

Performance Evaluation of Sequential Minimal Optimization and K* Algorithms for Predicting Burst Header Packet Flooding Attacks on Optical Burst Switching Networks

Ebru Efeoglu and Gurkan Tuna


Abstract— Optical burst switching networks are vulnerable to various threats including Burst Header Packet Flooding attack, Circulating Burst Header attack, Address Spoofing, and Replay attack. Therefore, detecting such threats play a key role in taking appropriate security measures. One of the major challenges in identifying the risks of Burst Header Packet flooding attacks is the lack or insufficiency of reliable historical data. In this paper, firstly, Burst Header Packet flooding attacks are classified into four categories, Misbehaving-Block, Behaving-No Block, Misbehaving-No Block and Misbehaving-Wait, using Sequential Minimal Optimization (SMO) and K* algorithms. Using performance metrics obtained both after testing on the same set and after applying 10-fold cross validation, the performance of SMO and K* algorithms is compared based on commonly used performance metrics. As the results show, compared to SMO, K* algorithm is more suitable for predicting Burst Header Packet Flooding attacks.

Index Terms— Burst header packet flooding, Classification algorithms, Optical burst switching, Performance evaluation, Security threats.


I. INTRODUCTION

OVER THE last couple of decades, optical networks have experienced a tremendous growth rate in parallel with the ever increasing bandwidth demand for the transmission of multimedia data [1]. As a consequence, nowadays optical networks are used as not only backbones but also access networks. Optical Burst Switching (OBS) combines the advantages of Optical Circuit Switching (OCS) and Optical Packet Switching (OPS) while considering the main limitations of the current all-optical technology [2].

EBRU EFEOGLU is with Kütahya Dumlupınar Üniversitesi Yazılım Mühendisliği Bölümü, Kütahya, Turkey, (e-mail: ebru.efeoğlu@dpu.edu.tr).

 <https://orcid.org/0000-0001-5444-6647>

GURKAN TUNA is with Trakya University, Edirne, Turkey, (e-mail: gurkantuna@trakya.edu.tr).

 <https://orcid.org/0000-0002-6466-4696>

Manuscript received March 7, 2021; accepted September 18, 2021.

DOI: [10.17694/bajece.892150](https://doi.org/10.17694/bajece.892150)

OBS networks have the advantage of statistical multiplexing and low control overhead requirements [3]. To achieve these, in OBS networks, after being collected at the edge of the network, the user data is first sorted based on its destination address and then grouped into variable sized bursts. Before these bursts are transmitted, a control packet, called as Burst Header Packet (BHP), is generated for each of them and then transmitted to the destination node in order to establish a bufferless path, with the goal of reserving network resources, for these corresponding bursts. After a predetermined delay time, called offset time, each of these bursts itself is transmitted without waiting an acknowledgment from the destination [2].

In OBS networks, there is a possibility of some bursts contend at the same outgoing channel and at the same instant, thereby creating a Burst Contention [4]. Therefore, proposed a channel scheduling algorithm to efficiently schedule bursts to outgoing wavelength links with the goal of reducing transmission losses [5]. In recent years, the use of artificial intelligence techniques has been proposed to improve the performance and security of OBS networks. It has been shown that predictive models and feature reduction methods can be used to predict burst contention/blocking probability in OBS networks; this way the number of burst losses can be reduced and the performance of OBS networks can be improved drastically [6]. Similarly, Wang, [7] proposed a burst assembly algorithm, which adjusts the time threshold adaptively depending on the Quality of Service (QoS) priority to reduce the assembly time delay of the services requiring high QoS by 2.81%-14.68% without additional overhead.

Security threats that OBS networks suffer from can be categorized into two groups: Orphan Bursts and Malicious Burst Headers [8]. An orphan burst occurs during transmission when the scheduling request for a BHP is rejected because of excessive demands and the corresponding data burst is not handled [9]. As a consequence, by flowing along an unintended path it may waste the bandwidth or might be tapped by attackers. BHPs compromised by attackers may lead to forming malicious burst headers and can be used in different types of attacks including Burst Hijacking, Burst Control Header Flooding attack, Land attack, Timeout attack and Replay attack [8].

