. If $R_2 \ncong \mathbb{Z}_2$, then each set in the collection $\mathcal{P}$ has at least 2 elements except C_{n-1}. Without loss of generality, by Theorem 11, we construct the set $\mathfrak{B}'$ which takes all elements but one from each element of $\mathcal{P} - C_{n-1}$. Now, it can be easily seen that the set $\mathfrak{B}'$ gives distinct representations to each vertex of $\Gamma(R_1 \times R_2)$. Therefore, there is a basis (which is both a metric as well as an upper basis) which does not contain the only element of C_{n-1}. Hence, the result follows by Theorem 14 and 11. Lastly, if $R_2 \cong \mathbb{Z}_2$, then the sets in the collection $\mathcal{P}$ that have only one element are B, C_{n-1} and D_{n-1}. Utilizing Theorem 11, we construct $\mathfrak{B}''$ by taking all elements but one from each element of $\mathcal{P} - \{B, C_{n-1}, D_{n-1}\}$. The set $\mathfrak{B}''$ so constructed gives distinct representations to all the vertices of $\Gamma(R_1 \times R_2)$. Hence, $\mathfrak{B}''$ is a resolving set and so $\dim^+(\Gamma(\mathbb{Z}_{2^n} \times \mathbb{Z}_2)) = \dim(\Gamma(\mathbb{Z}_{2^n} \times \mathbb{Z}_2))$ by Theorem 14 and 11.

(iii) Since R_1 is local finite commutative ring with unity, therefore $|R_1| = p^n$ for some prime p and a positive integer n. Let $Z^*(R_1) = \{r_1, r_2, \ldots, r_t\}$ be the set of all non-zero zero divisors of R_1 and $U(R_1)$ be the set of units of R_1. We partition the vertex set of $\Gamma(R_1 \times R_2)$ as follows:

$$X = Z^*(R_1) \times \{0\},\ Y = \{0\} \times R_2^*$$

$$Z = U(R_1) \times \{0\},\ X_i = \{r_i\} \times R_2^*,\ 1 \leq i \leq t.$$

If $R_1 \cong \dfrac{\mathbb{Z}_2[x]}{(x^2)}$, then the proof follows similarly as in the case when $R_1 \cong \mathbb{Z}_4$. Hence, we assume that $R_1 \ncong \dfrac{\mathbb{Z}_2[x]}{(x^2)}$ for the rest of the proof. Assume that $\Gamma(R_1)$ is a complete graph, therefore the set X induces a clique. Now, first consider the case, when $R_2 \ncong \mathbb{Z}_2$. In this case, $|X| > 1$, $|Y| > 1$, $|Z| > 1$ and $|X_i| > 1$ for each i, $1 \leq i \leq t$ and each of the sets X, Y, Z and X_i, $1 \leq i \leq t$ defines a distance similar equivalence class. Therefore, the result follows by Theorem 14. Now, let $R_2 \cong \mathbb{Z}_2$ and $|R_1| = p^n$. If $p > 2$, then $|R_1 \times R_2| = 2k$, where k is an odd integer and therefore the result follows by Theorem 9. Finally, let $|R_1| = 2^n$ and $R_2 = \mathbb{Z}_2$. In this case, we partition the vertex set of $\Gamma(R_1 \times R_2)$ into the sets,

$$X = Z^*(R_1) \times \{0\},\ Y = \{0\} \times R_2^* = \{(0,1)\}$$

$$Z = U(R_1) \times \{0\},\ \text{and}\ \widehat{Z} = Z^*(R_1) \times R_2^* = Z^*(R_1) \times \{1\}.$$

Notice that the set X induces a clique and each of the sets $Y, Z, \widehat{Z}$ is independent. Each $x \in X$ is adjacent to $(0,1)$ and $\widehat{z}$ for all $\widehat{z} \in \widehat{Z}$. The only element of Y i.e., $(0,1)$ is also adjacent to each $z \in Z$ and each element of Z is a pendent vertex. These are the only adjacencies in $\Gamma(R_1 \times \mathbb{Z}_2)$, where R_1 is local other that $\mathbb{Z}_{p^n}$ and not a domain.

The collection $\mathcal{P} = \{X, Y, Z, \widehat{Z}\}$ is the partition of $V(\Gamma(R_1 \times R_2))$ into distance similar equivalence classes. Without loss of generality, using Theorem 11, we form a set $\mathfrak{B}''' = \big(X - \{x\}\big) \cup \big(Z - \{z\}\big) \cup \big(\widehat{Z} - \{\widehat{z}\}\big)$ for some $x \in X,\ z \in Z$ and $\widehat{z} \in \widehat{Z}$. But however each vertex of $\Gamma(R_1 \times R_2)$ has a unique representation with respect to $\mathfrak{B}'''$, therefore $\mathfrak{B}'''$ forms a resolving set. Consequently, the unique element of Y does not belong to any metric basis (and upper basis). Therefore, by Theorem 14 and 11, $\mathtt{dim}^+(\Gamma(R_1 \times R_2)) = \mathtt{dim}(\Gamma(R_1 \times R_2))$ and in fact this number equals $(|X| - 1) + (|Z| - 1) + (|\widehat{Z}| - 1) = (|Z^*(R_1)| - 1) + (|U(R_1)| - 1) + (|Z^*(R_1)| - 1)$. Since R_1 is a finite commutative ring with unity and so each element is either a unit or a zero divisor, therefore $\mathtt{dim}^+(\Gamma(R_1 \times R_2)) = \mathtt{dim}(\Gamma(R_1 \times R_2)) = |R_1| + |Z^*(R_1)| - 4$. This completes the proof. □

Remark 23 Let $R = \{x_0, x_1, \ldots, x_n, \ldots\}$, where x_0 is the zero element of R, be an integral domain, the vertex set of the zero divisor graph $\Gamma(\mathbb{Z}_4 \times R)$ can always be partitioned into four distance similar equivalence classes namely, $A = \{1,3\} \times \{0\} = \{(1,0),(3,0)\}$, $B = \{0\} \times \{x_1, x_2, \ldots, x_n, \ldots\}$, $C = \{(2,0)\}$ and $D = \{2\} \times \{x_1, x_2, \ldots, x_n, \ldots\}$. Choose a vertex $a \in A,\ b \in B,\ c \in C$ and $d \in D$. Then for all a, b, c and d, we have $\mathtt{deg}(a) = |R| - 1, \mathtt{deg}(d) = 1,\ \mathtt{deg}(b) = 3$ and $\mathtt{deg}(c) = 2|R| - 2$. For an integral domain R, the zero divisor graphs associated to $\mathbb{Z}_4 \times R$ have similar shape except to the number of vertices in the partitions B and D, and degrees of vertices in the partitions A, C and D.

Example 24 Let $R_1 = \mathbb{Z}_4$ or $\dfrac{\mathbb{Z}_2[x]}{(x^2)}$ and $R_2 = \mathbb{F}_{16}$. Then, by Theorem 22, we have $\mathfrak{Dim}^+(\Gamma(\mathbb{Z}_4 \times \mathbb{F}_{16})) = \mathfrak{Dim}(\Gamma(\mathbb{Z}_4 \times \mathbb{F}_{16})) = 2|\mathbb{F}_{16}| - 3 = 29$. From the zero divisor graph of $\Gamma(\mathbb{Z}_4 \times \mathbb{F}_{16})$ (see Figure 2), we notice that for a field $\mathbb{F}$, a metric basis (or upper basis) for $\Gamma(\mathbb{Z}_4 \times \mathbb{F})$ can be formed by taking one element from $A = \{(1,0),(3,0)\}$ and any $|\mathbb{F}| - 1$ elements from each of the sets $B = \{0\} \times U(\mathbb{F})$ and $D = \{2\} \times U(\mathbb{F})$. Therefore, $\mathfrak{Dim}^+(\Gamma(\mathbb{Z}_4 \times \mathbb{F})) = \mathfrak{Dim}(\Gamma(\mathbb{Z}_4 \times \mathbb{F})) = 2(|\mathbb{F}| - 2) + 1 = 2|\mathbb{F}| - 3$.

Now, if $R_1 = \mathbb{Z}_{16}$ and $R_2 = \mathbb{Z}_2$, then under the notations of Theorem 22, we have

$$A = U(R_1) \times \{0\} = \{(1,0),(3,0),(5,0),(7,0),(9,0),(11,0),(13,0),(15,0)\}$$

$$B = \{0\} \times \{1\} = \{(0,1)\},\ C_1 = V_1 \times \{0\} = \{(2,0),(6,0),(10,0),(14,0)\}$$
$$C_2 = V_2 \times \{0\} = \{(4,0),(12,0)\},\ C_3 = V_3 \times \{0\} = \{(8,0)\}$$
$$D_1 = V_1 \times \{1\} = \{(2,1),(6,1),(10,1),(14,1)\}$$
$$D_2 = V_2 \times \{1\} = \{(4,1),(12,1)\},\ D_3 = V_3 \times \{1\} = \{(8,1)\}$$

Therefore, by Theorem 22, $\mathfrak{Dim}^+(\Gamma(\mathbb{Z}_{16}\times\mathbb{Z}_2)) = \mathfrak{Dim}(\Gamma(\mathbb{Z}_{16}\times\mathbb{Z}_2)) = (|A|-1)$ $+(|C_1|-1)$ $+(|C_2|-1)$ $+(|D_1|-1)$ $+(|D_2|-1) = 7+3+1+3+1 = 15$.

In general, we see that for any positive integer $n \geq 3$,

$$\begin{aligned}
\mathfrak{Dim}^+(\Gamma(\mathbb{Z}_{2^n} \times \mathbb{Z}_2)) &= \mathfrak{Dim}(\Gamma(\mathbb{Z}_{2^n} \times \mathbb{Z}_2)) \\
&= (|U(R_1)|-1) + \sum_{i=1}^{n-2}(|C_i|-1) + \sum_{i=1}^{n-2}(|D_i|-1) \\
&= (2^{n-1}-1) + 2\sum_{i=1}^{n-2}(|V_i|-1) \\
&= (2^{n-1}-1) + 2\big(|V_1| + |V_2| + \cdots + |V_{n-2}| - (n-2)\big) \\
&= (2^{n-1}-1) + 2(|\Gamma(\mathbb{Z}_{2^n})| - n + 1) \\
&= 2^n + 2^{n-1} - 2n - 1.
\end{aligned}$$

Now, let $S_1 = \dfrac{\mathbb{F}_4[y]}{(y^2)}$, where $\mathbb{F}_4 = \dfrac{\mathbb{Z}_2[x]}{(1+x+x^2)}$ is a field with four elements. Then S_1 is a local ring such that $\Gamma(S_1)$ is a complete graph ($\cong K_3$) and let $S_2 = \mathbb{Z}_2$. Then in the notations of Theorem 22, the distance similar equivalence partition for $\Gamma\left(\dfrac{\mathbb{F}_[y]}{(y^2)} \times \mathbb{Z}_2\right)$ is given as (see Figure 2, $\Gamma\left(\dfrac{\mathbb{F}_4[y]}{(y^2)} \times \mathbb{Z}_2\right)$,

$$X = \{(y,0),(xy,0),(xy+y,0)\}, \quad Y = \{(0,1)\}, \quad \widehat{Z} = \{(y,1),(xy,1),(xy+y,1)\}$$
$$Z = \{(1,0),(x,0),(1+x,0),(1+y,0),(x+y,0),(1+x+y,0),(xy+1,0),(x+xy,0),$$
$$(1+x+xy,0),(1+y+xy,0),(x+y+xy,0),(1+x+y+xy,0)\}.$$

The set of pendant vertices in $\Gamma\left(\dfrac{\mathbb{F}_4[y]}{(y^2)} \times \mathbb{Z}_2\right)$ is the set of vertices given by $Z = U(R_1) \times \{0\}$. Therefore, by Theorem 22, $\mathfrak{Dim}^+\left(\Gamma\left(\dfrac{\mathbb{F}_4[y]}{(y^2)} \times \mathbb{Z}_2\right)\right) = \mathfrak{Dim}\left(\Gamma\left(\dfrac{\mathbb{F}_4[y]}{(y^2)} \times \mathbb{Z}_2\right)\right) = \left|\dfrac{\mathbb{F}_4[y]}{(y^2)}\right| + \left|Z^*\left(\dfrac{\mathbb{F}_4[y]}{(y^2)}\right)\right| - 4 = 16+3-4 = 15.$

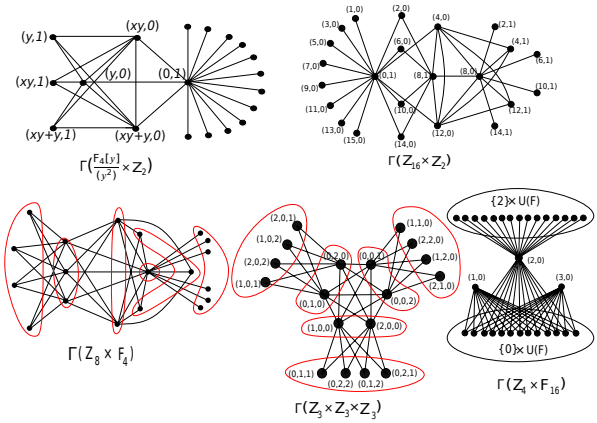


Figure 2:

A similar partition of $\Gamma(\mathbb{Z}_8 \times \mathbb{F}_4)$ and $\Gamma(\mathbb{Z}_3 \times \mathbb{Z}_3 \times \mathbb{Z}_3)$ into the distance similar equivalence classes is displayed in Figure 2.

We also notice that the graphs given in Figure 2 are either symmetric with respect to horizontal or vertical axis. This symmetry is achieved easily with the help of partition given in Theorem 22.

A (connected) graph G is said to be Hamiltonian if it contains a cycle that traverses every vertex of G. In the following theorem, $R^{\times}$ shall denote the set of units of the ring R.

Theorem 25 *Let* $R \cong R_1 \times R_2$ *be a commutative ring such that* $\dim^+(\Gamma(R)) < \infty$. *Then if* $\Gamma(R)$ *is Hamiltonian,* $\dim^+(\Gamma(R)) = \dim(\Gamma(R))$.

Proof. As $\dim^+(\Gamma(R)) < \infty$, therefore by Theorem 7, R is finite. We claim that if $\Gamma(R)$ has to be Hamiltonian then both R_1 and R_2 must be integral domains. Assume to the contrary and define $X = \{0\} \times Z^*(R_2)$ and $Y = \big(R_1 - Z(R_1)\big) \times Z^*(R_2)$. Then there is $x \in X$ and $y \in Y$ such that $xy = 0$ and for every $y_1, y_2 \in Y$, we have $y_1 y_2 \neq 0$, i.e., Y is an independent subset of $V(\Gamma(R))$. Now, by definition, a Hamiltonian cycle in $\Gamma(R)$ contains all vertices of Y and therefore contains a matching between X and Y. As the set Y is an independent subset of vertices, it follows that $|Y| \leq |X|$. But this implies that $|R_1 - Z(R_1)| \leq 1$ whence it follows that identity element is the only unit in R_1. Therefore, $R_1 \cong \Pi \mathbb{Z}_2^k$ for some positive integer k. Let $z_1 = (1, 1, \cdots, 1, 0) \in R_1$, then the vertex $(z_1, 1) \in V(\Gamma(R_1 \times R_2))$ is only adjacent to $z_2 = (0, 0, \cdots, 0, 1, 0)$, which is a contradiction to the fact that $\Gamma(R)$ is Hamiltonian. Thus, both R_1 as well as R_2 are integral domains, therefore the vertex set of $\Gamma(R)$ can be partitioned

into two distance similar equivalence classes $V_1 = R_1^\times \times \{0\}$ and $V_2 = \{0\} \times R_2^\times$ (of orders $|R_1| - 1$ and $|R_2| - 1$). Now, if either $|R_1| = 2$ or $|R_2| = 2$. then $\Gamma(R)$ is a star graph, otherwise the result follows by Theorem 14. □

In the following theorem, we give a formula for computing the metric and upper dimension of zero divisor graph of a class of rings given by $\Gamma(R) \times \Gamma(\mathbb{Z}_2 \times \mathbb{Z}_2)$ and prove that the two are equal.

Theorem 26 *Let* R *be a finite commutative ring with unity such that* $xy = 0$ *for all* $x, y \in Z^*(R)$. *Then* $\dim(\Gamma(R) \times \Gamma(\mathbb{Z}_2 \times \mathbb{Z}_2)) = \dim^+(\Gamma(R) \times \Gamma(\mathbb{Z}_2 \times \mathbb{Z}_2)) = |Z^*(R)| - 1$, *where* $|Z^*(R)| \geq 3$.

Proof. Let $G = \Gamma(R) \times \Gamma(\mathbb{Z}_2 \times \mathbb{Z}_2)$. Assume the two copies of $\Gamma(R)$ in G be denoted by Γ_1 and Γ_2. Let $V(G) = \{x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n\}$, where $|Z^*(R)| = n$, such that $x_i's$ and $y_j's$, $1 \leq i,\ j \leq n$, are vertices of Γ_1 and Γ_2 respectively and suppose, without loss of generality, the adjacencies between Γ_1 and Γ_2 be $x_i \sim y_j$ if and only if $i = j$.

For n =3, it is easily verified that any two vertex subset of Γ_i, $i = 1, 2$, is a metric and an upper basis for G. Note that in this case, the basis sets of $\Gamma(R)$ are the only basis sets of $\Gamma(R) \times \Gamma(\mathbb{Z}_2 \times \mathbb{Z}_2)$. So assume $n \geq 4$.

Let $\mathfrak{B}$ be a minimal resolving set for G. If $x_i, x_j \in V(G) - \mathfrak{B}$ with $i \neq j$ such that $y_i \notin \mathfrak{B}$ and $y_j \notin \mathfrak{B}$, then $r(x_i|\mathfrak{B}) = r(x_j|\mathfrak{B})$, since $d(x_t, x_i) = d(x_t, x_j) = 1$ for all $x_t \in \mathfrak{B}$ and $d(y_s, x_i) = d(y_s, x_j) = 2$ for all $y_s \in \mathfrak{B}$. Hence, $|\mathfrak{B}| \geq n - 1$.

For an example of a minimal resolving set of order $n-1$, consider $\mathfrak{B}_0 = \{x_1, x_2, \ldots, x_{n-1}\}$. Note that $r(x_n|\mathfrak{B}_0) = (1, 1, \ldots, 1)$, $r(y_n|\mathfrak{B}_0) = (2, 2, \ldots, 2)$ and, for each $1 \leq i < n$, $r(y_i|\mathfrak{B}_0)$ is the vector with 1 in the ith coordinate and 2 in all other coordinates. With a similar argument, it can be shown that every subset $\mathfrak{B}_1$ of order $n - 1$ for which $\mathfrak{B}_1 \cap \{x_i, y_i\} = \varphi$ for only one index i and $|\mathfrak{B}_1 \cap \{x_j, y_j\}| = 1$ for all $j \neq i$ is a minimal resolving set.

Next, assume $\mathfrak{B}_2$ is a minimal resolving set with $|\mathfrak{B}_2| \geq n$. Then $\mathfrak{B}_2$ cannot contain a subset of the type described in the previous paragraph. Hence, there must be some k such that $x_k \in \mathfrak{B}_2$ and $y_k \in \mathfrak{B}_2$.

Consider $\mathfrak{B}_3 = \mathfrak{B}_2 - \{x_k\}$. We will show that $\mathfrak{B}_3$ is a resolving set. Suppose $a, b \in V(G) - \mathfrak{B}_2$ with $a \neq b$ and $r(a|\mathfrak{B}_3) = r(b|\mathfrak{B}_3)$ but $r(a|\mathfrak{B}_2) \neq r(b|\mathfrak{B}_2)$. This means, without loss of generality, $d(a, x_k) = 1$ and $d(b, x_k) = 2$. Thus $a = x_r$ for some $r \neq k$ and $b = y_q$ for some $q \neq k$. But then $d(a, y_k) = 2$ and $d(b, y_k) = 1$, contradicting $r(a|\mathfrak{B}_3) = r(b|\mathfrak{B}_3)$. Hence, if $a, b \in V(G) - \mathfrak{B}_2$ with $a \neq b$, then $r(a|\mathfrak{B}_3) \neq r(b|\mathfrak{B}_3)$.

Finally, assume $c \in V(G) - \mathfrak{B}_2$ such that $r(c|\mathfrak{B}_3) = r(x_k|\mathfrak{B}_3)$. Since this implies $d(c, y_k) = d(x_k, y_k) = 1$, $c = y_p$ for some $p \neq k$. If there is some

$y_m \in \mathfrak{B}_3$ with $m \notin \{k, p\}$, then $d(x_k, y_m) = 2$ and $d(c, y_m) = 1$. If there is no such $y_m \in W_3$, since $|\mathfrak{B}_3| \geq n - 1 \geq 3$, there must be some $x_g \in \mathfrak{B}_3$ with $g \notin \{k, p\}$. Then, $d(x_k, x_g) = 1$ and $d(c, x_g) = 2$. In all possible cases, $r(x_k|\mathfrak{B}_3) \neq r(c|\mathfrak{B}_3)$. Thus W_3 is a resolving set, showing that $\mathfrak{B}_2$ was not a minimal resolving set. Hence, any minimal resolving set must have $n - 1$ elements. □

Note. If R is a finite commutative ring with $|Z^*(R)| = 2$, then $\Gamma(R) \cong K_2$ so that $R \cong \mathbb{Z}_2 \times \mathbb{Z}_2$, or Z_9 or $\dfrac{Z_2[x]}{(x^2)}$ and so in this case $\Gamma(R) \times \Gamma(\mathbb{Z}_2 \times \mathbb{Z}_2) \cong K_{2,2}$, therefore $\dim(\Gamma(R) \times \Gamma(\mathbb{Z}_2 \times \mathbb{Z}_2)) = \dim^+(\Gamma(R) \times \Gamma(\mathbb{Z}_2 \times \mathbb{Z}_2)) = 2$.

Corollary 27 *Let $\mathbb{F}$ be a finite field and let $R = \dfrac{\mathbb{F}[x_1, x_2, \cdots, x_n]}{I}$, where I is the ideal generated by the set $\{x_i x_j | 1 \leq i \leq n, 1 \leq j \leq n\}$. Then $\dim^+(\Gamma(R) \times \Gamma(\mathbb{Z}_2 \times \mathbb{Z}_2)) = \dim(\Gamma(R) \times \Gamma(\mathbb{Z}_2 \times \mathbb{Z}_2)) = |\mathbb{F}|^n - 2$.*

Proof. We write $R = \dfrac{\mathbb{F}[x_1, x_2, \cdots, x_n]}{I} = \{a_0 + a_1x_1 + \cdots + a_nx_n : \ a_i \in \mathbb{F}\}$. Thus, $Z(R) = \{a_1x_1 + \cdots + a_nx_n : \ a_i \in \mathbb{F}, \ i = 1, 2, \ldots, n\}$. Hence $Z^*(R) = |\mathbb{F}|^n - 1$. Clearly, the product of any two elements of $Z(R)$ is zero. Hence the result follows. □

Theorem 28 *If $n \geq 4$ is a positive integer, then $\dim^+(\Pi\mathbb{Z}_2^n) \geq n$.*

Proof. Assume $n \geq 4$, and choose a subset $\mathfrak{B} = \{e_1, e_2, \ldots, e_n\} \subset V(\Gamma(\Pi\mathbb{Z}_2^n))$, where e_i has i^{th} coordinate as non-zero and all other coordinates zero. Put $e = e_1 + e_2 + \cdots + e_n$, and let z be any vertex in $V(\Pi\mathbb{Z}_2^n) - \mathfrak{B}$. Then $d(v, e_i) = 1$ if and only if i^{th} coordinate of v is zero, otherwise $d(v, e_i) = 2$. Therefore, $r(v|\mathfrak{B}) = v + e$ and so $\mathfrak{B}$ forms a resolving set for $\Gamma(\Pi\mathbb{Z}_2^n)$. Further, one can see that $\mathfrak{B}$ forms a minimal resolving set, as by removing e_i from $\mathfrak{B}$ to obtain $\mathfrak{B}_i$, $1 \leq i \leq n-1$, the vertices $x = e_1 + e_2 + \cdots + e_{n-1}$ and $y = e_1 + e_2 + \cdots + e_{n-1} - e_i$ have the same representations with respect to $\mathfrak{B} - \{e_i\}$. Also, with respect to $\mathfrak{B}_n = W - \{e_n\}$, the vertices $x' = e_2 + e_3 + \cdots + e_n$ and $y' = x' - e_n$ have the same representations. This shows that B forms an upper basis for $\Pi\mathbb{Z}_2^n$. Consequently it follows that $\dim^+(\Pi\mathbb{Z}_2^n) \geq n$. □

Remark 29 As an illustration to Theorem 28, we choose an example of a minimal resolving set for $\Pi\mathbb{Z}_2^5$ as $\mathfrak{B}$ =(1, 0, 0, 0, 0), (0, 1, 0, 0, 0), (0, 0, 1, 0, 0), (0, 0, 0, 1, 0), (0, 0, 0, 0, 1), where by removing $(1, 0, 0, 0, 0)$ from $\mathfrak{B}$ to obtain $\mathfrak{B}_1$, we have $r((1, 1, 1, 1, 0)| \ \mathfrak{B}_1) = r((0, 1, 1, 1, 0)$

$| \mathfrak{B}_1)$ =(2, 2, 2, 1), removing (0,1,0,0,0) to get $\mathfrak{B}_2$ gives $r((1,1,1,1,0) \mid \mathfrak{B}_2)$ $= r((1,0,1,1,0)|\mathfrak{B}_2) = (2,2,2,1)$, removing $(0,0,1,0,0)$ to obtain $\mathfrak{B}_3$ gives $r((1,1,1,1,0) \mid \mathfrak{B}_3) = r((1,1,0,1,0) \mid \mathfrak{B}_3)= (2,2,2,1)$, removing (0, 0, 0, 1, 0) to get $\mathfrak{B}_4$ gives $r((1,1,1,1,0) \mid \mathfrak{B}_4) = r((1,1,1,0,0) \mid \mathfrak{B}_4) = (2,2,2,1)$ and removing (0, 0, 0, 0, 1) to get $\mathfrak{B}_5$ gives $r((0,1,1,1,1) \mid \mathfrak{B}_5) = r((0,1,1,1,0) \mid \mathfrak{B}_5) = (1,2,2,2)$.

While examining the metric dimension (and upper dimension) of zero divisor graphs of small finite commutative rings R with $|V(\Gamma(R))| \leq 14$, we found that there is only one ring i.e., $R \cong \Pi\mathbb{Z}_2^n,\ n \geq 4$ for which $\dim^+(\Gamma(R)) \neq \dim(\Gamma(R))$. It has been earlier shown in Remark 21 that $\dim(\Gamma(\Pi\mathbb{Z}_2^4)) = 3$, whereas $\dim^+(\Gamma(\Pi\mathbb{Z}_2^4)) = 4$. It is also not difficult to check that a set of $\frac{n(n-1)}{2} - 1$ elements of $\Pi\mathbb{Z}_2^n,\ n \geq 4$, that have exactly two non-zero coordinates forms a minimal resolving set for $\Gamma(\Pi\mathbb{Z}_2^n)$. A complete list of rings with 14 or fewer vertices with given metric dimension can be found in [19] and the zero divisor graphs of such rings can be found in [22].

Unlike Theorem 1 for graphs in general, with the results obtained in this paper and the observations made during the work and by the inspection of the zero divisor graphs of rings, there is a reason to believe that the metric dimension and the upper dimension of zero divisor graph of a ring R is always same, unless $R \cong \Pi\mathbb{Z}_2^n,\ n \geq 4$. We conclude the paper with the following open problem.

Conjecture 30 *Let R be a finite commutative ring with unity $1 \neq 0$, then $\dim^+(\Gamma(R)) = \dim(\Gamma(R))$, unless $R \cong \Pi\mathbb{Z}_2^n$, where $n \geq 4$.*

Acknowledgements. *This research is supported by JRF financial assistance (M. Aijaz) by Council of Scientific and Industrial Research (CSIR), New Delhi India.*

References

[1] S. Akbari, A. Mohammadian, On the zero-divisor graph of finite rings, *J. Algebra* **314,** 1 (2007) 168–184. ⇒88

[2] D. F. Anderson and P. S. Livingston, The zero-divisor graph of a commutative ring, *J. Algebra* **217** (1999) 434–447. ⇒85

[3] D. D. Anderson, M. Naseer, Beck's coloring of a commutative ring, *J. Algebra* **159** (1993) 500–517. ⇒85

[4] M. F. Atiyah, I. G. MacDonald, *Introduction to Commutative Algebra*, Addison-Wesley, Reading, MA, (1969). ⇒86

[5] M. Baziar, E. Momtahan, S. Safaeeyan, Zero-divisor graph of abelian groups, *J. Algebra Appl.* **13,** 6 (2014) 1450007, 13 pages. ⇒85

[6] I. Beck, Coloring of commutative rings, *J. Algebra* **116** (1988) 208–226. ⇒85
[7] G. Chartrand, C. Poisson and P. Zhang, Resolvability and the upper dimension of graphs, *Int. J. Computers and Mathematics with Appl.* **39** (2000) 19–28. ⇒86, 87
[8] A. Das, On nonzero component graph of vector spaces over finite fields, *J. Algebra Appl.* (2016), doi: 10.1142/S0219498817500074. ⇒85
[9] D. Dolzan, P. Oblak, The zero divisor graphs on rings and semi rings, *International J. Algebra Computation* **22,4** (2012) 1250033, 2 pages. ⇒85
[10] D. Garijo, A. Gonzalez, A. Marquez, On the metric dimension, the upper dimension and the resolving number of graphs, *Discrete Appl. Math.* **161** (2013) 1440–1447. ⇒86
[11] F. Harary, R. A. Melter, On the metric dimension of a graph, *Ars Combin.* **2** (1976) 191–195. ⇒86
[12] I. Kaplansky, *Commutative Rings*, rev. ed., Univ. of Chicago Press, Chicago, (1974). ⇒86
[13] C. F. Kimball, J. D. LaGrange, The idempotent-divisor graphs of a commutative ring, Comm. in Algebra, February 2018. DOI: 10.1080/00927872.2018.1427245 ⇒85
[14] S. Pirzada, *An Introduction to Graph Theory*, Univ. Press, Hyderabad, India, 2012. ⇒86
[15] S.Pirzada, M. Aijaz, On graphs with same metric and upper dimension, Communicated. ⇒87, 88
[16] S. Pirzada, M. Aijaz, M. Imran Bhat, On zero-divisor graphs of the rings $\mathbb{Z}_n$, *Afrika Matematika* **31,** (2020) 727–737. ⇒90
[17] S. Pirzada, M. Aijaz, S. P. Redmond, Upper dimension and bases of zero-divisor graphs of commutative rings, *AKCE International J. Graphs Comb.* 2019 ⇒85, 86, 88, 91
[18] S. Pirzada, M. Aijaz, S. P. Redmond, On upper dimension of some graphs and their bases sets, Communicated. ⇒87, 88, 89
[19] S. Pirzada, R. Raja, S. P. Redmond, Locating sets and numbers of graphs associated to commutative rings, *J. Algebra Appl.* **13,** 7 (2014) 1450047, 18 pages. ⇒91, 100
[20] S. Pirzada, R. Raja, On the metric dimension of a zero-divisor graph, *Comm. Algebra* **45** (2017) 1399–1408. ⇒86
[21] R. Raja, S. Pirzada, S. P. Redmond, On Locating numbers and codes of zero-divisor graphs associated with commutative rings, *J. Algebra Appl.* **15**, 1 (2016) 1650014, 22 pages. ⇒86, 87, 91
[22] S. P. Redmond, On zero divisor graphs of small finite commutative rings, *Discrete Math.* **307** (2007) 1155–1166. ⇒100
[23] S. Safaeeyan, M. Baziar, E. Momtahan, A generalization of the zero-divisor graph for modules, *J. Korean Math. Soc.* **51,** 1 (2014) 87–98. ⇒85
[24] P. J. Slater, Leaves of trees, *Congr. Number.* **14** (1975) 549–559. ⇒86

Received: February 20, 2020 • Revised: March 23, 2020

Acta Univ. Sapientiae, Informatica **12,** 1 (2010) 102–111

DOI: 10.2478/ausi-2020-0007

Encouraging an appropriate representation simplifies training of neural networks

Krisztian BUZA[a,b]
[a] Faculty of Informatics
Eövös Loránd Unviersity (ELTE)
Budapest, Hungary
[b] Center for the Study of Complexity
Babeş-Bolyai University
Cluj Napoca, Romania
email: buza@biointelligence.hu

Abstract. A common assumption about neural networks is that they can learn an appropriate internal representation on their own, see e.g. end-to-end learning. In this work we challenge this assumption. We consider two simple tasks and show that the state-of-the-art training algorithm fails, although the model itself is able to represent an appropriate solution. We will demonstrate that encouraging an appropriate internal representation allows the same model to solve these tasks. While we do not claim that it is impossible to solve these tasks by other means (such as neural networks with more layers), our results illustrate that integration of domain knowledge in form of a desired internal representation may improve the generalization ability of neural networks.

1 Introduction

Traditionally, the applications of machine learning algorithms were closely coupled with careful feature engineering requiring extensive domain knowledge. In contrast, in the era of deep learning, a common assumption is that neural

Computing Classification System 1998: I.2.6
Mathematics Subject Classification 2010: 68T07
Key words and phrases: neural networks, representation, integration of domain knowledge

networks are able to learn appropriate representations that may be better than the features defined by domain experts, see end-to-end learning for self-driving cars [4], speech recognition [2] and other applications [7, 14].

Recent success stories related to neural networks include mastering board games [15], the diagnosis of various diseases, such as skin cancer [5], retinal disease [6] and mild cognitive impairment [9]. Despite these (and many other) spectacular results, there is increasing evidence indicating that neural networks do not learn the underlying concepts: minor alterations of images, that are invisible to humans, may lead to erroneous recognition [16, 12], e.g., slight modifications of traffic signs "can completely fool machine learning algorithms" [1].

In this paper we will demonstrate that encouraging an appropriate representation may substantially simplify the training of (deep) neural networks. In particular, we consider two simple tasks and show that the state-of-the-art training algorithm fails, although the model itself is able to represent an appropriate solution. Subsequently, we will see that encouraging an appropriate internal representation allows the same model to solve the same tasks. The resulting networks not only have good generalization abilities, but they are more understandable to human experts as well.

The reminder of the paper is organised as follows: next, we will explain what we mean by "encouraging a representation" (Section 2). In Section 3 and Section 4 we will demonstrate in the context of two transformation tasks that the proposed idea may substantially improve the accuracy of the model. Finally, we discuss the implications of our observations to other applications in Section 5.

2 Encouraging a representation

With "encouraging a representation" we mean to train a neural network in a way that some of the hidden units correspond to predefined concepts. We would like to emphasize that this requirement is meant for a relatively small subset of all the hidden units. For example, if we want to train a network for the recognition of traffic signs in images, the network is likely to have thousands of hidden units and we may require that some of those hidden nodes recognize particular shapes or letters, i.e., their activations should be related to the presence of a triangle, an octagon, or particular letters like 'S', 'T', 'O' and 'P'.

The idea of encouraging a representation may be implemented in various ways: for example, in the cost function, we may include a regularisation term that penalises the situation if the activation of some given nodes is inappropriate. In one of our previous works, in the context of matrix factorisation for drug-target interaction prediction, we encouraged a lower dimensional representation so that the distances between drugs and targets are in accordance with their chemical and genomic similarities [13].

In this paper, we consider encoder-decoder networks. In particular, we will "encourage" the encoder to learn a pre-defined representation. This representation is the output of the encoder and serves as the input of the decoder. We demonstrate that encouraging an appropriate representation may lead to a substantial increase of the accuracy.

3 TraNet: a network for translation and transcription

We consider two tasks, Translation and Transcription (see Section 3.1). We try to solve both of these tasks with the same neural network, called *TraNet* (as "Tran" is a common prefix of the names of these two tasks). When solving the two tasks, the only difference is the input layer in accordance with the input of the tasks.

We implemented all the experiments in Python using numpy, matplotlib and TensorFlow with the keras API. In order to assist reproducibility, our implementation is available at `http://www.biointelligence.hu/encourage/` .

3.1 Benchmark tasks: translation and transcription

As benchmark tasks, we consider the translation of written numbers from English to German (e.g. *twenty-five* should be translated to *funfundzwanzig*[1]), and recognition of 4-digit handwritten numbers where the desired output is the number written in English, see also Fig 1. For simplicity, we will refer to these tasks as *Translation* and *Transcription*.

As neural networks are used for substantially more complex tasks, see e.g. [18] for a review, we expect them to have an excellent performance in case of our tasks.

[1]For simplicity, instead of the German special letters 'ä', 'ö', 'ü' and 'β', we use 'a', 'o', 'u' and 'ss' respectively.

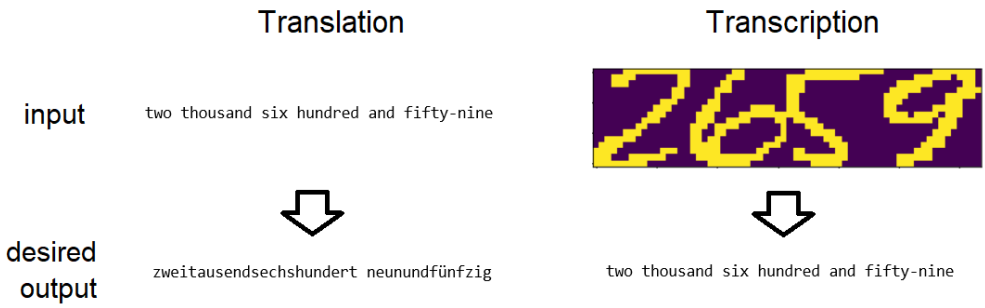


Figure 1: Illustration of the considered tasks, Translation (left) and Transcription (right).

3.2 Architecture

TraNet is a feed-forward encoder-decoder network. For strings (i.e. numbers written in English or German) we use letter-wise one-hot encoding. For example, 'a' is coded as $(1, 0, 0, 0, ..., 0)$, 'b' is coded as $(0, 1, 0, 0, ..., 0)$, etc. The entire string is coded as the concatenation of the codes of each letter. We allow at most a length of 50 letters, therefore, the output layer of TraNet contains $50 \times 29 = 1450$ units. Similarly, in the case of the Translation task, the input layer of TraNet contains 1450 units as well. When TraNet is used for Transcription, the input is a binary image with $64 \times 16 = 1024$ pixels, therefore, the input layer contains 1024 units, each one corresponding to one of the pixels. The input layer does not perform any operation, its sole purpose is to represent the input data.

An appropriate internal representation, which *could* be learned by the network, is the *digit-wise one-hot encoding* of the 4-digit number that was shown to the network. Each digit between 0 and 9 is coded as a binary vector of length 10. In particular, digit '0' is coded as $(1, 0, 0, 0, ..., 0)$, '1' is coded as $(0, 1, 0, 0, ..., 0)$, etc. The digit-wise one-hot encoding of a 4-digit number is the concatenation of the vectors corresponding to the digits of the number, therefore, it has a total length of $4 \times 10 = 40$.

TraNet contains 3 hidden layers. The first and third layers contain 1000 units with ReLU activation function. The second hidden layer contains 40 units so that it might potentially learn the aforementioned digit-wise one-hot encoding. In the second layer, we use the sigmoid activation function, as it may be suited to the digit-wise one-hot encoding.

As loss function, we use binary cross-entropy. We trained TraNet for 100 epochs using the ADAM optimizer [8]. We performed the computations on CPU (i.e., no GPU/TPU support was used).

4 Results

Next, we compare conventional training and training with encouraging the digit-wise one-hot encoding as internal representation.

4.1 Translation

In the case of Translation task, we considered the numbers between 0 and 9999, written in English (input) and German (desired output). A randomly selected subset of 100 numbers was used as test data, while the remaining 9900 instances were used as training data. We repeated the experiments 5-times with a different initial split of the data into training and test data.

In case of conventional training, denoted as *Conventional TraNet*, the resulting network was not able to give an exact translation for any of the numbers of the test set. While in some cases the translations generated by the network were at least partially understandable to humans, in other cases the network failed to translate the numbers, see Tab 1 for some examples.

Although there are many possibilities to improve the model, next, we show that there is nothing wrong with the model: TraNet itself is able to represent a function that gives a reasonably good translation, the problem is the conventional training.

In order to encourage the model to learn an appropriate representation, we train the encoder and decoder separately. The encoder consists of the input layer and the first two hidden layers. It is trained to translate from English to the digit-wise one-hot encoding (described in Section 3.2), i.e., we expect the output of the encoder to be the digit-wise one-hot encoding. The input of the decoder is the second hidden layer of TraNet. Additionally, the third hidden layer of TraNet and its output layer belong to the decoder. The decoder is trained to translate from digit-wise one-hot encoding to German.

When training the encoder and decoder separately, so that the digit-wise one-hot encoding is encouraged, we observed that the network was able to translate on average 95.8 % of the numbers perfectly. From the practical point of view, the quality of the translation is even better, because in many cases when the translation was not perfect, we observed only minor spelling mistakes, such as “einhundert einundneenzig” instead of “einhundert einundneun-

Input	**Output of conventional TraNet**	**Output of encouraged TraNet**
one hundred and ninety one	einaaudenaaaiaaaaaaaaaa	einhundert einundneenzig
four thousand two hundred and twenty-five	vieatausenazieihundert aieaanaaaanai	viertausendzweihundert funfundzwanzig
eight thousand eight hundred and sixty	acattausenaaaeihundert aieaaaaa	achttausendachthundert sechzig
seven hundred and sixty-six	aaebenaunaeaaaaaanan-aaaaaiaeaa	siebenhundert sechsundsehhzig

Table 1: Examples for translation of numbers from English to German *with* and *without* encouraging the digit-wise one-hot encoding as internal representation, denoted as "encouraged TraNet" and "conventional TraNet", respectively.

zig", or "siebenhundert sechsundsehhzig" instead of "siebenhundert sechsundsechzig". Consequently, encouraging an appropriate representation lead to an accurate model for the translation of numbers from English to German.