In OBS networks, BHPs undergo Optical/Electrical/Optical conversion process at the intermediate nodes; although, their bursts are transmitted all-optically. Sending BHPs ahead of bursts exposes the bursts to various security threats, particularly Data Burst Redirection (DBR) and Denial of Service (DoS) attacks; because, if the BHPs are compromised, then the corresponding bursts will possibly be compromised [10]. In recent years, various defense mechanisms have been proposed against the security threats, including self-healing survivable optical rings, optical encryption, optical steganography, optical code-division multiple access confidentiality, and anti-jamming. However, integrating some of the functionalities require real-time processing of optical signal [11]. Coulibaly et al. [10] proposed a Rivest-Shamir-Adleman (RSA) algorithm based-approach to deal with DBR attacks and proved that the proposed approach could reduce the number of compromised BHPs in case of DBR attacks. Considering the vulnerability of OBS networks against physical layer attacks, particularly DoS attacks, Sliti, Hamdi, and Boudriga [1] proposed an architecture of a firewall node with dynamic roles to protect OBS networks.

One of the most challenging threats for OB networks is BHP flooding attack; but, identifying it is challenging because of the scarcity of reliable historical data. In this paper, the use of SMO and K* algorithms for classifying BHP flooding attacks is proposed.

Different from the existing works presented in the related work section, in this study two different algorithms which were not previously used with the employed dataset are used. Using SMO and K* algorithms, the attacks are first classified into four main classes as Misbehaving-Block, Behaving-No Block, Misbehaving-No Block and Misbehaving-Wait. Then, using well-known performance metrics obtained both after testing on the same set and after applying 10-fold cross validation, the performance of SMO and K* algorithms is compared based on commonly used performance metrics. Thus, the performance of the algorithms in the classification of data that is not in the database has been also examined. While performing the performance evaluation, a comprehensive performance analysis has been made by using many different performance metrics, not an analysis based on a single performance metric.

II. RELATED WORK

Previously, the dataset used in this study was classified by Rajab using decision tree algorithms and an accuracy rate of 87% was obtained [12]. Again, using the same dataset, classification was made with Naïve Bayes and Bayes Net algorithms and 69% and 85% accuracy rates were obtained, respectively [13].

In another study using the same dataset, Decision Table, JRIP, OneR, PART-m, ZeroR, Decision Stump, Hoeffding Tree, J48, LMT and REP Tree algorithms were used, and high accuracy rates were obtained with LMT and PART-m algorithms [14]. Compared to LMT, PART-m took less time for the classification [14].

Using the same dataset, performance analysis of J48, Multilayer Perceptron (MLP), Naïve Bayes, Logistic, Random Tree (RT), Reduce Error Pruning (REP) Tree algorithms were made in [15]. As a result of the analysis, it was seen that the most successful algorithm among these algorithms was the J48 algorithm [15].

III. MATERIAL AND METHOD

A. Dataset

The dataset used in the study was downloaded from the UCI data library [11]. In the dataset, there are 1075 samples belonging to 4 classes: Misbehaving-Block (Block), Behaving-No Block (No Block), Misbehaving-No Block (NB-No Block), and Misbehaving-Wait (NB-Wait). As listed in Table 1, the number of attributes of the data is 22. Although there is an unbalanced dataset, no sample has been deleted or a sample has been reduced from the dataset. In order to obtain more reliable results while performing performance analysis, Matthews Correlation Coefficient (MCC) metric, which is used in evaluating the classification performance of unbalanced datasets, has been added to the performance metrics.

TABLE 1. ATTRIBUTE INFORMATION

	Attribute	Type
1	Node	Numeric
2	Utilized Bandwidth Rate	Numeric
3	Packet Drop Rate	Numeric
4	Reserved Bandwidth	Numeric
5	Average_Delay_Time_Per_Sec	Numeric
6	Percentage_Of_Lost_Packet_Rate	Numeric
7	Percentage_Of_Lost_Byte_Rate	Numeric
8	Packet Received Rate	Numeric
9	Used Bandwidth	Numeric
10	Lost Bandwidth	Numeric
11	Packet Size_Byte	Numeric
12	Packet_Transmitted	Numeric
13	Packet_Received	Numeric
14	Packet_lost	Numeric
15	Transmitted_Byte	Numeric
16	Received_Byte	Numeric
17	10-Run-AVG-Drop-Rate	Numeric
18	10-Run-AVG-Bandwidth-Use	Numeric
19	10-Run-Delay	Numeric
20	Node Status' {B, NB, P NB}	Categorical
21	Flood Status	Numeric
22	Class ' {NB-No Block, Block, No Block, NB-Wait}	Categorical