4.2 Transcription

In order to obtain 4-digit handwritten numbers, we considered the *Semeion* dataset[2], which is a publicly available dataset containing images of handwritten digits. As each of the images shows a single digit, in order to obtain an image of a four-digit handwritten number, we choose four images randomly and stack them horizontally, see Fig. 1 for an example. The last 100 images of the Semeion dataset are used to obtain test images, whereas the training data is obtained from the first 1493 images. In total, we obtained 100 000 training images and 1000 test images of 4-digit numbers. We repeated the experiment 5-times with different training and test images.

Conventional training of TraNet, i.e., training without encouraging the digit-wise one-hot encoding as intermediate representation, lead to a network that could not transcribe any of the images correctly. Instead, the output of TraNet was always a number-like phrase without clear meaning, such as "fivet huusadd ne hundded and" or "tivethhusand ne hundred and". In contrast, encouraging the digit-wise one-hot encoding, i.e., training the encoder and decoder sepa-

[2] `https://archive.ics.uci.edu/ml/datasets/Semeion+Handwritten+Digit`

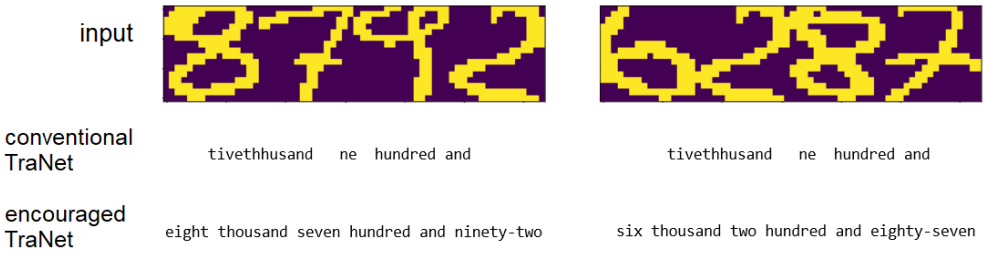


Figure 2: Examples for transcription *with* and *without* encouraging the digit-wise one-hot encoding as internal representation, denoted as "encouraged TraNet" (bottom) and "conventional TraNet" (center), respectively.

rately, lead to a network that was able to perfectly transcribe on average 74.9 % numbers of the test data, while it made mostly minor spelling mistakes in the remaining cases, see Fig. 2 for some illustrative examples.

5 Discussion

While we do not claim that there are no other ways to find an acceptable solution, in fact, we observed that using more deep neural networks achieved comparable performance to that of encouraged TraNet[3], our results clearly demonstrate the power of encouraging the digit-wise one-hot encoding.

Besides allowing a relatively simple model to achieve good generalisation ability, our encouraged models are better understandable to domain experts. On the one hand, this may increase the trust in the model, and it allows to "debug" the model: if the output is incorrect, one could check whether the activations in the 2nd hidden layer are close to that of in case of the desired internal representation.

Although the fact that a neural network may learn concepts that are substantially different from human concepts, may be considered as an advantage in many cases, we argue that it may be worth to use encouraged models together with conventional models. For example, in critical applications (credit scoring, medical diagnosis, etc.), it may be worth to carefully consider cases when conventional and encouraged models disagree.

The computational cost of training conventional and encouraged TraNet is similar: while performing forward propagation, after the calculation of the

[3]We note that, due to its simplicity, our translation task can be solved perfectly using techniques from the classic theory of formal languages.

activities in the second layer (which is necessary in both cases in order to determine the gradients of the connections in the previous layers), in case of encouraged TraNet, the desired activations (i.e., activations corresponding to the digit-wise one-hot encoding in our case) are propagated to the third layer. In case of backpropagation, the only difference between conventional and encouraged training is that in case of encouraged training, the back-propagated activitions of the second layer do not need to be calculated: the desired activations should be used instead. Therefore, the cost of performing one training iteration is asymptotically same in both cases. This is consistent with our observations, according to which encouraged training was slightly ($\approx 5\%$) faster.

Note however that we considered illustrative examples, in which conventional training failed in the sense that it did not lead to any reasonable model. In other cases, in order to achieve comparable accuracy, conventional training may require much more training epochs compared to encouraged training. Thus, encouraged training may be beneficial in terms of the total computational cost.

Although the appropriate representation depends on the underlying task, we believe that the general idea of encouraging models to learn concepts that are similar to human concepts may be beneficial in various applications ranging from the analysis of biomedical signals [10], such as ECG [17] to image classification [11]. Furthermore, in many cases, see e.g. intrusion detection, recent studies used models other than neural networks [3]. Encouraging appropriate representations might allow neural networks to be applied in such cases too.

Last, but not least, we point out that defining an appropriate representation may be seen as a way of integrating domain knowledge into the training procedure, thus it can be seen as a way of collaboration between human intelligence and machine intelligence.

Acknowledgements

K. Buza was supported by the project ED_18-1-2019-0030. Project no. ED_18-1-2019-0030 (Application domain specific highly reliable IT solutions subprogramme) has been implemented with the support provided from the National Research, Development and Innovation Fund of Hungary, financed under the Thematic Excellence Programme funding scheme. K. Buza received the “Professor Ferencz Radó” Fellowship of the Babeş-Bolyai University, Cluj Napoca, Romania.

References

[1] E. Ackerman, Slight street sign modifications can completely fool machine learning algorithms, *IEEE Spectrum* 6 (2019). ⇒103

[2] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng, G. Chen, et al., Deep speech 2: end-to-end speech recognition in english and mandarin, in: *International conference on machine learning*, 2016, pp. 173–182. ⇒103

[3] M. Antal, E. Egyed-Zsigmond, Intrusion detection using mouse dynamics *IET Biometrics* 8 (5) (2019) 285–294. ⇒109

[4] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, et al., End to end learning for self-driving cars, *arXiv* (2016) arXiv:1604.07316. ⇒103

[5] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, S. Thrun, Dermatologist-level classification of skin cancer with deep neural networks, *Nature* 542 (2017) 115. ⇒103

[6] J. De Fauw, J. R. Ledsam, B. Romera-Paredes, S. Nikolov, N. Tomasev, S. Blackwell, H. Askham, X. Glorot, B. O'Donoghue, D. Visentin, et al., Clinically applicable deep learning for diagnosis and referral in retinal disease, *Nature medicine* 24 (2018) 1342. ⇒103

[7] A. Gordo, J. Almazan, J. Revaud, D. Larlus, End-to-end learning of deep visual representations for image retrieval, *International Journal of Computer Vision* 124 (2017) 237–254. ⇒103

[8] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, *arXiv* (2014) arXiv:1412.6980. ⇒106

[9] R. J. Meszlényi, K. Buza, Z. Vidnyánszky, Resting state fmri functional connectivity-based classification using a convolutional neural network architecture, *Frontiers in neuroinformatics* 11 (2017) 61. ⇒103

[10] K. Miok, D. Nguyen-Doan, M. Robnik-Sikonja, D. Zaharie, Multiple Imputation for Biomedical Data using Monte Carlo Dropout Autoencoders, in: *E-Health and Bioengineering Conference (EHB)* (2019) ⇒109

[11] T. Nyíri, A. Kiss, Novel Ensembling Methods for Dermatological Image Classification, in: *International Conference on Theory and Practice of Natural Computing* (2018) ⇒109

[12] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, A. Swami, Practical black-box attacks against machine learning, in: *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, ACM, 2017, pp. 506–519. ⇒103

[13] L. Peška, K. Buza, J. Koller, Drug-target interaction prediction: A bayesian ranking approach, *Computer methods and programs in biomedicine* 152 (2017) 15–21. ⇒104

[14] D. Silver, H. van Hasselt, M. Hessel, T. Schaul, A. Guez, T. Harley, G. Dulac-Arnold, D. Reichert, N. Rabinowitz, A. Barreto, et al., The predictron: end-to-end learning and planning, in: *Proceedings of the 34th International Conference on Machine Learning-Volume 70* , JMLR. org, 2017, pp. 3191–3199. ⇒103

[15] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, et al., Mastering the game of go with deep neural networks and tree search, *Nature* 529 (2016) 484. ⇒103

[16] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, R. Fergus, Intriguing properties of neural networks, *arXiv* (2013) arXiv:1312.6199. ⇒103

[17] S. M. Szilagyi, L. Szilagyi, D. Iclanzan, Z. Benyó, Unified Neural Network Based Adaptive ECG Signal Analysis and Compression, *Scientific Bulletin of the Politechnica University of Timisoara, Transactions on Automatic Control and Computer Science* 51.65 (2006) 27–36. ⇒109

[18] J. Zhang, C. Zong, et al., Deep neural networks in machine translation: An overview, *IEEE Intelligent Systems* 30 (2015) 16–25. ⇒104

Received: March 26, 2020 • Revised: June 7, 2020

Acta Univ. Sapientiae, Informatica **12,** 1 (2020) 112–136

DOI: 10.2478/ausi-2020-0008

Modeling reactive magnetron sputtering: a survey of different modeling approaches

Rossi Róbert MADARÁSZ
Doctoral School
of Applied Informatics and Applied Mathematics
Óbuda University Budapest, Hungary
email: madarasz.rossi@phd.uni-obuda.hu

András KELEMEN
Sapientia Hungarian University
of Transylvania
Cluj-Napoca, Romania
email: kandras@ms.sapientia.ro

Péter KÁDÁR
Óbuda University
Budapest, Hungary
email: kadar@uni-obuda.hu

Abstract. The paper focuses on providing an insight into the current state of computational modeling regarding reactive magnetron sputtering systems. A detailed compilation of developed models is gathered and grouped into categories based on the phenomena being modeled. The survey covers models developed for the analysis of magnetron discharges, particle -surface interactions at the target and the substrate, as well as macroscopic models. Corresponding software packages available online are also presented. After gaining the necessary insight into the current state of research, a list of the most challenging tasks is given, comparing different approaches, that have been used to combat the encountered difficulties. The challenges associated with modeling tasks range from analytical complexity, mathematical know-how used for model approximation and reduction, as well as optimization between computational load and result accuracy. As a conclusion, the future challenges are compiled into a list and a probable direction in modeling is given, that is likely to be further pursued.

Computing Classification System 1998: I.6.5
Mathematics Subject Classification 2010: 93-10
Key words and phrases: mathematical modeling, parameter identification, control theory, computer simulation, reactive magnetron sputtering

1 Introduction

The goal of the survey is to offer a brief insight into the current state and results of different mathematical and computer modeling techniques and approaches used to investigate reactive magnetron sputtering.

Sputtering is a subclass of PVD - Physical Vapor Deposition, which is an essential process in manufacturing integrated circuits, photovoltaic modules, specialized optical and mechanical equipment. It is a process used for the creation of extremely thin (even down to atomic) coating layers on different surfaces in order to achieve better physical, chemical, optical or electrical characteristics, than those of the coated material itself. Among others, the use of this technique allows the formation of transparent but also conducting layers, which are essential in optics. The process essentially consists of the removal of material from one or more target surfaces and their deposition onto a surface that is required to have improved characteristics, also known as the substrate. This removal process is accomplished with the help of high energy noble gas ions which bombard the surface of the target due to the presence of an electric field. With the addition of a magnet behind the target one can achieve magnetron sputtering, greatly increasing the ionization efficiency and the sputtering yield.

With the introduction of one or more reactive gases into the vacuum chamber, reactive magnetron sputtering is achieved, which allows the formation of different compounds like oxides and nitrides. Reactive magnetron sputtering has been extensively studied and modeled in the past few decades.

There are several approaches and technologies used for magnetron sputtering, such as DC (direct current) sputtering, pulsed DC sputtering, AC (alternating current) sputtering, HiPIMS (high power impulse magnetron sputtering) and RF (radio frequency) sputtering. All of the different coating techniques mentioned require highly specialized operating environments in order to function correctly. The creation of a high vacuum environment (pressure in the range of a few Pascals) is mandatory. At such low pressures, different metals begin to evaporate, hindering the insertion of any nonspecialized equipment into the sputtering chamber. Simple questions as measuring the total or partial pressure inside the chamber become ever more difficult, requiring special processes to be put in place and the indirect calculation of the desired quantity from the gathered data. In situations like these, the use of a state identification algorithm becomes favored. From the control perspective, the system quickly becomes complex when one simply attempts to count the number of inputs, outputs and their non-linear cross coupled effect on each other. Each

of the magnetron sputtering processes set particular challenges from the point of view of modeling. This paper concentrates on gathering and presenting only computer modeling approaches used for DC reactive magnetron sputtering.

Due to the specifics of the environment, the overwhelming amount of process values to which attention has to be paid, the lack of quick and reliable access to measurable data, control of the process becomes complex enough that many laboratory experiments are conducted in an open-loop fashion. Therefore, reliable reproduction of experiments is still nearly impossible.

These constraints have compelled researchers to develop, test and validate complex models of the process. Most of the time, financial reasons are the main driving force behind the need for modeling, especially in industrial applications. The simplest desire that can be placed is the optimization of the process to achieve maximum yield. Another desire, which is far more complex is the achievement of required stoichiometric ratios between the deposited materials on the surface.

The paper is structured as follows: chapter 1 gives a brief introduction to the field of study, chapter 2 gives an overview of several prominent research results in the field of modeling, chapter 3 presents the different modeling tasks and challenges, and finally, chapter 4 outlines directions that are likely to be further pursued.

2 Literature survey of modeling approaches

Reactive magnetron sputtering is a complex process with several interacting subprocesses, representing difficult modeling tasks on their own. One of the possible approaches is the development of multi-physics models by coupling the models which describe the plasma, the magnetic field, the transportation phenomena in the chamber, the interaction of particles with the target, the thin film deposition on the substrate, etc. Such a model based on the perfect understanding of the physical phenomena would represent a perfect tool for the industry. However, the feasibility of a "virtual sputter tool" [58] resulting from this bottom-up approach is low, mainly because the level of understanding of different phenomena is not the same, and there will always be a weak element in the chain which might vitiate the overall results. A top-bottom model development, leading to a "holistic model" [58], seems to be more feasible. According to this approach, the development starts from a simplest possible model which describes "well enough" the phenomena of interest, and further on this model suffers extensions according to the necessities to cover other phe-

nomena as well. Contrary to these "white box" approaches, the input-output modeling of the system, which has little to do with the understanding of the details of its physics, can be very useful for control purposes [62].

According to the bottom-up approach, one may study the phenomena that are taking place inside the plasma discharge between charged particles. Such a model must also consider the magnetic field present in the chamber and its effects on the movement of charged particles.

A very intriguing phenomenon that can be modeled is the interaction between particles at the surface level of the target, where collisions either lead to implantation or material removal.

Particle-surface interactions are also studied at the substrate, where the formation of crystallographic structures is analyzed.

The top-bottom approach is represented in this paper by the extensions of the Berg model, the simplest one, which can efficiently describe the hysteresis phenomena in case of a single target, single reactive gas magnetron sputtering. These are macroscopic models based on equilibrium equations, and are not more complicated than necessary to reproduce the essential behavior of the system, trying to remain simple enough to facilitate control.

2.1 Modeling of magnetron discharges

An electric discharge taking place in vacuum can be divided into the following categories: dark, glow and arc discharge, each with a higher discharge current than the former.

According to Bogaerts et al. [9] there are three types of model categories for plasma discharges:

1. Fluid models, which are modeling species in the discharge as ones being in hydrodynamic equilibrium. The equations that govern are the continuity equations (fluxes and densities of the species) and the Poisson equation for the electric field. The fluid model is considered to be the simplest but also the most inaccurate one.

2. Kinetic models, which describe the plasma species as beams of particles. The movement of particles is described by the Boltzmann transport equations, combined with the Poisson equation.
3. Monte Carlo simulations, that describe the different plasma particles separately on a statistical basis. This is considered to be the most accurate method, but it requires considerable computational effort.

Depla and Mahieu compiled a comprehensive study on modeling the electric discharge in reactive magnetron sputtering [26]. They have grouped models into two main categories i.e. analytical and numerical. They further split numerical into fluid and kinetic ones. In this approach, kinetic types are either based on numerical solutions of the Boltzmann equation or on Monte Carlo simulations, which are also called particle simulations.

Wendt and Lieberman [60, 61] developed a two dimensional analytical model of the discharge process, taking into account both magnetic and electric fields.

Fluid models [14, 22, 23] have also been presented. These models calculate the electric field distribution by coupling the Poisson equation to the continuity and conservation of energy and momentum equations.

The solution of the Boltzmann equation has been widely used for modeling plasma discharges. The addition of the Lorentz force term complicates the solver, as presented by Porokhova et al. in [49, 50, 51].

In Monte Carlo simulations the path of a small number of particles is analyzed, their movement and their collisions being modeled using probabilities and random numbers [53, 29].

Hybrid models have been conceived with the assumption that heavy ions are practically not affected by the magnetic field, as opposed to the relatively lightweight electrons. Therefore, ions can be modeled by a fluid model and electrons can be modeled as particles. This assumption combines high precision of particle simulation codes with lower computational efforts of fluid codes [52, 35]. Bogaerts et al. have presented several papers modeling the magnetron discharge process using Monte-Carlo and hybrid Monte-Carlo simulations [11, 9, 10, 12].

Particle in Cell-Monte Carlo Collision (PIC-MCC) simulations are considered to be the most powerful available numerical tools. PIC-MCC simulations combine MC methods with the simulation of the electric field being produced by an external power supply and with the spatial distribution of the charged particles in the plasma [8].

2.2 Modeling particle-target interaction

When studying particle-target interactions at the surface of the target, one may find in the literature that depending on the kinetic energy and the incident angle of impact, some ions reach the surface and through chemisorption get chemically attached to the outermost layer. Others may effectively transfer their kinetic energy and lead to material removal from the target, and some may be so energetic that they get implanted into the subsurface and stay there until enough material is removed for them to resurface. Depla, Strijckmans and separately Kubart extended existing models by also considering chemisorption, knock-on yield, ion implantation, redeposition and reaction rate, leading to far more accurate but also far more complex models [25, 27, 56, 57, 36, 6].

According to [48] Monte Carlo type simulation codes, which have been proven to be the most effective, have the following major components:

1. the cross-section data for all processes being considered,
2. the algorithms used for the particle transport,
3. the methods used for geometry representation and to handle particle passage through volumes,
4. the methods to determine quantities of interest and analysis of the information obtained during the simulation.

Such an example is TRIM, which is a Monte Carlo computer program that calculates the interactions of energetic ions with amorphous targets (see Fig. 1). SRIM is a collection of software packages that calculate many features of the transport of ions in matter (see Fig. 2). Both programs were developed and are maintained by James F. Ziegler and can be downloaded from http://www.srim.org/.

Other MC based simulation packages are TRIDYN [44, 45], KALYPSO [32] and ACAT [68]. A comprehensive compilation and review of many other simulation codes for surface-ion interactions can be found in Chapter 1.2 of [26].

2.3 Modeling crystal formation on the substrate surface

Modeling the growth of the deposited layer on the substrate is one of the most complex aspects of modeling the sputtering process. The set of particles arriving at the substrate include not only outsputtered particles from the target, but also electrons, gas ions and neutrals [26]. The incoming particles either settle at the surface by forming chemical bonds depending on the sticking coefficient of the atomic pair in discussion, or they migrate on the surface until

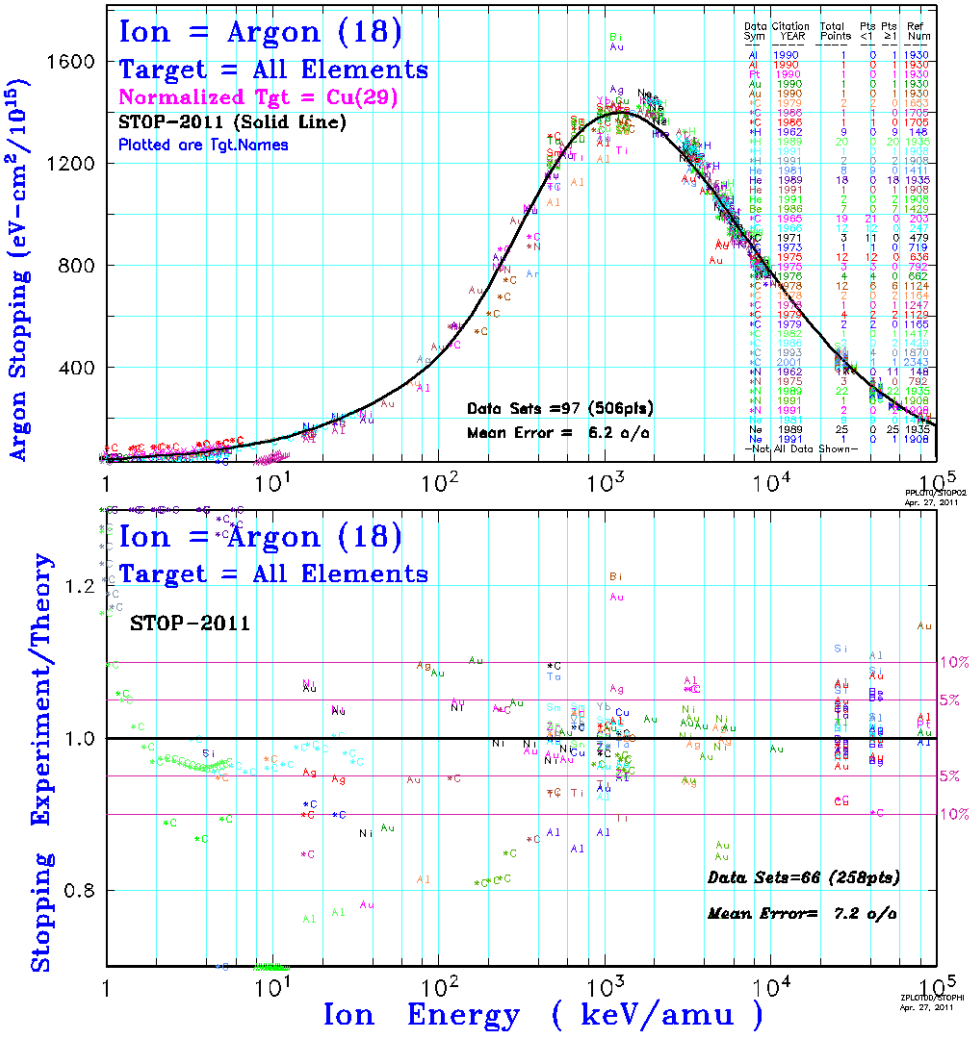


Figure 1: Application example of the TRIM program. Stopping range as a function of accelerated Argon ion energies using different targets; comparison of historically measured and simulated data (from http://www.srim.org/).

they reach a large enough crystal lump where they can take part in crystal formation on the grain boundary [30]. Because of this, the formation of a large area, even layer is not too common, rather ragged microstructures are formed [47]. The modeling of such microstructures is essential for example in the fab-

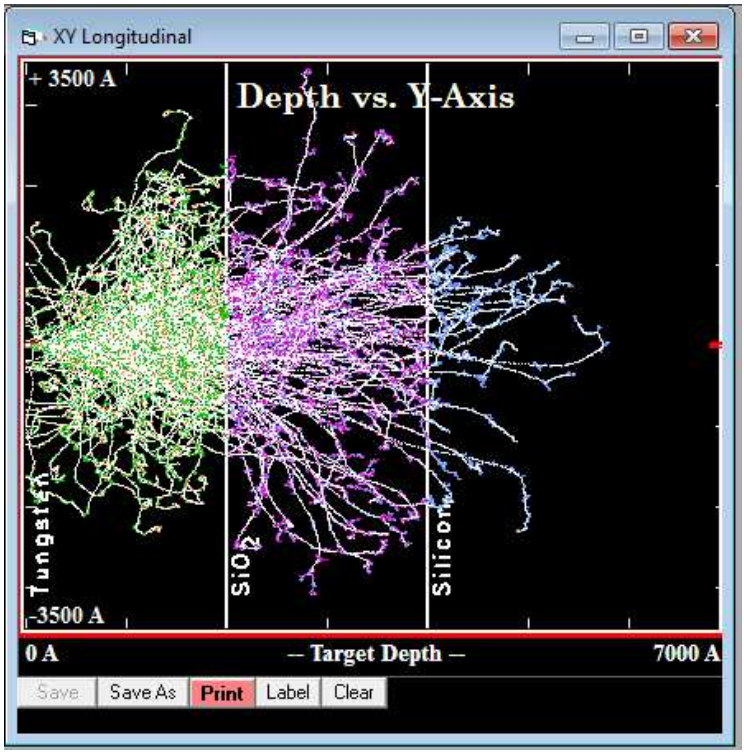


Figure 2: Application example of the SRIM program. Simulation of implantation depth of Boron ions into a multilayered $W/SiO_2/Silicon$ target using SRIM-2013.00 software (from http://www.srim.org/).

rication of microelectronics, where one may want to fill a narrow trench or via. In [7] the formation of some crystal structures is predicted based on operating conditions.

An example software is SiMTra, which is a binary collision Monte Carlo program (see Fig. 3) that allows the user to simulate the transport of sputtered particles through the gas phase flux during sputtering.

2.4 Macroscopic modeling of the reactive sputtering process

Similarly to those described in the previous sections, it is obvious that one must find a balance between complex accurate modeling and simple straightforward usability when developing a model. Berg et al. developed a model that found a balance between simplifications whilst still being accurate enough to

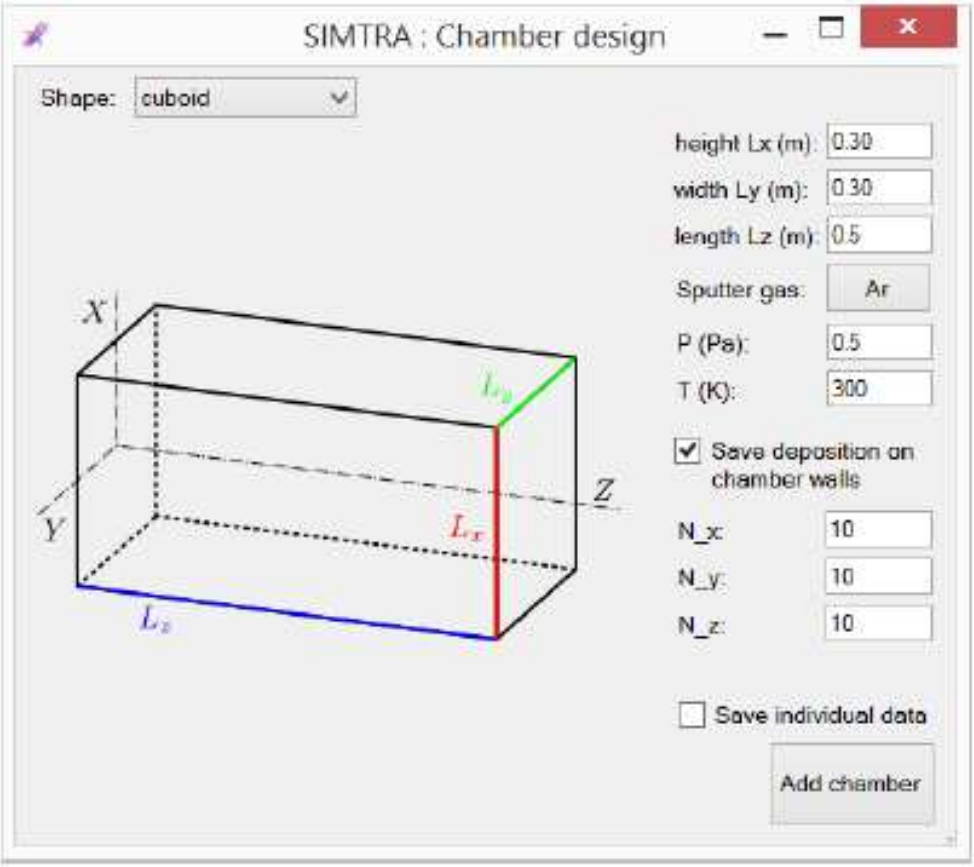


Figure 3: An input dialog screen of the SIMTRA software package used to configure the simulation setup (from http://www.draft.ugent.be/).

predict outcomes of experiments [3, 5, 4]. This model is commonly referred to in the literature as "Berg's Model". The model was constructed by focusing on the equilibrium equations, studying independent contributions to different quantities used to describe the process (see Figure 4). The main area where this model proved to be more successful than others is in the modeling of the hysteresis behavior. There have been a number of models [31, 1, 54] developed before Berg's Model that had problems reproducing the hysteresis behavior.

Berg employed variables θ_i to denote the coverage of the surfaces of interest (target and substrate) with compound molecules (the normalized amount of reacted surface area). The coverage takes values in the interval of $[0, 1]$, where

a value of 1 denotes a fully covered surface. This model is a monolayer surface model which only considers a chemisorption mechanism on the target (and substrate) surfaces. No redeposition can be considered.

The equation that defines the balance between supply and consumption of the reactive gas is given in Eq. (1), with the definitions of gas consumption on the surface of the target Q_t in Eq. (2), gas consumption on the surface of the substrate Q_c in Eq. (3) and of pumping flow Q_p in Eq. (4). The supply and usage of reactive gas is also presented in Fig. 4. The sticking coefficients are denoted with α, the flux of reactive gas atoms with F and the corresponding areas of the studied surfaces with A. S represents the pumping speed while P represents the partial pressure.

$$Q_{tot} = Q_t + Q_c + Q_p \tag{1}$$

$$Q_t = \alpha F(1 - \theta_t)A_t \tag{2}$$

$$Q_c = \alpha F(1 - \theta_c)A_c \tag{3}$$

$$Q_p = SP \tag{4}$$

The balance equations describing the interactions on the target and substrate surface are given in Eq. (5) and (6), respectively. The outsputtered compound F_c and metallic F_m flux is given in Eq. (7) and (8), respectively. Y denotes the sputtering yield, q is the electron charge and J denotes the ion current density of the background gas. Equations (1-8) have been presented in this form in [26].

$$\frac{J}{q}Y_c\theta_t = \alpha 2F(1 - \theta_t) \tag{5}$$

$$F_c(1 - \theta_c) + 2Q_c = F_m\theta_c \tag{6}$$

$$F_c = \frac{J}{q}Y_c\theta_t A_t \tag{7}$$

$$F_m = \frac{J}{q}Y_m(1 - \theta_t)A_t \tag{8}$$

Due to the reactive gas poisoning of the target, a process that tends to reduce the sputtering yield, a hysteresis can be observed due to the positive

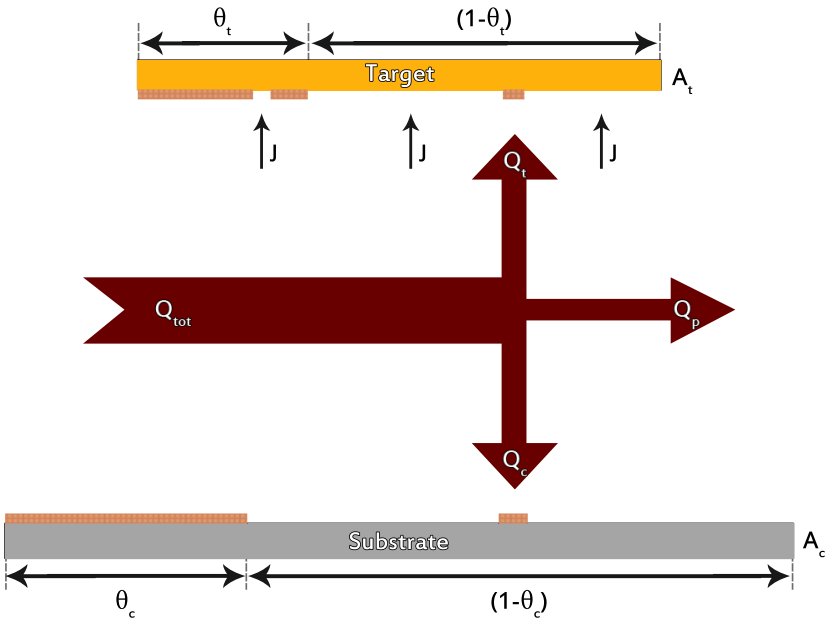


Figure 4: The supply and consumption of reactive gas during sputtering [5].

feedback between the partial pressure of the reactive gas and the target surface coverage (see Fig. 5). Since the publication of the Berg model, several extensions based on equilibrium equations have been developed in order to include modeling of multiple targets, multiple reactive gases, ion implantation, particle redeposition, multiple deposited layers, deposition of multiple compounds, spatial resolution and dynamics of phenomena, etc.

Given that many times the formation of complex films is required, which in turn require more than one reactive gas, Carlsson et al. further developed the model to work with two reactive gases as inputs, though only considering binary compounds as result of the interaction processes [15].

The research group named DRAFT, led by Diederik Depla, made their work of more than a decade available free of charge as a downloadable package from their university website http://www.draft.ugent.be/. The package contains their model named RSD2013 and it also comes with a detailed PDF guide to help the researcher quickly get started. The package also includes a piece of software called SIMTRA, which aids the simulation of deposition profiles. This model is an extension of the Berg model, having the most im-

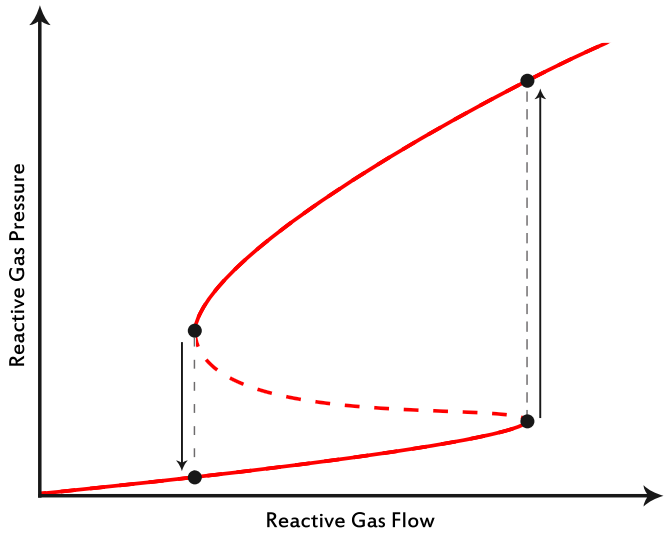


Figure 5: The observed hysteresis behavior of the system on a p-q plot [5].

portant extra feature of subsurface implantation of reactive gas ions/atoms and the compound forming 2nd order reaction mechanism in the subsurface. A chemisorption mechanism governs the reaction on the surface and redeposition of sputtered material back on the target is taken into account [58].

3 Modeling tasks and challenges

Throughout the following subsections a comparative study analysis is given under each modeling task.

3.1 Parameter identification approaches

One of the biggest, if not the biggest challenge in modeling reactive magnetron sputtering is the lack of readily available parameters or tools to obtain those parameters.

The lack of identified sticking coefficients led Bogaerts et al. [13] to attempt to identify them by trying to select the ones that would have had resulted in correct estimation of the process, after the experiment was undertaken. The chosen experimental setup was one, in which the sticking coefficient played

a significant role. The ion density at a certain distance from the substrate was measured and simulated. They performed the simulation for every possible value of the sticking coefficient and later checked which ones were in accordance with the measured data. What they found is that two given circumstances resulted in different parameters, meaning that the model wasn't detailed enough and some other effect also had to be considered. Their result, however, narrowed the plausible range for the sought parameter.

Leroy et al. attempted the parameter identification by analyzing samples of deposited films and estimating the actual content of a given material in them [37, 38]. Having this estimate, it was possible to calculate what the missing parameter should have had been. An extensive guide has been given on how this should be undertaken. The process requires several specialized equipment and techniques (XPS - X-ray Photoelectron Spectroscopy, EPMA - Electron-probe microanalysis) and complex calculations to be able to reach a result.

Strijckmans et al. also undertook this challenge of parameter identification in [56]. They used their most detailed RSD2013 model, missing only 3 parameters (see Fig. 6), and wrote a parallel C++ program to run the model on a supercomputer, to see which parameter sets would be usable.

The parameter fitting algorithm has two main components: a parameter set selection component and a parameter set evaluation component. A 2-D representation of the parameter set selection component can be seen in Fig. 7. Increasing the dimensionality of the parameter space is feasible, though with increased computational load. Three ordered lists have been employed to differentiate between unfinished (U), finished (F) and rejected (R) parameter sets. A parameter set is considered rejected (R) when the result of the model simulation does not agree with the measured data.

The evaluation component has an acceptation criterion which is determined by the PQ - pressure-flow relation of the simulation result. The critical flow values (the limits of the unstable region) are observed at three different current values and the maximum of the six squared errors is considered.

Before this study there was complete disagreement among researchers, many of them having measured completely different values for the same parameter. The results of this attempt further narrowed down the possible range of parameter sets.

3.2 Modeling the produced compounds

Berg et al. laid down the foundation for macroscopic modeling of magnetron sputtering with one reactive gas [3, 4, 5]. When the need arose to model binary or ternary compounds this model has been expanded.

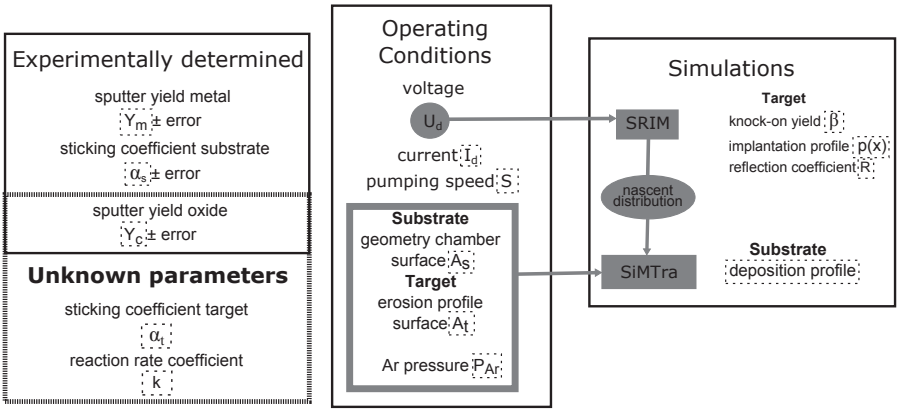


Figure 6: A scheme visualizing the input parameters of the RSD2013 model [58]

Carlsson et al. published an article where two reactive gases were used and binary compounds were modeled [15]. This model neglected ternary compounds for the sake of simplicity. When the growth of a ternary compound such as of TiOxNy is the requirement, one must find a different model.

Kelemen et al. published a macroscopic model that studies the formation of ternary compounds on the substrate [33]. The model follows the same approach as the two mentioned above, allocating 8 regions on the substrate covered with different active, passive, binary and ternary compounds.

3.3 Modeling different magnetron types

Dual magnetron co-sputtering allows the deposition of alloy material without having to manufacture alloy targets [55, 43, 34]. Multicomponent models have also been developed and analyzed in [17, 18, 46]. This setup requires two separate magnetrons to be installed, each of them becoming the anode of the other during sputtering. Such a setup requires a power supply that can generate pulsed bipolar signals. Given the fact, that each magnetron is switched between anode and cathode roles, the typical charge build-up on poisoned areas is neutralized due to sequential target bombing with ions as well as electrons. The absence of local charge build-up also leads to an arc free operation, which is mandatory for consistent large area, even layer deposition.

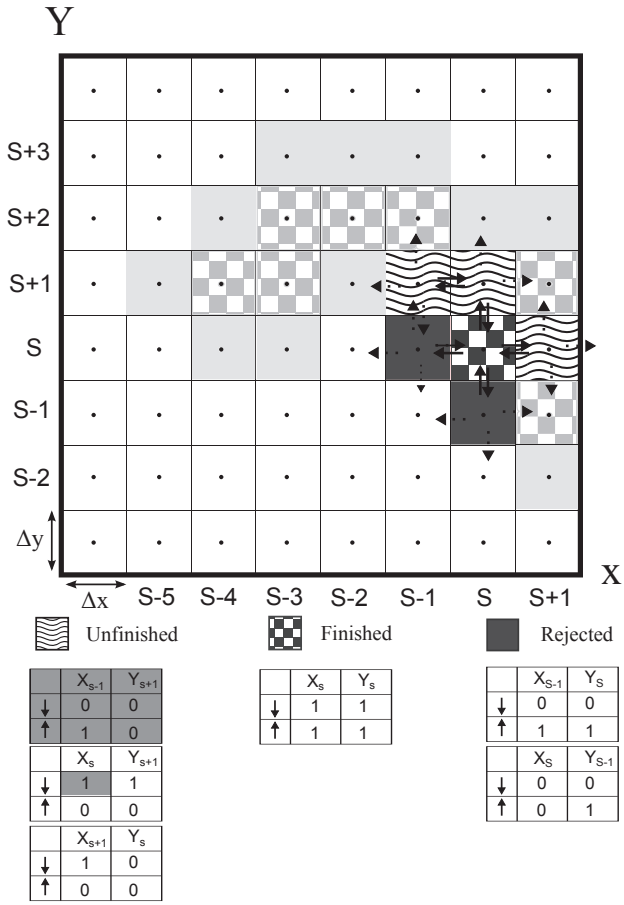


Figure 7: A 2-D illustration of the parameter selecting component of the searching algorithm [58].

Cylindrical rotating magnetron sputtering was introduced to reduce the operational costs of large scale industrial coaters, due to the inherent disadvantages of sputtering of stationary planar magnetrons. One of these disadvantages is the well-known formation of a so-called "race track", a high erosion zone on the surface of the target, that is caused by the inhomogeneous magnetic field that "shapes" the plasma above the target. By rotating the target with respect to the magnetic field no high erosion zone is formed. This leads

to a very efficient process due to the uniform wear the target experiences. The uniform wear maximizes target usability by avoiding the formation of a localized deep race track. The effects of rotation have been modeled and the influences of rotation speed have been studied in [27, 24, 42, 39]. It was observed that the rotation speed highly influences the shape and location of the hysteresis curve.