B. Classifiers

SMO is basically an algorithm that uses support vectors [16]. It is used to train the support vector classifier using a polynomial kernel. This way it globally replaces all missing values and converts nominal attributes to binary ones. It also normalizes all attributes with predefined values. On the other hand, K* algorithm is an instance-based classifier. Sample-

based methods are based on comparing a sample with an unknown attribute in the test dataset with the samples in the training dataset that were previously classified in the database but not revealed. The difference of this algorithm from other sample-based learners is that it uses an entropy-based distance function [17].

There are several metrics used to evaluate the performance of classifiers. In this study, accuracy, precision, sensitivity, specificity and F-measure, well known and commonly used performance metrics, have been obtained using a confusion matrix. A confusion matrix example created for a binary classification is given in Fig.1. TP (True Positives) and TN (True Negatives) indicate total number of instances predicted correctly, FP (False Positives) and FN (False Negatives) indicate total number of instances predicted incorrectly. F-measure is the harmonic mean of Sensitivity and Precision metrics and it is used in case of incompatible Precision and Sensitivity values (low Sensitivity and high Precision, or vice versa). In case of imbalanced datasets, Matthew's Correlation Coefficient (MCC) is used. Kappa is a metric used to compare an observed accuracy with an expected accuracy. Root Mean Square (RMS) indicates how much error the algorithm makes while performing a classification process.

		Predicted Class		
		Positive	Negative	
Actual Class	Positive	True Positive (TP)	False Negative (FN)	Sensitivity $\frac{TP}{(TP + FN)}$
	Negative	False Positive (FP)	True Negative (TN)	Specificity $\frac{TN}{(TN + FP)}$
		Precision $\frac{TP}{(TP + FP)}$	F-Measure $\frac{TP}{TP + 1/2(FP+FN)}$	Accuracy $\frac{TP + TN}{(TP + TN + FP + FN)}$
$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$				

Fig.1. Confusion matrix

VI. RESULTS

Confusion matrices of SMO and K* algorithms have been obtained both after testing on the same set and after applying 10-fold cross validation. They are shown in Fig.2 and Fig.3. The boxes in red in the confusion matrices show the number of instances that the algorithms have predicted incorrectly, and the boxes in blue show the number of instances they have correctly predicted. For example, SMO algorithm has correctly predicted 415 instances in training that were actually NB-No Block. Likewise, it has estimated 130 instances that were actually NB-Wait as NB-No Block. In the confusion matrices, the following class labels are used to abbreviate the long sentences. Block refers to Misbehaving-Block, No Block refers to Behaving-No Block, NB-No Block refers to Misbehaving-No Block, and finally, NB-Wait refers to Misbehaving-Wait.

SMO-Training					
		Predicted class			
		NB-No Block	Block	No Block	NB-Wait
Actual class	NB-No Block	415	0	10	75
	Block	0	105	0	15
	No Block	0	0	155	0
	NB-Wait	130	70	0	170

(a)

SMO-Cross Validation					
		Predicted class			
		NB-No Block	Block	No Block	NB-Wait
Actual class	NB-No Block	420	0	10	70
	Block	0	105	0	15
	No Block	0	0	155	0
	NB-Wait	131	0	0	169

(b)

Fig.2. Confusion matrices of SMO, a) Training (testing on the same set), b) After applying 10-fold cross validation

K*-Training					
		Predicted class			
		NB-No Block	Block	No Block	NB-Wait
Actual class	NB-No Block	500	0	0	0
	Block	0	120	0	0
	No Block	0	0	155	0
	NB-Wait	0	0	0	300

(a)

K*-Cross Validation					
		Predicted class			
		NB-No Block	Block	No Block	NB-Wait
Actual class	NB-No Block	500	0	0	0
	Block	0	120	0	0
	No Block	0	0	155	0
	NB-Wait	0	0	0	300

(b)

Fig.3. Confusion matrices of K*, a) Training (testing on the same set), b) After applying 10-fold cross validation

Total number of correct and incorrect predictions of algorithms are shown in Fig. 4. It is seen that SMO algorithm has predicted 845 instances correctly and 230 instances incorrectly. But, when cross validated, it has predicted 849 instances correctly and 227 instances incorrectly. On the other hand, K* has shown the same performance both prior to and after cross validation, and has predicted 1075 instances correctly and 0 incorrectly.