3.4 Modeling for control structure synthesis

The study of the system dynamics and the control structure synthesis require the development of time dependent, dynamic models. Based on macroscopic models, several authors have developed time dependent models, usually in state-space formulation [33, 57, 19, 20, 21, 2]. When derived from the original Berg model, the dynamic model represents a control-affine system of third order with linear and bilinear terms [62]. Woelfel et al. also presented the Berg model in a state-space form along with a detailed explanation of its construction (Eq. (9) and (10)). Coefficients $a_{i,j}$ and $b_{i,j}$ include parameters like sticking coefficients, sputtering efficiencies, sputtering current density, gas constant, temperature, etc. The state variables are the partial pressure of the reactive gas and the coverages of the target and of the substrate, while the input of the system is the input flow rate of the reactive gas.

$$\begin{pmatrix} x1 \\ x2 \\ x3 \end{pmatrix} = \begin{pmatrix} P \\ \theta_t \\ \theta_s \end{pmatrix} \tag{9}$$

$$\begin{aligned} \dot{x_1}(t) &= x_1(t)(-a_{1,1} + a_{1,2}x_2(t) + a_{1,3}x_3(t)) + bu(t) \\ \dot{x_2}(t) &= a_{2,12}x_1(t)(1 - x_2(t)) - a_{2,2}x_2(t) \\ \dot{x_3}(t) &= a_{3,13}x_1(t)(1 - x_3(t)) + a_{3,23}x_2(t)(1 - x_3(t)) - a_{3,32}x_3(t)(1 - x_2(t)) \end{aligned} \tag{10}$$

The magnetron sputtering system is essentially nonlinear, but several studies attempt to use linearization around the equilibrium points to study the stability and controllability of the system. Due to the difficulties to apply the results of control theory, there have been studies and even patents on how to control a system of such dynamics through heuristic synthesis of complex control structures. In [59] a control structure was presented in which the chosen manipulated quantities influence the controlled variable indirectly, through a chain of effects in the system.

Woelfel had a different approach by studying the essential input-output interactions in different operating conditions in order to develop a simplified model that was conceived with the aim of being practical for control purposes [62, 63, 64, 67]. Woelfel et al. presented a control design method that is based on artificial neural networks and ordinary differential equations [66, 65]. The goal of using neural networks was to decouple the plasma, electrical and gas subsystems from each other. Two multilayer perceptron type networks have been used, both employing a hidden layer of 10 neurons with sigmoid activation functions.

Our previous work also concentrated on control structure synthesis [40]. It started with the assumption that an on-line state identification algorithm can be developed, and with its use, the stoichiometry of the process can be controlled. The input-output characteristics have been plotted to observe the location of the peak of the substrate coverage with the desired compound (see Fig. 8). By selecting a preferred subsurface by formulating first order equations to delimit a monotone region (see Fig. 9) it was possible to conceive a state-dependent state limiting algorithm that can operate the system with two independent continuously pumped reactive gas inputs. This came as a proposed alternative for RGPP - Reactive Gas Pulsed Process [16, 41] which for the sake of stability had to compromise on stoichiometry.

One would be mistaken by thinking that better control of the process can only be achieved by ever more complicated control structures. Some researchers focused on model reduction, in order to facilitate the implementation of less complex control algorithms. The work done by Woelfel et al. concentrates on developing models that suffice the control goal [67]. It was mentioned that most results in the field were provided from either a vacuum science or a thin solid film approach, therefore most models developed were not usable for control purposes. It was emphasized that not enough work has been done to identify the input-output behavior of the reactive sputtering process.

They recognized [62] that for control purposes the Berg model can be reduced to the Abel differential equation (see Eq. (11)). In Eq. (11) the system input $u(t)$ is the gas flow and the system output $y(t)$ is the sputtering voltage.

$$\dot{y}(t) = Ay^3(t) + By^2(t) + Cy(t) + D + Eu(t) \tag{11}$$

Using this model it was later demonstrated that a venerable PID (proportional–integral–derivative) type controller can be used to control and stabilize the complex system. Detailed description has been given for the necessary preconditions that are needed to apply the given tuning rules [64].

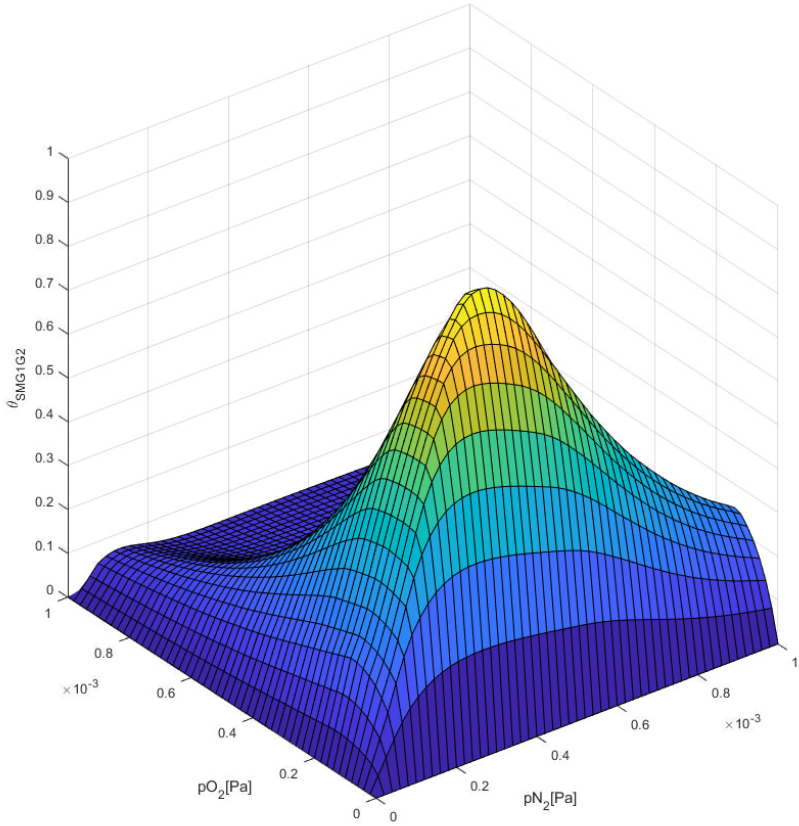


Figure 8: Substrate coverage with $\theta_{MG_1G_2}$ at $I = 0.1[A]$ for different reactive partial pressure combinations [40].

In [62], the parameter identification of the reduced model in Eq. (11) has been presented. The identification process begins with the construction of a matrix using the inputs and outputs of the system (see Eq. (12)). According to this approach, the parameters are identified from measured samples of the system input, system output and of the derivative of the system output.

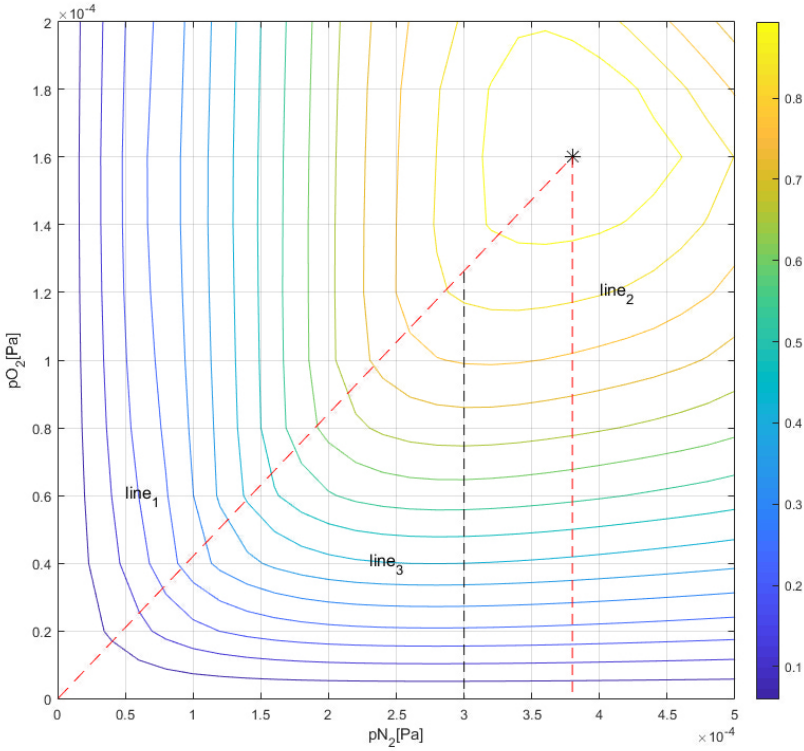


Figure 9: The contour plot of the substrate coverage and the lines used to delimit the preferred area [40].

$$\underbrace{\begin{pmatrix} y_{m1}^3 & y_{m1}^2 & y_{m1} & 1 & u_{m1} \\ y_{m2}^3 & y_{m2}^2 & y_{m2} & 1 & u_{m2} \\ y_{m3}^3 & y_{m3}^2 & y_{m3} & 1 & u_{m3} \\ y_{m4}^3 & y_{m4}^2 & y_{m4} & 1 & u_{m4} \\ y_{m5}^3 & y_{m5}^2 & y_{m5} & 1 & u_{m5} \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix}}_{M} \underbrace{\begin{pmatrix} A \\ B \\ C \\ D \\ E \end{pmatrix}}_{n} = \underbrace{\begin{pmatrix} \dot{y}_{m1} \\ \dot{y}_{m2} \\ \dot{y}_{m3} \\ \dot{y}_{m4} \\ \dot{y}_{m5} \\ \vdots \end{pmatrix}}_{r} \tag{12}$$

Calculating the Moore-Penrose type left pseudoinverse matrix gives the parameter vector n of the system under examination (see Eq. (13)). The parameter vector obtained by Eq. (13) can be used to calculate the parameters of

the original state-space system ($a_{i,j}$). Equations (11-13) have been presented in this form in [67].

$$n = M^{+} r \tag{13}$$

4 Conclusion and future challenges

In 2019, Depla et al. published an article on the current opportunities and challenges in modeling reactive magnetron sputtering [28]. They mentioned four areas that still need considerable attention:

1. redeposition of sputtered atoms,
2. studying I-V characteristics,
3. sample rotation,
4. pulsing discharge currents.

In our opinion, in future research on this topic emphasis should and probably will be placed on the following key points:

1. delimitation of ranges for some parameters in macroscopic models with the aid of the results obtained from detailed kinematic modeling,
2. precise identification of parameters with the help of the macroscopic dynamic models using measureable data such as partial pressures, gas flows, substrate temperatures, spectrum intensities, voltages and current values,
3. the use of online parameter identification algorithms to identify real-time substrate coverage and composition, mainly for control purposes,
4. development of macroscopic models that, from the perspective of process control, adequately integrate the models used to describe phenomena on the surfaces of both target and substrate as well as models used to describe the state of the plasma discharge.

Acknowledgements

This work has been partially funded by the EFOP-3.4.4-16-2017-00019 grant entitled "STEM Fejlesztések az Óbudai Egyetemen", which was aimed at increasing interest for STEM among students at the University of Óbuda, Budapest Hungary.

The authors gratefully acknowledge the support of Koen Strijckmans, of Gent University, for providing original figures from his work in order to facilitate conveying the surveyed ideas.

References

[1] T. Abe,T. Yamashina. The deposition rate of metallic thin films in the reactive sputtering process. *Thin Solid Films*, **30,** 1 (1975) 19–27. ⇒120

[2] Z. Ahmad, B. Abdallah. Controllability analysis of reactive magnetron sputtering process. *Acta Physica Polonica, A.*, **123,** 1 2013. ⇒127

[3] S. Berg, H.-O. Blom, T. Larsson, C. Nender. Modeling of reactive sputtering of compound materials. *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films*, **5,** 2 (1987) 202–207. ⇒120, 124

[4] S. Berg, C. Nender. Modeling of mass transport and gas kinetics of the reactive sputtering process. *Le Journal de Physique IV*, 5 (C5):C5–45, 1995. ⇒120, 124

[5] S. Berg, T. Nyberg, H.-O. Blom, C. Nender. Computer modeling as a tool to predict deposition rate and film composition in the reactive sputtering process. *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films*, 16, 3 (1998) 1277–1285. ⇒120, 122, 123, 124

[6] S. Berg, T. Nyberg. Fundamental understanding and modeling of reactive sputtering processes. *Thin solid films*, **476,** 2 (2005) 215–230. ⇒117

[7] M.-M.-M. Bilek, D.-R. McKenzie. Predicting the structure of plasma deposited materials. *Czechoslovak Journal of Physics* **52** (2002) 905–920. ⇒119

[8] C.-K. Birdsall, A.-B. Langdon. *Plasma Physics Via Computer Simulation, Bristol, UK.* IOP Publishing, 1991. ⇒116

[9] A. Bogaerts, M. van Straaten, R. Gijbels. Monte Carlo simulation of an analytical glow discharge: motion of electrons, ions and fast neutrals in the cathode dark space. *Spectrochimica Acta Part B: Atomic Spectroscopy*, **50,** 2 (1995) 179–196. ⇒115, 116

[10] A. Bogaerts, R. Gijbels, W.-J. Goedheer. Hybrid Monte Carlo-fluid model of a direct current glow discharge. *Journal of Applied Physics*, **78,** 4 (1995) 2233–2241. ⇒116

[11] A. Bogaerts, M. van Straaten, R. Gijbels. Description of the thermalization process of the sputtered atoms in a glow discharge using a three-dimensional Monte Carlo method. *Journal of applied physics*, **77,** 5 (1995) 1868–1874. ⇒116

[12] A. Bogaerts, R. Gijbels, W.-J. Goedheer. Two-dimensional model of a direct current glow discharge: Description of the electrons, argon ions, and fast argon atoms. *Analytical Chemistry*, **68,** 14 (1996) 2296–2303. ⇒116

[13] A. Bogaerts, J. Naylor, M. Hatcher, W.-J. Jones, R. Mason. Influence of sticking coefficients on the behavior of sputtered atoms in an argon glow discharge: Modeling and comparison with experiment. *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films*, **16,** 4 (1998) 2400–2410. ⇒123

[14] J.-W. Bradley. The plasma properties adjacent to the target in a magnetron sputtering source. *Plasma sources science and technology*, **5,** 4 (1996) 622. ⇒116

[15] P. Carlsson, C. Nender, H. Barankova, S. Berg. Reactive sputtering using two reactive gases, experiments and computer modeling. *Journal of Vacuum Science*

& Technology A: Vacuum, Surfaces, and Films, **11,** 4 (1993) 1534–1539. ⇒122, 125

[16] J.-M. Chappé, N. Martin, J. Lintymer, F. Sthal, G. Terwagne, J. Takadoum. Titanium oxynitride thin films sputter deposited by the reactive gas pulsing process. *Applied Surface Science*, **253,** 12 (2007) 5312–5316. ⇒128

[17] D.-J. Christie, W.-D. Sproul, D. Carter. Mid-frequency dual magnetron reactive co-sputtering for deposition of customized index optical films. In *Society of Vacuum Coaters 46 th Annual Technical Conference*, 2003 pp. 393–398. ⇒125

[18] D.-J. Christie. *Power conversion and control for pulsed magnetron reactive sputtering.* PhD thesis, Colorado State University, 2004. ⇒125

[19] D.-J. Christie. Making magnetron sputtering work: Modelling reactive sputtering dynamics, part 1. *SVC Bulletin*, 2014. pp. 24–27. ⇒127

[20] D.-J. Christie. Making magnetron sputtering work: Modelling reactive sputtering dynamics, part 2. *SVC Bulletin*, 2015, pp. 30–33. ⇒127

[21] D.-J. Christie. Making magnetron sputtering work: Modelling reactive sputtering dynamics, part 3. *SVC Bulletin*, 2015, pp. 38–41. ⇒127

[22] C. Costin, L. Marques, G. Popa, G. Gousset. Two-dimensional fluid approach to the dc magnetron discharge. *Plasma Sources Science and Technology*, **14,** 1 (2005) 168. ⇒116

[23] N.-F. Cramer. Analysis of a one-dimensional, steady-state magnetron discharge. *Journal of Physics D: Applied Physics*, **30,** 18 (1997) 2573. ⇒116

[24] D. Depla, J. Haemers, G. Buyle, R. De Gryse. Hysteresis behavior during reactive magnetron sputtering of Al2O3 using a rotating cylindrical magnetron. *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films*, **24,** 4 (2006) 934–938. ⇒127

[25] D. Depla, S. Heirwegh, S. Mahieu, R. De Gryse. Towards a more complete model for reactive magnetron sputtering. *Journal of Physics D: Applied Physics*, 40, 7 (2007) 1957. ⇒117

[26] D. Depla, S. Mahieu, et al. *Reactive Sputter Deposition, volume 109.* Springer, 2008. ⇒116, 117, 121

[27] D. Depla, X.-Y. Li, S. Mahieu, K.-V. Aeken, W.-P. Leroy, J. Haemers, R. De Gryse, A. Bogaerts. Rotating cylindrical magnetron sputtering: Simulation of the reactive process. *Journal of Applied Physics*, **107,** 11 (2010) 113307. ⇒117, 127

[28] D. Depla, K. Strijckmans, A. Dulmaa, F. Cougnon, R. Dedoncker, R. Schelfhout, I. Schramm, F. Moens, R. De Gryse. Modeling reactive magnetron sputtering: Opportunities and challenges. *Thin Solid Films*, 2019. ⇒131

[29] J. Goree, T.-E. Sheridan. Magnetic field dependence of sputtering magnetron efficiency. *Applied physics letters*, **59,** 9 (1991) 1052–1054. ⇒116

[30] T. Hammerschmidt, A. Kersch, P. Vogl. Embedded atom simulations of titanium systems with grain boundaries. *Physical Review B*, **71,** 20 (2005) 205409. ⇒118

[31] J. Heller. Reactive sputtering of metals in oxidizing atmospheres. *Thin Solid Films*, 17, 2 (1973) 163–176. ⇒120

[32] M.-A. Karolewski. Kalypso: a software package for molecular dynamics simulation of atomic collisions at surfaces. *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, 230, 1-4 (2005) 402–405. ⇒117

[33] A. Kelemen, D. Biró, A.-Zs. Fekete, L. Jakab-Farkas, R.-R. Madarász. Macroscopic thin film deposition model for the two-reactive-gas sputtering process. *Acta Universitatis Sapientiae Electrical and Mechanical Engineering*, **8** (2016) 62–78. ⇒125, 127

[34] S. Kikkawa, M. Fujiki, M. Takahashi, and F. Kanamaru. Reactive co-sputter deposition and succesive annealing of fe-al-n thin film. *Journal of the Japan Society of Powder and Powder Metallurgy*, **44,** 7 (1997) 674–677. ⇒125

[35] R.-L. Kinder, M.-J. Kushner. Wave propagation and power deposition in magnetically enhanced inductively coupled and helicon plasma sources. *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films*, **19,** 1 (2001) 76–86. ⇒116

[36] T. Kubart, O. Kappertz, T. Nyberg, S. Berg. Dynamic behaviour of the reactive sputtering process. *Thin Solid Films*, **515,** 2 (2006) 421–424. ⇒117

[37] W.-P. Leroy, S. Mahieu, R. Persoons, D. Depla. Method to determine the sticking coefficient of o2 on deposited al during reactive magnetron sputtering, using mass spectrometry. *Plasma Processes and Polymers*, 51, 6 (2009) S342–S346. ⇒124

[38] W.-P. Leroy, S. Mahieu, R. Persoons, D. Depla. Quantification of the incorporation coefficient of a reactive gas on a metallic film during magnetron sputtering: The method and results. *Thin Solid Films*, 518, 5 (2009) 1527–1531. ⇒124

[39] W.-P. Leroy, S. Mahieu, D. Depla, A.-P. Ehiasarian. High power impulse magnetron sputtering using a rotating cylindrical magnetron. *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films*, **28,** 1 (2010) 108–111. ⇒127

[40] R.-R. Madarász and A. Kelemen. Stoichiometry control of the two gas reactive sputtering process. In *2019 IEEE 19th International Symposium on Computational Intelligence and Informatics and 7th IEEE International Conference on Recent Achievements in Mechatronics, Automation, Computer Sciences and Robotics (CINTI-MACRo)*, pp. 000217–000222. IEEE, 2019. ⇒128, 129, 130

[41] N. Martin, R. Sanjines, J. Takadoum, F. Lévy. Enhanced sputtering of titanium oxide, nitride and oxynitride thin films by the reactive gas pulsing technique. *Surface and Coatings Technology*, **142** (2001) 615–620. ⇒128

[42] H.-E. McKelvey. Rotatable sputtering apparatus, May 1 1984. US Patent 4,445,997. ⇒127

[43] C. Misiano and E. Simonetti. 4.4 co-sputtered optical films. *Vacuum*, 27, 4 (1977) 403–406. ⇒125

[44] W. Möller, W. Eckstein, J.-P. Biersack. Tridyn-binary collision simulation of atomic collisions and dynamic composition changes in solids. *Computer Physics Communications*, **51,** 3 (1988) 355–368. ⇒117

[45] W. Möller, M. Posselt. *TRIDYN_FZR User Manual.* FZR Dresden, 2001. ⇒117

[46] M. Moradi, C. Nender, S. Berg, H.-O. Blom, A. Belkind, Z. Orban. Modeling of multicomponent reactive sputtering. *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films*, **9,** 3 (1991) 619–624. ⇒125

[47] E. Penilla, J. Wang. Pressure and temperature effects on stoichiometry and microstructure of nitrogen-rich tin thin films synthesized via reactive magnetron dc-sputtering. *Journal of Nanomaterials*, 2008. ⇒118

[48] L.-M. Popescu. A computer code package for Monte Carlo photon-electron transport simulation: Comparisons with experimental benchmarks. *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, **161** (2000) 318–322. ⇒117

[49] I.-A. Porokhova, Y.-B. Golubovskii, J. Bretagne, M. Tichy, J.-F. Behnke. Kinetic simulation model of magnetron discharges. *Physical Review E*, **63,** 5 (2001) 056408. ⇒116

[50] I.A. Porokhova, Y.-B. Golubovskii, J.-F. Behnke. Anisotropy of the electron component in a cylindrical magnetron discharge. i. theory of the multiterm analysis. *Physical Review E*, **71,** 6 (2005) 066406. ⇒116