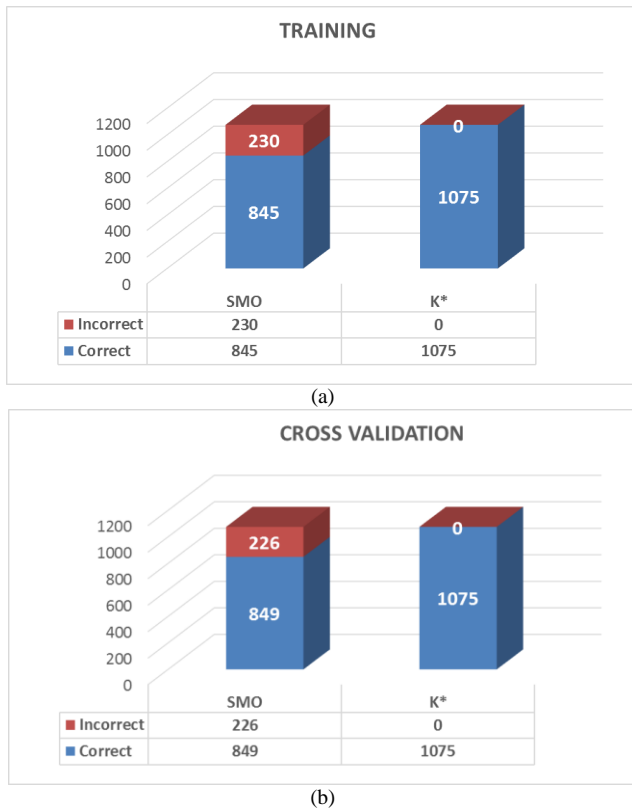


Fig.4. Number of correctly and incorrectly classified instances, a) Training (testing on the same set), b) After applying 10-fold cross validation

The distribution of the instances classified correctly and incorrectly by the algorithms are given in Fig.5 and Fig.6. The cross (x) sign in the figures represent the correctly classified instances and those shown with a square represent the incorrectly classified instances.

The values of other performance metrics used in comparing the performances of the classification algorithms are given in Table 2. Precision, Sensitivity, and F-Measure range from 0 to 1. On the other hand, Kappa and MCC range from -1 to 1, but usually range from 0 to 1. For all these metrics, a value of 1 indicates that the perfect classification has been made. For this reason, these values are desired to be as close to 1 as possible while making the classification. When Table 2 is examined and all the metrics are taken into consideration, it can be seen that K* has performed better than SMO in this classification task in both after testing on the same test and after applying 10-fold cross validation. In addition, the lower RMS values of K* confirms this conclusion.

TABLE 2. PERFORMANCE METRICS

Performance Metrics	Training		Cross validation	
	SMO	K*	SMO	K*
Accuracy (%)	78	100	0.78	100
Precision	0.78	1	0.79	1
Sensitivity	0.78	1	0.79	1
F-Measure	0.78	1	0.78	1
Kappa	0.67	1	0.68	1
RMS	0.33	0.0001	0.33	0.0002
MCC	0.65	1	0.66	1