[51] I.-A. Porokhova, Y.-B. Golubovskii, J.-F. Behnke. Anisotropy of the electron component in a cylindrical magnetron discharge. ii. application to real magnetron discharge. *Physical review E*, **71,** 6 (2005) 066407. ⇒116

[52] R.-K. Porteous, D.-B. Graves. Modeling and simulation of magnetically confined low-pressure plasmas in two dimensions. *IEEE transactions on plasma science*, **19,** 2 (1991) 204–213. ⇒116

[53] T.-E. Sheridan, M.-J. Goeckner, J. Goree. Model of energetic electron transport in magnetron discharges. *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films*, 8, 1 (1990) 30–37. ⇒116

[54] F. Shinoki, A. Itoh. Mechanism of rf reactive sputtering. *Journal of Applied Physics*, 46, 8 (1975) 3381–3384. ⇒120

[55] W.-D. Sproul, D.-J. Christie, D.-C. Carter. Control of reactive sputtering processes. *Thin solid films*, **491,** 1-2 (2005) 1–17. ⇒125

[56] K. Strijckmans, W.-P. Leroy, R. De Gryse, D. Depla. Modeling reactive magnetron sputtering: Fixing the parameter set. *Surface and Coatings Technology*, 206, 17 (2012) 3666–3675. ⇒117, 124

[57] K. Strijckmans and D. Depla. A time-dependent model for reactive sputter deposition. *Journal of Physics D: Applied Physics*, 37, 23 (2014) 235302. ⇒117, 127

[58] K. Strijckmans. *Modeling the reactive magnetron sputtering process.* PhD Thesis, Ghent University, 2015. ⇒114, 123, 125, 126

[59] R. Terry, K. Gibbons, S. Zarrabian. Method and apparatus for reactive sputtering employing two control loops, August 22 2000. US Patent 6,106,676. ⇒ 127

[60] A.-E. Wendt, M.-A. Lieberman, H. Meuth. Radial current distribution at a plan ar magnetron cathode. *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films*, 6, 3 (1988) 1827–1831. ⇒116

[61] A.-E. Wendt, M.-A. Lieberman. Spatial structure of a planar magnetron discharge. *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films*, **8,** 2 (1990) 902–907. ⇒116

[62] C. Woelfel, P. Awakowicz, J. Lunze. Model reduction and identification of nonlinear reactive sputter processes. *IFAC-PapersOnLine*, 50, 1 (2017) 13728–13734. ⇒115, 127, 128, 129

[63] C. Woelfel, P. Awakowicz, J. Lunze. Robust high-gain control of nonlinear reactive sputter processes. In *2017 IEEE Conference on Control Technology and Applications (CCTA)*, IEEE, 2017, pp 25–30. ⇒128

[64] C. Woelfel, P. Awakowicz, J. Lunze. Tuning rule for linear control of nonlinear reactive sputter processes. In *2017 21st International Conference on Process Control (PC)*, IEEE, 2017, pp. 109–114. ⇒128

[65] C. Woelfel, S. Kockmann, P. Awakowicz, J. Lunze. Model identification of nonlinear sputter processes. In *2017 17th International Conference on Control, Automation and Systems (ICCAS)*, IEEE, 2017, pp. 182–187. ⇒128

[66] C. Woelfel, S. Kockmann, P. Awakowicz, J. Lunze. Neural network based linearization and control of sputter processes. In *2017 11th Asian Control Conference (ASCC)*, IEEE, 2017, pp. 2831–2836. ⇒128

[67] C. Woelfel, D. Bockhorn, P. Awakowicz, J. Lunze. Model approximation and stabilization of reactive sputter processes. *Journal of Process Control*, 2018. ⇒ 128, 131

[68] Y. Yamamura and M. Ishida. Monte Carlo simulation of the thermalization of sputtered atoms and reflected atoms in the magnetron sputtering discharge. *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films*, 13, 1 (1995) 101–112. ⇒117

Received: May 3, 2020 • Revised: June 8, 2020

Acta Univ. Sapientiae, Informatica **12,** 1 (2020) 137–157

DOI: 10.2478/ausi-2020-0009

$\mathscr{P}$-energy of graphs

Prajakta Bharat JOSHI
Department of Mathematics,
CHRIST(Deemed to be University),
Bangalore, India.
email:
prajaktabharat.joshi@res.christuniversity.in

Mayamma JOSEPH
Department of Mathematics,
CHRIST(Deemed to be University),
Bangalore,
India.
email:
mayamma.joseph@christuniversity.in

Abstract. Given a graph $G = (V, E)$, with respect to a vertex partition $\mathscr{P}$ we associate a matrix called $\mathscr{P}$-matrix and define the $\mathscr{P}$-energy, $E_{\mathscr{P}}(G)$ as the sum of $\mathscr{P}$-eigenvalues of $\mathscr{P}$-matrix of G. Apart from studying some properties of $\mathscr{P}$-matrix, its eigenvalues and obtaining bounds of $\mathscr{P}$-energy, we explore the robust(shear) $\mathscr{P}$-energy which is the maximum(minimum) value of $\mathscr{P}$-energy for some families of graphs. Further, we derive explicit formulas for $E_{\mathscr{P}}(G)$ of few classes of graphs with different vertex partitions.

1 Introduction

In this paper, we are concerned with simple and undirected graph $G = (V, E)$ of order n and size m. For spectral and graph theoretic terminologies we refer Cvetković et al. and West respectively [4, 14].

If $A(G)$ is the adjacency matrix of a graph G, then its *energy* is the sum of the absolute values of all the eigenvalues of $A(G)$ [5]. In 1978, Gutman introduced this concept and thereafter, extensive studies on the same have

Computing Classification System 1998: G.2.2
Mathematics Subject Classification 2010: 05C50, 05C69
Key words and phrases: graph energy, vertex partitions, k-partition matrix, k-partition energy

been carried out by several researchers on its theoretical as well as practical aspects and several variations of graph energy can be found in the literature [2, 6, 7, 12].

An interesting variation of graph energy is *the* k*-partition energy* defined by Sampathkumar et al. [12]. They have introduced this concept using the idea of a matrix called L-*matrix*, $P_k(G)$ with respect to a vertex partition P_k that uniquely represents the given graph G. *The* k*-partition energy,* $E_{P_k}(G)$ *is sum of the absolute values of* k*-partition eigenvalues of* $P_k(G)$. For a given graph G, the value of $E_{P_k}(G)$ varies according to different vertex partitions. It can be observed that the properties of elements in the vertex partition is not taken into consideration while determining the value of $E_{P_k}(G)$. In the present study, we consider this aspect and introduce $\mathscr{P}$-energy as a variation of k-partition energy.

Let $\mathscr{P} = \{V_1, V_2, \ldots, V_k\}$ be a partition of the vertex set $V(G)$ of a graph $G = (V, E)$. Then the $\mathscr{P}$*-matrix of* G, $A_{\mathscr{P}}(G) = D(G) + P_k(G)$, *where* $D(G)$ *is the diagonal matrix with the* i^{th} *diagonal entry, the cardinality of the set* $V_r \in \mathscr{P}$ *containing the vertex* v_i. In other words, $A_{\mathscr{P}}(G) = (a_{ij})_{n\times n}$ where

$$a_{ij} = \begin{cases} |V_r| & \text{if } i = j \text{ and } v_i = v_j \in V_r, \text{ for } r = 1, 2, \ldots k \\ 2 & \text{if } v_iv_j \in E(G) \text{ with } v_i, v_j \in V_r, \\ 1 & \text{if } v_iv_j \in E(G) \text{ with } v_i \in V_r \text{ and } v_j \in V_s \text{ for } r \neq s, \\ -1 & \text{if } v_iv_j \notin E(G) \text{ with } v_i, v_j \in V_r, \\ 0 & \text{otherwise.} \end{cases}$$

The characteristic polynomial of $A_{\mathscr{P}}(G)$ is denoted by $\phi_{\mathscr{P}}(G, \lambda)$ and *the eigenvalues* $\lambda_1, \lambda_2, \ldots, \lambda_n$ *of* $A_{\mathscr{P}}(G)$ *are called* $\mathscr{P}$*-eigenvalues.* If $g_1, g_2, \ldots, g_n$ are the multiplicities of $\lambda_1 > \lambda_2 > \ldots > \lambda_n$ respectively, then the $\mathscr{P}$-spectrum of G is

$$\mathrm{Spec}_{\mathscr{P}}(G) = \{\lambda_1^{g_1}, \lambda_2^{g_2}, \ldots, \lambda_n^{g_n}\}$$

and accordingly *the* $\mathscr{P}$*-energy,* $E_{\mathscr{P}}(G)$ *is sum of the absolute values of* $\mathscr{P}$*-eigenvalues of* $A_{\mathscr{P}}(G)$.

For a given vertex partition $\mathscr{P}$ of $V(G)$, the diagonal entries of $A_{\mathscr{P}}(G)$ are positive numbers, whereas the diagonal entries of $P_k(G)$ are zeros and remaining entries of these matrices are same which belongs to the set $\{2, 1, 0, -1\}$. Since the absolute values of the eigenvalues of any matrix are directly proportional to the maximum value of the absolute values of entries in the given matrix, one immediate observation is that if the cardinality of every mem-

ber of the given vertex partition $\mathscr{P}$ of a graph G is greater than 1, then $E_{\mathscr{P}}(G) \geq E_{P_k}(G)$.

Another interesting observation about $\mathscr{P}$-energy is that, as the order of vertex partition $\mathscr{P}$ of a given graph G increases, value of $E_{\mathscr{P}}(G)$ decreases. Hence, $\mathscr{P}$-energy of a graph G is maximum when the vertex partition $\mathscr{P} = \{V(G)\}$ and minimum when $\mathscr{P} = \{\{v_1\},\{v_2\},\ldots,\{v_n\}\}$. We call the maximum (minimum) value of $\mathscr{P}$-energy as the *robust(shear) $\mathscr{P}$-energy*, $E_{\mathscr{P}_r}(G)(E_{\mathscr{P}_s}(G))$, similar to the concepts of robust domination energy and shear domination energy of a graph introduced by Acharya et al. [1].

Example 1 *For a null graph H of order n, it can easily be verified that $E_{\mathscr{P}_r}(H) = n^2$ and $E_{\mathscr{P}_s}(H) = n$.*

Now, we state in the following remark some of the basic results from linear algebra which are required for the present study:

Remark 2 [11] *If A is a real or complex matrix of order $n \times n$ with the characteristic polynomial $\phi(G,\lambda)$ and eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$, then*

(i) A principal sub-matrix of order $r \times r$ of A is a sub-matrix consisting of the same set of r rows and r columns and a principal minor of order $r \times r$ of A is the determinant of a principal sub-matrix of order $r \times r$.

(ii) If $a_0, a_1, a_2, \ldots, a_n$ are the coefficients of $\phi(G,\lambda)$, then $(-1)^r a_r$ is the sum of principal minors of order $r \times r$.

(iii) If the rth symmetric function $S_r(A)$ is the sum of the product of the eigenvalues of A taken r at a time, then it is the sum of $r \times r$ principal minors of A.

(iv) Trace of A is the sum of diagonal entries of A and it can also be represented as $tr(A) = S_1(A) = -a_1$.

(v) $\prod_{i=1}^{n} \lambda_i = |A|$.

Theorem 3 [10] *If λ is an eigenvalue of the matrix $(a_{ij})_{n\times n}$, then*

$$|\lambda| \leq n \max_{i,j} |a_{ij}|.$$

Lemma 4 [4] *If $C = \begin{pmatrix} A & B \\ B & A \end{pmatrix}$ is a symmetric block matrix of order 2×2, then the spectrum of C is the union of the spectra of $A + B$ and $A - B$.*

Lemma 5 [4] *If M, N, P, Q are matrices where M is invertible and $S = \begin{pmatrix} M & N \\ P & Q \end{pmatrix}$, then $\det S = \det M \cdot \det[Q - PM^{-1}N]$.*

2 Properties of $\mathscr{P}$-eigenvalues of $A_{\mathscr{P}}(G)$

Before proceeding further, we present a few observations about $A_{\mathscr{P}}(G)$ with respect to the structure of a graph G.

Observation 1 *Given a graph $G = (V, E)$, the following are true for its $\mathscr{P}$-matrix $A_{\mathscr{P}}(G)$.*

(i) If $d(v_i)$ is the degree of $v_i \in V(G)$, then $d(v_i)$ is the number of positive off-diagonal entries of the ith row corresponding to the vertex v_i in $A_{\mathscr{P}}(G)$.

(ii) The elements of $A_{\mathscr{P}}(G)$, excluding its main diagonal entries has one-one correspondence with $P_k(G)$ with respect to the same vertex partition of a given graph G.

(iii) For the matrix $A_{\mathscr{P}}(G)$,

$$tr(A_{\mathscr{P}}(G)) = \sum_{i=1}^{n} a_{ii} = \sum_{i=1}^{k} |V_i|^2. \tag{1}$$

(iv) If m_1 is the number of edges of a graph G whose end vertices share the same partition, m_2 is the number of edges of G whose end vertices are in different partitions and m_3 is the number of pairs of non-adjacent vertices of G within the same partition, then

$$\sum_{1 \le i < j \le n} (a_{ij})^2 = 4m_1 + m_2 + m_3. \tag{2}$$

The following result that characterizes $\mathscr{P}$-matrix of a graph is similar to that of the characterization of L- matrix of a labeled graph as given in [13].

Theorem 6 *A symmetric matrix $A = (a_{ij})_{n \times n}$ with positive diagonal entries and off-diagonal entries belonging to the set {2, 1, 0, -1} is the $\mathscr{P}$-matrix graph G of order n with the vertex partition $\mathscr{P} = \{V_1, V_2, \ldots, V_k\}$ if and only if*

(i) $a_{ij}, a_{jk} \in \{2, -1\} \implies a_{ik} \in \{2, -1\}$,

(ii) $a_{ij} \in \{2, -1\}$ *and* $a_{jk} \in \{0, 1\} \implies a_{ik} \in \{0, 1\}$ *and*

(iii) $v_i \in V_r \implies a_{ii} = |V_r|$.

In the next result, we obtain the exact values of the coefficients of λ^n, λ^{n-1} and λ^{n-2} in the characteristic polynomial $\phi_{\mathscr{P}}(G,\lambda) = a_0\lambda^n + a_1\lambda^{n-1} + \ldots + a_{n-1}\lambda_1 + a_n$ of $A_{\mathscr{P}}(G)$.

Proposition 7 *If* G *is a graph with vertex partition* $\mathscr{P} = \{V_1, V_2, \ldots V_k\}$ *and* $\phi_{\mathscr{P}}(G,\lambda) = a_0\lambda^n + a_1\lambda^{n-1} + \ldots + a_n$*, then*

(i) $a_0 = 1$

(ii) $a_1 = -\sum_{i=1}^{k} |V_i|^2$

(iii) $a_2 = \sum_{1\le i<j\le k} |V_i||V_j| - (4m_1 + m_2 + m_3)$.

Proof.

(i) It holds directly, as the characteristic polynomial $\phi_{\mathscr{P}}(G,\lambda)$ is a monic polynomial.

(ii) From Remark 2(ii) and Equation (1), we get the result.

(iii) From Remark 2(ii),

$$\begin{aligned}(-1)^2 a_2 &= \sum_{1\le i<j\le n} \begin{vmatrix} a_{ii} & a_{ij} \\ a_{ji} & a_{jj} \end{vmatrix} \\ &= \sum_{1\le i<j\le n} a_{ii}a_{jj} - \sum_{1\le i<j\le n} a_{ij}a_{ji}.\end{aligned}$$

Since $A_{\mathscr{P}}(G)$ is a symmetric matrix,

$$\begin{aligned} a_2 &= \sum_{1\le i<j\le n} a_{ii}a_{jj} - \sum_{1\le i<j\le n} (a_{ij})^2 \\ &= \sum_{1\le i<j\le k} |V_i||V_j| - \sum_{1\le i<j\le n} (a_{ij})^2. \end{aligned} \tag{3}$$

Therefore from Equations (2) and (3), we obtain the result.

□

The trace of a matrix is the sum of the eigenvalues of that matrix, therefore the sum of $\mathscr{P}$-eigenvalues of $A_{\mathscr{P}}(G)$ of a graph G is non-zero. In the next proposition, we obtain its value in terms of cardinality of elements in the vertex partition $\mathscr{P} = \{V_1, V_2, \ldots V_k\}$ of G.

Proposition 8 *If* G *is a graph with the vertex partition* $\mathscr{P}$ *and* $\lambda_1, \lambda_2, \ldots, \lambda_n$ *are the* $\mathscr{P}$*-eigenvalues, then*

(i) $\sum_{i=1}^{n} \lambda_i = \sum_{i=1}^{k} |V_i|^2$

(ii) $\sum_{i=1}^{n} \lambda_i^2 = \sum_{i=1}^{k} |V_i|^3 + 8m_1 + 2m_2 + 2m_3.$

Proof.

(i) From Remark 2(iv) and Equation (1), the result holds.

(ii) For a matrix A of order $n \times n$, $tr(A^2) = (tr(A))^2 - 2S_2(A)$, where $S_2(A)$ is the 2nd symmetric function. This can be written as

$$\sum_{i=1}^{n} \lambda_i^2 = (\sum_{i=1}^{n} a_{ii})^2 - 2S_2(A).$$

By Remark 2(iii),

$$\begin{aligned}
\sum_{i=1}^{n} \lambda_i^2 &= (\sum_{i=1}^{n} a_{ii})^2 - 2\sum_{i<j}(a_{ii}a_{jj} - a_{ij}a_{ji}) \\
&= (\sum_{i=1}^{n} a_{ii})^2 - 2\sum_{i<j} a_{ii}a_{jj} + 2\sum_{i<j}(a_{ij})^2 \\
&= \sum_{i=1}^{n} a_{ii}^2 + 2\sum_{i<j} a_{ii}a_{jj} - 2\sum_{i<j} a_{ii}a_{jj} + 2\sum_{i<j}(a_{ij})^2 \\
&= \sum_{i=1}^{n} a_{ii}^2 + 2\sum_{i<j}(a_{ij})^2 \qquad (4)
\end{aligned}$$

and

$$\begin{aligned}
\sum_{i=1}^{n} a_{ii}^2 &= |V_1| \cdot |V_1|^2 + |V_2| \cdot |V_2|^2 + \ldots + |V_k| \cdot |V_k|^2 \\
&= |V_1|^3 + |V_2|^3 + \ldots + |V_k|^3 \\
&= \sum_{i=1}^{k} |V_i|^3. \qquad (5)
\end{aligned}$$

Therefore from Equations (2), (4) and (5),

$$\sum_{i=1}^{n} \lambda_i^2 = \sum_{i=1}^{k} |V_i|^3 + 2(4m_1 + m_2 + m_3). \tag{6}$$

□

The next proposition given without proof, follows from Cauchy-Schwartz inequality and Proposition 8 (ii). Note that, the symbols m_1, m_2, m_3 for graph G_1 are as given in the Observation 1(iv) and m_1', m_2', m_3' are the corresponding values for the graph G_2.

Proposition 9 *Let G_1 and G_2 be two graphs with respect to vertex partition $\mathscr{P} = \{V_1, V_2, \ldots, V_k\}$ and $\mathscr{P}' = \{V_1', V_2', \ldots, V_k'\}$ respectively. If $\{\lambda_1, \lambda_2, \ldots, \lambda_n\}$ and $\{\lambda_1', \lambda_2', \ldots, \lambda_n'\}$ are the $\mathscr{P}$-eigenvalues of $\mathscr{P}$-matrix of G_1 and G_2 respectively, then*

$$\sum_{i=1}^{n} \lambda_i \lambda_i' \leq \sqrt{\left[\sum_{i=1}^{k} |V_i|^3 + 8m_1 + 2m_2 + 2m_3\right]\left[\sum_{i=1}^{k} |V_i'|^3 + 8m_1' + 2m_2' + 2m_3'\right]}.$$

3 Bounds for $\mathscr{P}$-energy

Now we present some bounds for the $\mathscr{P}$-energy of a graph G in terms of its order and the cardinality of elements in its vertex partition. One obvious bound when $\mathscr{P} = \{V_1, V_2, \ldots V_k\}$ is

$$\sum_{i=1}^{k} |V_i|^2 \leq E_{\mathscr{P}}(G) \leq n^3.$$

The lower bound follows from the inequality $\sum_{i=1}^{n} \lambda_i \leq \sum_{i=1}^{n} |\lambda_i|$ whereas the upper bound is a direct deduction from Theorem 3.