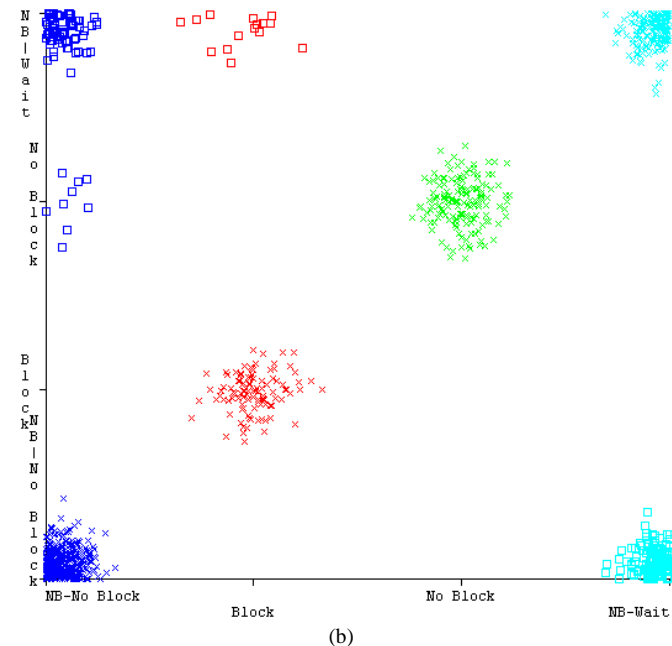
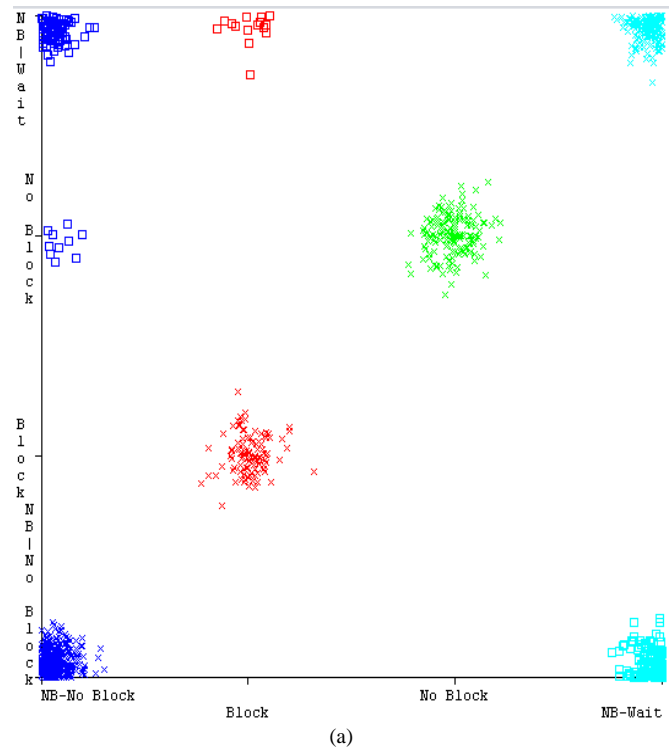


Fig. 5. SMO, a) Training (testing on the same set), b) After applying 10-fold cross validation

Although the use of Deep Convolution Neural Network (DCNN) was proposed in [18] and it was shown that DCCN had a better performance compared to Naïve Bayes, K-Nearest Neighbours and Support Vector Machine, excellent results have been achieved by K* algorithm in this study. Considering the fact that integrating some of the functionalities of security techniques require real-time processing of optical signal, the use of deep learning techniques for this purpose are questionable [11]. Since OBS networks are seen as the solution for future high-speed optical

networks [19], research on potential threats should be continued.