Theorem 10 *For any graph G with vertex partition $\mathscr{P} = \{V_1, V_2, \ldots V_k\}$,*

$$E_{\mathscr{P}}(G) \leq \sqrt{n\left\{\sum_{i=1}^{k} |V_i|^3 + 8m_1 + 2m_2 + 2m_3\right\}}. \tag{7}$$

Proof. By Cauchy-Schwartz inequality,

$$\Big(\sum_{i=1}^{n} a_i b_i\Big)^2 \leq \big(\sum_{i=1}^{n} a_i^2\big)\big(\sum_{i=1}^{n} b_i^2\big). \tag{8}$$

Replace $a_i = 1$ and $b_i = |\lambda_i|$ in Equation (8),

$$\begin{aligned}\Big(\sum_{i=1}^{n} |\lambda_i|\Big)^2 &\leq \big(\sum_{i=1}^{n} 1\big)\big(\sum_{i=1}^{n} |\lambda_i|^2\big)\\ &\leq n\sum_{i=1}^{n} \lambda_i^2.\end{aligned}$$

From Equation (6),

$$\Big(\sum_{i=1}^{n} |\lambda_i|\Big)^2 \leq n\Big\{\sum_{i=1}^{k} |V_i|^3 + 8m_1 + 2m_2 + 2m_3\Big\}.$$

Hence,

$$E_{\mathscr{P}}(G) \leq \sqrt{n\Big\{\sum_{i=1}^{k} |V_i|^3 + 8m_1 + 2m_2 + 2m_3\Big\}}.$$

□

Theorem 11 *Let G be a graph with vertex partition $\mathscr{P} = \{V_1, V_2, \ldots V_k\}$ and $|A_{\mathscr{P}}(G)|$ be the determinant of $A_{\mathscr{P}}(G)$. Then*

$$E_{\mathscr{P}}(G) \geq \sqrt{\sum_{i=1}^{k} |V_i|^3 + 8m_1 + 2m_2 + 2m_3 + n(n-1)|A_{\mathscr{P}}(G)|^{2/n}}. \tag{9}$$

Proof. By the definition of $\mathscr{P}$-energy,

$$\begin{aligned}\big[E_{\mathscr{P}}(G)\big]^2 = \big(\sum_{i=1}^{n} |\lambda_i|\big)^2 &= \big(\sum_{i=1}^{n} |\lambda_i|\big)\big(\sum_{j=1}^{n} |\lambda_j|\big)\\ &= \sum_{i=1}^{n} |\lambda_i|^2 + \sum_{i\neq j} |\lambda_i||\lambda_j|. \qquad (10)\end{aligned}$$

By using arithmetic and geometric mean inequality, Equation (10) can be written as follows

$$\begin{aligned}\left[E_{\mathscr{P}}(G)\right]^2 &\geq \sum_{i=1}^{n} |\lambda_i|^2 + n(n-1)\left(\prod_{i\neq j} |\lambda_i||\lambda_j|\right)^{\frac{1}{n(n-1)}} \\ &\geq \sum_{i=1}^{n} |\lambda_i|^2 + n(n-1)\left(\prod_{i=1}^{n} |\lambda_i|^{2(n-1)}\right)^{\frac{1}{n(n-1)}}.\end{aligned}$$

Therefore,

$$\left[E_{\mathscr{P}}(G)\right]^2 \geq \sum_{i=1}^{n} |\lambda_i|^2 + n(n-1)\left(\prod_{i=1}^{n} |\lambda_i|\right)^{\frac{2}{n}}.$$

Hence from Remark 2(v) and Equation (6),

$$E_{\mathscr{P}}(G) \geq \sqrt{\sum_{i=1}^{k} |V_i|^3 + 8m_1 + 2m_2 + 2m_3 + n(n-1)|A_{\mathscr{P}}(G)|^{2/n}}.$$

□

Remark 12 *Let* H *be a null graph of order* n*. Then the upper and lower bounds given by Equations* (7) *and* (9) *are sharp for the vertex partition* $\mathscr{P} = \{\{v_1\}, \{v_2\}, \ldots, \{v_n\}\}$ *of* H*.*

4 $\mathscr{P}$-energy of some graph families

In this section, we examine the $\mathscr{P}$-energy of some families of graphs for the trivial partitions $\mathscr{P} = \{V(G)\}$ and $\mathscr{P} = \{\{v_1\}, \{v_2\}, \ldots, \{v_n\}\}$ respectively. Recall that the extreme values of $\mathscr{P}$-energy is obtained with respect to these partitions and the largest value of $E_{\mathscr{P}}(G)$ denoted by $E_{\mathscr{P}_r}(G)$ is referred to as the robust $\mathscr{P}$-energy and the smallest denoted by $E_{\mathscr{P}_s}(G)$ is referred to as the shear $\mathscr{P}$-energy.

Theorem 13 *For the complete graph* K_n*,*

$$E_{\mathscr{P}_r}(K_n) = n^2 \text{ and } E_{\mathscr{P}_s}(K_n) = n.$$

Proof. Let K_n be a complete graph and $\mathscr{P} = \{V(G)\}$. The $\mathscr{P}$-matrix of K_n is

$$A_{\mathscr{P}}(K_n) = [(n-2)I + 2J]_{n\times n}$$

where J is the matrix of order $n \times n$ whose all entries are 1 and I is identity matrix of order $n \times n$. The characteristic polynomial is $\phi_{\mathscr{P}}(K_n, \lambda) = |\lambda I - A_{\mathscr{P}}(K_n)|$. Thus,

$$\begin{aligned}\phi_{\mathscr{P}}(K_n, \lambda) &= \begin{vmatrix} \lambda - n & -2 & -2 & \dots & -2 \\ -2 & \lambda - n & -2 & \dots & -2 \\ -2 & -2 & \lambda - n & \dots & -2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -2 & -2 & -2 & \dots & \lambda - n \end{vmatrix}_{n\times n} \\ &= [\lambda - (n-2)]^{(n-1)}[\lambda - (3n-2)].\end{aligned}$$

Therefore,

$$\mathrm{Spec}_{\mathscr{P}}(K_n) = \{(3n-2)^1, (n-2)^{(n-1)}\}$$

and

$$E_{\mathscr{P}}(K_n) = n^2 \text{ with respect to the vertex partition } \mathscr{P} = \{V(G)\}.$$

Hence, $E_{\mathscr{P}_r}(K_n) = n^2$. Now, let $\mathscr{P}$ be a vertex partition of K_n such that $\mathscr{P} = \{\{v_1\}, \{v_2\}, \dots, \{v_n\}\}$. The $\mathscr{P}$-matrix of K_n is

$$A_{\mathscr{P}}(K_n) = J_{n\times n}$$

and

$$\begin{aligned}\phi_{\mathscr{P}}(K_n, \lambda) &= \begin{vmatrix} \lambda - 1 & -1 & -1 & \dots & -1 \\ -1 & \lambda - 1 & -1 & \dots & -1 \\ -1 & -1 & \lambda - 1 & \dots & -1 \\ \vdots & \vdots & \vdots & \ddots & -1 \\ -1 & -1 & -1 & \dots & \lambda - 1 \end{vmatrix}_{n\times n} \\ &= \lambda^{(n-1)}(\lambda - n).\end{aligned}$$

Therefore,

$$\mathrm{Spec}_{\mathscr{P}}(K_n) = \{n^1, 0^{(n-1)}\}$$

and

$$E_{\mathscr{P}}(K_n) = n \text{ with respect to the vertex partition } \mathscr{P} = \{\{v_1\}, \{v_2\}, \dots, \{v_n\}\}.$$

Hence, $E_{\mathscr{P}_s}(K_n) = n$. □

Remark 14 *For a complete graph* K_n *and a null graph* H*,*

$$E_{\mathscr{P}_r}(K_n) = E_{\mathscr{P}_r}(H) \text{ and } E_{\mathscr{P}_s}(K_n) = E_{\mathscr{P}_s}(H).$$

Note that K_n *and* H *are non-cospectral equi-*$\mathscr{P}$*-energetic graphs, since the* $\mathscr{P}$*-eigenvalues of* $\mathscr{P}$*-matrices of both the graphs differ but the values of their robust and shear* $\mathscr{P}$*-energies coincide.*

The following result deals with the robust and shear $\mathscr{P}$-energy of a star. We omit its proof, since its proof technique is similar to that of Theorem 13.

Theorem 15 *If* $K_{1,n-1}$ *is a star of order* $n \geq 2$*, then*

$$E_{\mathscr{P}_r}(K_{1,n-1}) = 2n - 4 + \sqrt{n^2 + 12n - 12} \text{ and}$$
$$E_{\mathscr{P}_s}(K_{1,n-1}) = (n-2) + 2\sqrt{n-1}.$$

If we join the maximum degree vertex of two copies of $K_{1,r-1}$ of order $r (r \geq 2)$, then the resultant graph is called a double star $B_{r,r}$ of order $n = 2r$.

Theorem 16 *If* $B_{r,r}$ *is a double star of order* n*, then*

$$E_{\mathscr{P}_r}(B_{r,r}) = n^2 \quad \text{for } n \geq 2,$$
$$E_{\mathscr{P}_s}(B_{r,r}) = \begin{cases} n - 1 + \sqrt{2n-3} & \text{for } 2 \leq n < 8, \\ (n-4) + 2\sqrt{2n-3} & \text{for } n \geq 8. \end{cases}$$

Proof. Let $B_{r,r}$ be a double star of order n with respect to the vertex partition $\mathscr{P} = \{V(G)\}$. Then

$$A_{\mathscr{P}}(B_{r,r}) = \begin{pmatrix} n & 2 & 2 & \dots & 2 & 2 & -1 & -1 & \dots & -1 \\ 2 & n & -1 & \dots & -1 & -1 & -1 & -1 & \dots & -1 \\ 2 & -1 & n & \dots & -1 & -1 & -1 & -1 & \dots & -1 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 2 & -1 & -1 & \dots & n & -1 & -1 & -1 & \dots & -1 \\ 2 & -1 & -1 & \dots & -1 & n & 2 & 2 & \dots & 2 \\ -1 & -1 & -1 & \dots & -1 & 2 & n & -1 & \dots & -1 \\ -1 & -1 & -1 & \dots & -1 & 2 & -1 & n & \dots & -1 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ -1 & -1 & -1 & \dots & -1 & 2 & -1 & -1 & \dots & n \end{pmatrix}_{n \times n}.$$

Clearly, it is of the form $\left(\begin{smallmatrix} A & B \\ B & A \end{smallmatrix}\right)$. To get $\mathrm{Spec}_{\mathscr{P}}(B_{r,r})$, we need to find $\mathrm{Spec}_{\mathscr{P}}(A+B)$ and $\mathrm{Spec}_{\mathscr{P}}(A-B)$ by solving its respective characteristic polynomials.

By applying a series of row and column operations on $\phi_{\mathscr{P}}(A+B,\lambda)$ and $\phi_{\mathscr{P}}(A-B,\lambda)$, and using Lemma 5,

$$\begin{aligned}\phi_{\mathscr{P}}(A+B,\lambda) =&[\lambda-(n+1)]^{(r-2)}\left[\lambda-\frac{(n+5)+\sqrt{n^2+12n-3}}{2}\right]\\ &\left[\lambda-\frac{(n+5)-\sqrt{n^2+12n-3}}{2}\right]\end{aligned} \tag{11}$$

and

$$\begin{aligned}\phi_{\mathscr{P}}(A-B,\lambda) =&[\lambda-(n+1)]^{(r-2)}\left[\lambda-\frac{(2n-1)+3\sqrt{2n-3}}{2}\right]\\ &\left[\lambda-\frac{(2n-1)-3\sqrt{2n-3}}{2}\right].\end{aligned} \tag{12}$$

Therefore from Equations (11) and (12), and by the Lemma 4,

$$\begin{aligned}\mathrm{Spec}_{\mathscr{P}}(B_{r,r}) = \Bigg\{&\left[\frac{(n+5)+\sqrt{n^2+12n-3}}{2}\right]^1, \left[\frac{(2n-1)+3\sqrt{2n-3}}{2}\right]^1,\\ &(n+1)^{(n-4)}, \left[\frac{(2n-1)-3\sqrt{2n-3}}{2}\right]^1,\\ &\left[\frac{(n+5)-\sqrt{n^2+12n-3}}{2}\right]^1\Bigg\}.\end{aligned}$$

Hence, $E_{\mathscr{P}_r}(B_{r,r}) = n^2$, for $n \geq 2$. Now, we consider the vertex partition $\mathscr{P} = \{\{v_1\},\{v_2\},\ldots,\{v_n\}\}$ of $B_{r,r}$ and the corresponding $\mathscr{P}$-matrix of $B_{r,r}$ is

$$A_{\mathscr{P}}(B_{r,r}) = \begin{pmatrix} 1 & 1 & 1 & \ldots & 1 & 1 & 0 & 0 & \ldots & 0\\ 1 & 1 & 0 & \ldots & 0 & 0 & 0 & 0 & \ldots & 0\\ 1 & 0 & 1 & \ldots & 0 & 0 & 0 & 0 & \ldots & 0\\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots\\ 1 & 0 & 0 & \ldots & 1 & 0 & 0 & 0 & \ldots & 0\\ 1 & 0 & 0 & \ldots & 0 & 1 & 1 & 1 & \ldots & 1\\ 0 & 0 & 0 & \ldots & 0 & 1 & 1 & 0 & \ldots & 0\\ 0 & 0 & 0 & \ldots & 0 & 1 & 0 & 1 & \ldots & 0\\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & 0 & \ldots & 0 & 1 & 0 & 0 & \ldots & 1 \end{pmatrix}_{n\times n}.$$

It is of the form $\left(\begin{smallmatrix} A & B \\ B & A \end{smallmatrix}\right)$. Thus,

$$\phi_{\mathscr{P}}(A+B,\lambda) = (\lambda-1)^{(r-2)}\left[\lambda - \left(\frac{3 \pm \sqrt{4r-3}}{2}\right)\right] \tag{13}$$

and

$$\phi_{\mathscr{P}}(A-B,\lambda) = (\lambda-1)^{(r-2)}\left[\lambda - \left(\frac{1 \pm \sqrt{4r-3}}{2}\right)\right]. \tag{14}$$

Therefore from Lemma 4 and Equations (13) and (14),

$$\begin{aligned} \mathrm{Spec}_{\mathscr{P}}(B_{r,r}) = &\left\{\left(\frac{3+\sqrt{4r-3}}{2}\right)^1, \left(\frac{1+\sqrt{4r-3}}{2}\right)^1, \right. \\ &\left. \left(\frac{1-\sqrt{4r-3}}{2}\right)^1, \left(\frac{3-\sqrt{4r-3}}{2}\right)^1, 1^{(n-4)}\right\}. \end{aligned}$$

Hence, $E_{\mathscr{P}_s}(B_{r,r}) = n-1+\sqrt{2n-3}$, for $2 \le n < 8$
and
$E_{\mathscr{P}_s}(B_{r,r}) = (n-4)+2\sqrt{2n-3}$, for $n \ge 8$. □

Theorem 17 *If $K_{r,r}$ is a complete bipartite graph of order $n = 2r \ge 2$, then*

$$E_{\mathscr{P}_r}(K_{r,r}) = n^2+2n-2 \text{ and } E_{\mathscr{P}_s}(K_{r,r}) = 2(n-1).$$

Proof. Let $K_{r,r}$ be a complete bipartite graph of order n and let $\mathscr{P} = \{V(G)\}$. The $\mathscr{P}$-matrix of $K_{r,r}$ is a 2×2 block matrix which can be represented as $\left(\begin{smallmatrix} A & B \\ B & A \end{smallmatrix}\right)$.

$$A_{\mathscr{P}}(K_{r,r}) = \begin{pmatrix} [(n+1)I-J]_{r\times r} & [2J]_{r\times r} \\ [2J]_{r\times r} & [(n+1)I-J]_{r\times r} \end{pmatrix}.$$

Therefore, from Lemma 4 its $\mathscr{P}$-spectrum is given by

$$\mathrm{Spec}_{\mathscr{P}}(K_{r,r}) = \mathrm{Spec}_{\mathscr{P}}(A+B) \cup \mathrm{Spec}_{\mathscr{P}}(A-B). \tag{15}$$

By applying successive row and column operations on $\phi_{\mathscr{P}}(A+B,\lambda)$ and $\phi_{\mathscr{P}}(A-B,\lambda)$, and simplifying using Lemma 5, we get

$$\phi_{\mathscr{P}}(A+B,\lambda) = [\lambda-(3r+1)][\lambda-(n+1)]^{(r-1)} \tag{16}$$

and

$$\phi_{\mathscr{P}}(A-B,\lambda) = [\lambda+(r-1)][\lambda-(2n+1)][\lambda-(n+1)]^{(r-2)}. \tag{17}$$

Thus, from Equations (15), (16) and (17)

$$\phi_{\mathscr{P}}(K_{r,r},\lambda) = [\lambda+(r-1)][\lambda-(3r+1)][\lambda-(2n+1)][\lambda-(n+1)]^{(n-3)}.$$

Therefore,

$$\mathrm{Spec}_{\mathscr{P}}(K_{r,r}) = \{(2n+1)^1, (3r+1)^1, (n+1)^{(n-3)}, [-(r-1)]^1\}$$

and

$$E_{\mathscr{P}_r}(K_{r,r}) = n^2+2n-2.$$

Now, if we consider $\mathscr{P} = \{\{v_1\},\{v_2\},\ldots,\{v_n\}\}$ as the vertex partition of $K_{r,r}$, then the corresponding $\mathscr{P}$-energy will be shear $\mathscr{P}$-energy of G. So, consider $K_{r,r}$ with respect to $\mathscr{P} = \{\{v_1\},\{v_2\},\ldots,\{v_n\}\}$ and the $\mathscr{P}$-matrix of $K_{r,r}$ is $A_{\mathscr{P}}(K_{r,r}) = \left(\begin{smallmatrix} I & J \\ J & I \end{smallmatrix}\right)$. Therefore, from Lemma 4

$$\mathrm{Spec}_{\mathscr{P}}(K_{r,r}) = \mathrm{Spec}_{\mathscr{P}}(I+J) \cup \mathrm{Spec}_{\mathscr{P}}(I-J). \tag{18}$$

By applying successive row and column operations on $\phi_{\mathscr{P}}(A+B,\lambda)$ and $\phi_{\mathscr{P}}(A-B,\lambda)$, and simplifying using Lemma 5, we get the corresponding $\mathscr{P}$-eigenvalues. Thus,

$$\mathrm{Spec}_{\mathscr{P}}(A+B) = \{1^{(r-1)}, (r+1)^1\} \tag{19}$$

and

$$\mathrm{Spec}_{\mathscr{P}}(A-B) = \{1^{(r-1)}, [-(r-1)]^1\}. \tag{20}$$

Therefore, from Equations (18), (19) and (20)

$$\mathrm{Spec}_{\mathscr{P}}(K_{r,r}) = \{(r+1)^1, 1^{(n-2)}, [-(r-1)]^1\}$$

and

$$E_{\mathscr{P}_s}(K_{r,r}) = 2(n-1).$$

□

Remark 18 *We observe that, the robust $\mathscr{P}$-energy of $K_{r,r}$, 2-partition energy of $K_{1,n-1}$ and color energy of $K_{1,n-1}$ with respect to minimum number of colors χ are same, that is $E_{\mathscr{P}_s}(K_{r,r}) = E_{P_2}(K_{1,n-1}) = E_{\chi}(K_{1,n-1})$.*

Now, we proceed to determine $E_{\mathscr{P}}(G)$ for some families of graphs with respect to non-trivial vertex partitions.

Theorem 19 *For the star* $K_{1,n-1}, n \geq 3$, *with vertex partition* $\mathscr{P} = \{\{v_1\}, \{v_2, v_3, \ldots, v_n\}\}$ *where* v_1 *is the central vertex and* $v_2, v_3, \ldots, v_n$ *are pendant vertices of* $K_{1,n-1}$,

$$E_{\mathscr{P}}(K_{1,n-1}) = n(n-2) + 2\sqrt{n-1}.$$

Proof. The $\mathscr{P}$-matrix of $K_{1,n-1}$ is

$$A_{\mathscr{P}}(K_{1,n-1}) = \begin{pmatrix} 1 & 1 & 1 & \ldots & 1 \\ 1 & n-1 & -1 & \ldots & -1 \\ 1 & -1 & n-1 & \ldots & -1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & -1 & -1 & \ldots & n-1 \end{pmatrix}_{n\times n}.$$

Thus, the characteristic polynomial of $A_{\mathscr{P}}(K_{1,n-1})$ is

$$\phi_{\mathscr{P}}(K_{1,n-1}, \lambda) = (\lambda - n)^{(n-2)}[\lambda - (1 \pm \sqrt{(n-1)})].$$

Hence,

$$\mathrm{Spec}_{\mathscr{P}}(K_{1,n-1}) = \{[1 + \sqrt{(n-1)}]^1, [1 - \sqrt{(n-1)}]^1, n^{(n-2)}\}$$

and

$$\begin{aligned} E_{\mathscr{P}}(K_{1,n-1}) &= n(n-2) + |1 + \sqrt{n-1}| + |1 - \sqrt{n-1}| \\ &= n(n-2) + 2\sqrt{n-1}, \text{ for } n \geq 3. \end{aligned}$$

□

Now, we derive $\mathscr{P}$-energy of a double star $E_{\mathscr{P}}(B_{s,s})$ for different partitions and for that we consider, $V(B_{s,s}) = \{u_1, u_2, \ldots, u_s, v_1, v_2, \ldots, v_s\}$ such that u_1, v_1 are the maximum degree (central) vertices. Note that, the pendant vertices u_i's are attached to u_1 and the pendant vertices v_i's are attached to v_1, for $i = 2, 3, \ldots, s$.

Theorem 20 *If* $B_{s,s}$ *is a double star of order* $n \geq 6$ *with the vertex partition* $\mathscr{P} = \{\{u_1, v_1\}, \{u_2, u_3, \ldots, u_s, v_2, v_3, \ldots, v_s\}\}$ *where* u_1 *and* v_1 *are the central vertices, then*

$$E_{\mathscr{P}}(B_{s,s}) = \begin{cases} (n-4)(n-1) + \sqrt{n^2-3} + 5 & \textit{for } n = 6 \textit{ and } 8, \\ (n-4)(n-1) + \sqrt{n^2-3} + \frac{1}{2}(5 + \sqrt{2n+5}) & \textit{for } n = 10, \\ (n-4)(n-1) + \sqrt{n^2-3} + \sqrt{2n+5} & \textit{for } n \geq 12. \end{cases}$$

Proof. The $\mathscr{P}$-matrix of $B_{s,s}$ is a 2×2 block circulant matrix which can be represented as $\left(\begin{smallmatrix} A & B \\ B & A \end{smallmatrix}\right)$. Therefore, by Lemma 4 its spectrum is given by

$$\mathrm{Spec}_{\mathscr{P}}(B_{s,s}) = \mathrm{Spec}_{\mathscr{P}}(A+B) \cup \mathrm{Spec}_{\mathscr{P}}(A-B). \tag{21}$$

By applying successive row and column operations on $\phi_{\mathscr{P}}(A+B,\lambda)$ and $\phi_{\mathscr{P}}(A-B,\lambda)$, we get

$$\mathrm{Spec}_{\mathscr{P}}(A+B) = \left\{(n-1)^{(s-2)}, \left[\frac{5+\sqrt{2n+5}}{2}\right]^1, \left[\frac{5-\sqrt{2n+5}}{2}\right]^1\right\} \tag{22}$$

and

$$\mathrm{Spec}_{\mathscr{P}}(A-B) = \left\{(n-1)^{(s-2)}, \left[\frac{(n-1)+\sqrt{n^2-3}}{2}\right]^1, \left[\frac{(n-1)-\sqrt{n^2-3}}{2}\right]^1\right\} \tag{23}$$

respectively. Therefore, from Equations (21), (22) and (23)

$$\begin{aligned}\mathrm{Spec}_{\mathscr{P}}(B_{s,s}) =& \left\{(n-1)^{(n-4)}, \left[\frac{(n-1)+\sqrt{n^2-3}}{2}\right]^1, \left[\frac{5+\sqrt{2n+5}}{2}\right]^1,\right.\\ &\left.\left[\frac{5-\sqrt{2n+5}}{2}\right]^1, \left[\frac{(n-1)-\sqrt{n^2-3}}{2}\right]^1\right\}\end{aligned}$$

and

$$\begin{aligned}E_{\mathscr{P}}(B_{s,s}) =& (n-4)(n-1) + \left|\frac{5+\sqrt{2n+5}}{2}\right| + \left|\frac{5-\sqrt{2n+5}}{2}\right| \\ &+ \left|\frac{(n-1)+\sqrt{n^2-3}}{2}\right| + \left|\frac{(n-1)-\sqrt{n^2-3}}{2}\right|.\end{aligned}$$

Hence from this, the result follows. □

Next, we consider another partition $\mathscr{P}' = \{\{u_1, v_2, v_3, \ldots, v_s\}, \{v_1, u_2, u_3, \ldots, u_s\}\}$ of $V(B_{s,s})$ such that $\{u_1, v_2, v_3, \ldots, v_s\}$ and $\{v_1, u_2, u_3, \ldots, u_s\}$ are two independent sets where u_1, v_1 are the central vertices and $u_2, u_3, \ldots, u_s, v_2, v_3, \ldots, v_s$ are the pendent vertices of $B_{s,s}$.

Theorem 21 *Let $B_{s,s}$ be a double star of order $n \geq 6$ with the vertex partition $\mathscr{P}' = \{\{u_1, v_2, v_3, \ldots, v_s\}, \{v_1, u_2, u_3, \ldots, u_s\}\}$ where u_1, v_1 are the central vertices of $B_{s,s}$. Then*

$$E_{\mathscr{P}'}(B_{s,s}) = \frac{1}{2}\left[n^2 - 2n - 4 + \sqrt{n^2 + 20n - 28}\right].$$

Proof. The $\mathscr{P}$-matrix of $B_{s,s}$ is a 2×2 block circulant matrix which can be represented as $\left(\begin{smallmatrix} A & B \\ B & A \end{smallmatrix}\right)$. Therefore, its spectrum is given by

$$\mathrm{Spec}_{\mathscr{P}'}(B_{s,s}) = \mathrm{Spec}_{\mathscr{P}'}(A+B) \cup \mathrm{Spec}_{\mathscr{P}'}(A-B). \tag{24}$$

By applying successive row and column operations on $\phi_{\mathscr{P}}(A+B,\lambda)$ and $\phi_{\mathscr{P}}(A-B,\lambda)$, we get

$$\mathrm{Spec}_{\mathscr{P}'}(A+B) = \{(s+1)^{(s-2)}, 2^1\} \tag{25}$$

and

$$\begin{aligned}\mathrm{Spec}_{\mathscr{P}'}(A-B) =& \left\{ \left[\frac{(s+1)+\sqrt{s^2+10s-7}}{2}\right]^1, (s+1)^{(s-2)}, \right. \\ & \left. \left[\frac{(s+1)-\sqrt{s^2+10s-7}}{2}\right]^1 \right\}. \end{aligned} \tag{26}$$

Therefore, from Equations (24), (25) and (26)

$$\begin{aligned}\mathrm{Spec}_{\mathscr{P}'}(B_{s,s}) =& \left\{ \left[\frac{(s+1)+\sqrt{s^2+10s-7}}{2}\right]^1, (s+1)^{(n-4)}, \right. \\ & \left. \left[\frac{(s+1)-\sqrt{s^2+10s-7}}{2}\right]^1, 2^1 \right\}\end{aligned}$$

and

$$\begin{aligned}E_{\mathscr{P}'}(B_{s,s}) =& 2 + (s+1)(n-4) + \left|\frac{(s+1)+\sqrt{s^2+10s-7}}{2}\right| + \\ & \left|\frac{(s+1)-\sqrt{s^2+10s-7}}{2}\right|.\end{aligned}$$

On simplifying the above equation, we get the result. □

Another possibility of the vertex partition $\mathscr{P}$ having 2 elements for a $B_{s,s}$ is taking one copy of a star $K_{1,r-1}$ in each of the two elements of $\mathscr{P}$. The next result gives its corresponding $\mathscr{P}$-energy. We omit its proof as it is similar to the proofs of Theorems 20 and 21.

Theorem 22 *Let $B_{s,s}$ be a double star of order $n \geq 6$ with the vertex partition $\mathscr{P}'' = \{\{u_1, u_2, u_3, \ldots, u_s\}, \{v_1, v_2, v_3, \ldots, v_s\}\}$ where u_1, v_1 are the central vertices of $B_{s,s}$. Then*

$$E_{\mathscr{P}''}(B_{s,s}) = \frac{1}{2}\left[n^2 - 2n - 8 + \sqrt{n^2+20n-28} + \sqrt{n^2+28n-60}\right].$$

Remark 23 *From Theorems 20, 21 and 22, we observe that*

$$E_{\mathscr{P}} \geq E_{\mathscr{P}''} \geq E_{\mathscr{P}'}.$$

Theorem 24 *Let $K_{r,r}$ be a complete bipartite graph of order n with a vertex partition $\mathscr{P} = \{V_1, V_2\}$ such that V_1 and V_2 are two partite sets of $K_{r,r}$. Then*

$$E_{\mathscr{P}}(K_{r,r}) = \frac{1}{2}[n^2 + 2n - 4].$$

Proof. Let $V_1 = \{u_1, u_2, \ldots, u_r\}$ and $V_2 = \{v_1, v_2, \ldots, v_r\}$. The $\mathscr{P}$-matrix of $K_{r,r}$ is

$$A_{\mathscr{P}}(K_{r,r}) = \begin{pmatrix} [(r+1)I - J]_{r\times r} & J_{r\times r} \\ J_{r\times r} & [(r+1)I - J]_{r\times r} \end{pmatrix}_{n\times n}.$$

To get the $\mathscr{P}$-spectra of $K_{r\times r}$, by Lemma 4, it is sufficient to find $\mathscr{P}$-spectra of $[(r+1)I]_{r\times r}$ and $[(r+1)I - 2J]_{r\times r}$. Since $[(r+1)I]_{r\times r}$ is a diagonal matrix,

$$\mathrm{Spec}_{\mathscr{P}}((r+1)I) = \{(r+1)^r\}.$$

After applying a series of row and column operations on $\phi_{\mathscr{P}}([(r+1)I - 2J], \lambda)$ and using Lemma 5, we get the corresponding $\mathscr{P}$-eigenvalues as

$$\mathrm{Spec}_{\mathscr{P}}((r+1)I - 2J) = \{(r+1)^{(r-1)}, [-(r-1)]^1\}.$$

Therefore, from Lemma 4

$$\mathrm{Spec}_{\mathscr{P}}(K_{r,r}) = \{(r+1)^{(n-1)}, [-(r-1)]^1\}$$

and

$$E_{\mathscr{P}}(K_{r,r}) = \frac{1}{2}[n^2 + 2n - 4].$$

□

Theorem 25 *Let $K_{r,r}$ be a complete bipartite graph of order n with bipartite sets $\{u_1, u_2, \ldots, u_r\}$ and $\{v_1, v_2, \ldots, v_r\}$ and the vertex partition $\mathscr{P} = \{\{u_i, v_i\}$, for $1 \leq i \leq r\}$. Then*

$$E_{\mathscr{P}}(K_{r,r}) = 3n - 2.$$

Proof. The $\mathscr{P}$-matrix of $K_{r,r}$ is

$$A_{\mathscr{P}}(K_{r,r}) = \begin{pmatrix} 2I_{r\times r} & (J+I)_{r\times r} \\ (J+I)_{r\times r} & 2I_{r\times r} \end{pmatrix}_{n\times n}.$$

Thus, the characteristic polynomial of $A_{\mathscr{P}}(K_{r,r})$ is

$$\phi_{\mathscr{P}}(K_{r,r},\lambda) = [\lambda + (r-1)](\lambda-1)^{(r-1)}(\lambda-3)^{(r-1)}[\lambda-(r+3)].$$

Hence,

$$\mathrm{Spec}_{\mathscr{P}}(K_{r,r}) = \{(r+3)^1, 3^{(r-1)}, 1^{(r-1)}, [-(r-1)]^1\}$$

and

$$E_{\mathscr{P}}(K_{r,r}) = 3n-2.$$

□

Now, consider a graph obtained by removing 1-factor F_1 from a complete bipartite graph $K_{r,r}$ and denote is by $K_{r,r} - F_1$ [3].

Theorem 26 *Let $K_{r,r} - F_1$ be a graph of order $n = 2r$, for $r \geq 3$ with a vertex partition $\mathscr{P} = \{V_1, V_2\}$ such that V_1 and V_2 are two partite sets of $K_{r,r} - F_1$. Then*

$$E_{\mathscr{P}}(K_{r,r} - F_1) = \frac{1}{2}[n^2 + 2n - 8].$$

Proof. The $\mathscr{P}$-matrix of $K_{r,r} - F_1$ for the given vertex partition is

$$A_{\mathscr{P}}(K_{r,r} - F_1) = \begin{pmatrix} [(r+1)I - J]_{r\times r} & (J-I)_{r\times r} \\ (J-I)_{r\times r} & [(r+1)I-J]_{r\times r} \end{pmatrix}_{n\times n}.$$

By Lemma 4 and 5,

$$\mathrm{Spec}_{\mathscr{P}}(K_{r,r} - F_1) = \{(r+2)^{(r-1)}, r^r, [-(r-2)]^1\}.$$

Therefore,

$$E_{\mathscr{P}}(K_{r,r} - F_1) = \frac{1}{2}[n^2 + 2n - 8].$$

□

In the next theorem, we consider the partition $\mathscr{P} = \{\{u_i, v_i\},\ \text{for } 1 \leq i \leq r\}$ and determine the corresponding $\mathscr{P}$-energy for $K_{r,r} - F_1$.

Theorem 27 *Let $K_{r,r} - F_1$ be a graph of order $n = 2r$, for $r \geq 3$ with a vertex partition $\mathscr{P} = \{\{u_i, v_i\},\ \text{for } 1 \leq i \leq r\}$. Then*

$$E_{\mathscr{P}}(K_{r,r} - F_1) = \begin{cases} 2n & \text{for } n = 6 \text{ and } 8, \\ 3n-8 & \text{for } n > 8. \end{cases}$$

.

Proof. The $\mathscr{P}$-matrix of $K_{r,r} - F_1$ is

$$A_{\mathscr{P}}(K_{r,r} - F_1) = \begin{pmatrix} 2I_{r\times r} & (J-2I)_{r\times r} \\ (J-2I)_{r\times r} & 2I_{r\times r} \end{pmatrix}_{n\times n}.$$

Thus, the characteristic polynomial of $A_{\mathscr{P}}(K_{r,r} - F_1)$ is

$$\phi_{\mathscr{P}}(K_{r,r} - F_1, \lambda) = \lambda^{(r-1)}(\lambda-4)^{(r-1)}(\lambda-r)[\lambda+(r-4)].$$

Therefore,

$$\mathrm{Spec}_{\mathscr{P}}(K_{r,r} - F_1) = \left\{\frac{n}{2}^1, 4^{(\frac{n}{2}-1)}, 0^{(\frac{n}{2}-1)}, \left[-\left(\frac{n}{2}-4\right)\right]^1\right\}.$$

Hence, $E_{\mathscr{P}}(K_{r,r} - F_1) = 2n$, for $n = 6, 8$ and $E_{\mathscr{P}}(K_{r,r} - F_1) = 3n - 8$, for $n > 8$. □

5 Conclusion

The significance of $\mathscr{P}$-energy stems from the importance of vertex partition problems in graph theory. As observed from the discussions, the value of $E_{\mathscr{P}}(G)$ depends on factors such as the number of elements in the partition, the nature of the vertex subsets in the partition and the specific properties that determines the partition. In this direction, there is also much scope for extension of the study of the concept of $\mathscr{P}$-energy as we can consider specific vertex partitions such as domatic partitions and equitable degree partitions and study the relation between the corresponding $\mathscr{P}$-energy and other graph parameters.

It is to be noted that there are various algorithms available for partitioning a graph (or a network) [8, 9]. Their applications are well known such as partitioning a network into clusters [15], community detection problem in social sciences etc. [8]. We have observed that, in [9], the authors have presented certain parameters for measuring some key aspects of the network like modularity, z-score etc using quantities such as number of partitions k, and the numbers m_1, m_2 which are mentioned in Observation 1(iv) in a similar context. So by using these algorithms and with the help of $\mathscr{P}$-energy, there is a possibility for developing a tool for a given network to find its specific properties such as spectral clustering or community structures.

References

[1] B. D. Acharya, S. Rao, P. Sumathi, V. Swaminathan, Energy of a set of vertices in a graph, *AKCE Int. J. Graphs. Combin.* **4,** 2 (2007) 144–152. ⇒139

[2] C. Adiga, E. Sampathkumar, M. A. Sriraj, A. S. Shrikanth, Color energy of a graph, *Proc. Jangjeon Math. Soc.* **16,** 3 (2013) 335–351. ⇒138
[3] D. Archdeacon, M. Debowsky, J. Dinitz, H. Gavlas, Cycle systems in the complete bipartite graph minus a one-factor, *Discrete Math.* **284,** 1-3 (2004) 37–43. ⇒155
[4] D. M. Cvetković, M. Doob, H. Sachs, *Spectra of Graphs*, Academic Press, New York, 1980. ⇒137, 139
[5] I. Gutman, The energy of a graph *Ber. Math. Stat. Sekt. Forschungsz. Graz.* **103** (1978) 1–22. ⇒137
[6] I. Gutman, B. Furtula, The total π-electron energy saga, *Croat. Chem. Acta.* **90,** 3 (2017) 359–368. ⇒138
[7] I. Gutman, X. Li, J. Zhang, *Graph Energy*, Springer, New York, 2012. ⇒138
[8] M. Girvan, M. E. J. Newman, Community structure in social and biological networks, *PNAS USA* **99,** 12 (2002) 7821–7826. ⇒156
[9] R. Guimera, L. A. N. Amaral, Functional cartography of complex metabolic networks *nature*, **433,** 7028 (2005) 895–900. ⇒156
[10] M. Marcus, H. Minc, *A Survey of Matrix Theory and Matrix Inequalities*, Courier Corporation, 1992. ⇒139
[11] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*, Siam, 2000. ⇒139
[12] E. Sampathkumar, S. V. Roopa, K. A. Vidya, M. A. Sriraj, Partition energy of a graph, *Proc. Jangjeon Math. Soc.* **18,** 4 (2015) 473–493. ⇒138
[13] E. Sampathkumar, M. A. Sriraj, Vertex labeled/colored graphs, matrices and signed graphs, *J. Comb. Inf. Syst. Sci.* **38** (2013) 113–120. ⇒140
[14] D. B. West, *Introduction to Graph Theory*, Pearson, New Jersey, 2001. ⇒137
[15] M. Zhang, J. Deng, C. V. Fang, X. Zhang, L. J. Lu, Molecular network analysis and applications *Knowledge-Based Bioinformatics: John Wiley and Sons, Ltd.* (2010) 251–287. ⇒156

Received: April 23, 2020 • Revised: June 10, 2020

Acta Univ. Sapientiae, Informatica **12,** 1 (2020) 158–158

Errata: Heuristic method to determine lucky k-polynomials for k-colorable graphs

Johan KOK
CHRIST (Deemed to be a University), Bangalore,
India
email: jacotype@gmail.com

1 Errata note

In [1] titled, *Heuristic method to determine lucky* k*-polynomials for* k*-colorable graphs*, regrettable errors occurred in Table 1 on page 210. The errors relate to the coefficients of the lucky 3-polynomials of the null graphs for $n = 7$ and 8, respectively. The corrected table is depicted below.

n	$\mathfrak{N}_n$, [7]	P_n	C_n
3	$\lambda(\lambda-1)(\lambda-2)$	$\lambda(\lambda-1)(\lambda-2)$	$\lambda(\lambda-1)(\lambda-2)$
4	$6\lambda(\lambda-1)(\lambda-2)$	$3\lambda(\lambda-1)(\lambda-2)$	$2\lambda(\lambda-1)(\lambda-2)$
5	$15\lambda(\lambda-1)(\lambda-2)$	$6\lambda(\lambda-1)(\lambda-2)$	$5\lambda(\lambda-1)(\lambda-2)$
6	$15\lambda(\lambda-1)(\lambda-2)$	$5\lambda(\lambda-1)(\lambda-2)$	$4\lambda(\lambda-1)(\lambda-2)$
7	$105\lambda(\lambda-1)(\lambda-2)$	$16\lambda(\lambda-1)(\lambda-2)$	$13\lambda(\lambda-1)(\lambda-2)$
8	$280\lambda(\lambda-1)(\lambda-2)$	$41\lambda(\lambda-1)(\lambda-2)$	$34\lambda(\lambda-1)(\lambda-2)$

References

[1] J. Kok, Heuristic method to determine lucky k-polynomials for k-colorable graphs, *Acta Univ. Sapientiae, Informatica* **11,** 2 (2019), 205–213. $\Rightarrow$158

Computing Classification System 1998: G.2.2
Mathematics Subject Classification 2010: 05C15, 05C38, 05C75, 05C85
Key words and phrases: chromatic completion, perfect lucky 3-coloring, lucky 3-polynomial

Acta Universitatis Sapientiae

The scientific journal of Sapientia Hungarian University of Transylvania publishes original papers and surveys in several areas of sciences written in English. Information about each series can be found at
http://www.acta.sapientia.ro.

Acta Universitatis Sapientiae, Informatica

Each volume contains two issues.

Sapientia University
Sciendo by De Gruyter
Scientia Publishing House

ISSN 1844-6086

http://www.acta.sapientia.ro

Information for authors

Acta Universitatis Sapientiae, Informatica publishes original papers and surveys in various fields of Computer Science. All papers are peer-reviewed.

Papers published in current and previous volumes can be found in Portable Document Format (pdf) form at the address: `http://www.acta.sapientia.ro`.

The submitted papers should not be considered for publication by other journals. The corresponding author is responsible for obtaining the permission of coauthors and of the authorities of institutes, if needed, for publication, the Editorial Board is disclaiming any responsibility.

Submission must be made by email (`acta-inf@acta.sapientia.ro`) only, using the LaTeX style and sample file at the address `http://www.acta.sapientia.ro`. Beside the LaTeX source a pdf format of the paper is necessary too.

Prepare your paper carefully, including keywords, ACM Computing Classification System codes (`http://www.acm.org/about/class/1998`) and AMS Mathematics Subject Classification codes (`http://www.ams.org/msc/`).

References should be listed alphabetically based on the Intructions for Authors given at the address `http://www.acta.sapientia.ro`.

Contact address and subscription:
Acta Universitatis Sapientiae, Informatica
RO 400112 Cluj-Napoca
Str. Matei Corvin nr. 4.
Email: acta-inf@acta.sapientia.ro

Printed by F&F INTERNATIONAL
Director: Enikő Ambrus

ISSN 1844-6086
http://www.acta.sapientia.ro

Megvalósult a Magyar Kormány Támogatásával
Miniszterelnökség Nemzetpolitikai Államtitkárság
Bethlen Gábor Alap