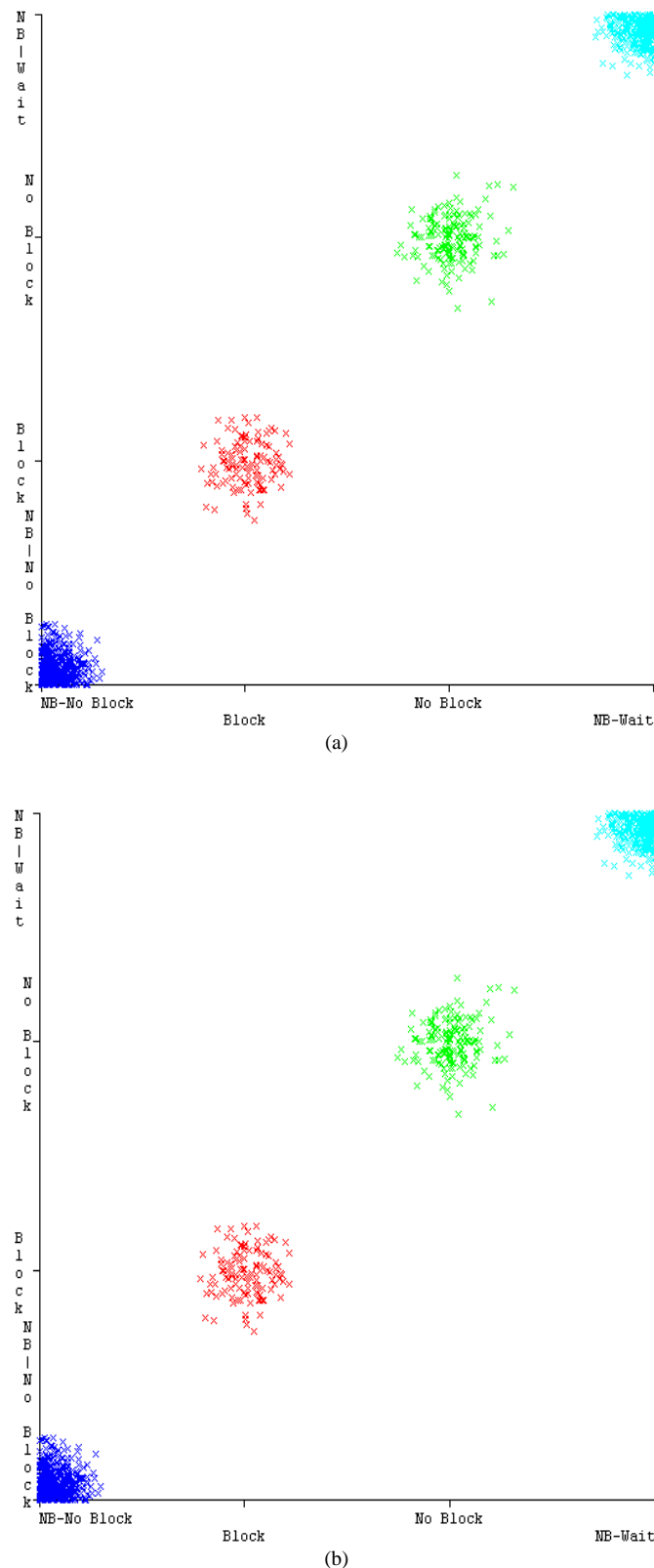


Fig. 6. K*, a) Training (testing on the same set), b) After applying 10-fold cross validation

V. CONCLUSION

Although optical networks have unique properties such as broadband operation, optical processing-instantaneous response, electromagnetic immunity, low latency and compactness, it is known that they are vulnerable to various attacks, including physical infrastructure attacks, jamming, eavesdropping, and interception; therefore, the physical layer security of optical networks cannot be overlooked. OBS networks are particularly vulnerable to various threats relating to BHPs; therefore, identifying those threats play a key role in securing OBS networks.

One of those threats is BHP flooding attack; however, identifying it is challenging due to the scarcity of reliable historical data. In this paper, firstly, BHP flooding attacks have been classified into four main categories, Misbehaving-Block, Behaving-No Block, Misbehaving-No Block and Misbehaving-Wait, using SMO and K* algorithms. Using performance metrics obtained both after testing on the same set and after applying 10-fold cross validation, the performance of SMO and K* algorithms has been compared based on the commonly used performance metrics. As the results show, compared to SMO, K* algorithm is more suitable for predicting BHP flooding attacks.

REFERENCES

- [1] M. Sliti, M. Hamdi, and N. Boudriga, "A novel optical firewall architecture for burst switched networks," in *2010 12th International Conference on Transparent Optical Networks*, 2010: IEEE, pp. 1-5. doi: 10.1109/ICTON.2010.5549054
- [2] T. Battestilli and H. Perros, "An introduction to optical burst switching," *IEEE communications magazine*, vol. 41, no. 8, pp. S10-S15, 2003. doi: 10.1109/MCOM.2003.1222715
- [3] M. Maier, *Optical switching networks*. Cambridge University Press, 2008. doi:10.1017/CBO9780511619731.011
- [4] V. M. Vokkarane, J. P. Jue, and S. Sitaraman, "Burst segmentation: an approach for reducing packet loss in optical burst switched networks," in *2002 IEEE International Conference on Communications. Conference Proceedings. ICC 2002 (Cat. No. 02CH37333)*, 2002, vol. 5: IEEE, pp. 2673-2677. doi: 10.1109/ICC.2002.997328
- [5] P. K. Chandra, A. K. Turuk, and B. Sahoo, "Survey on optical burst switching in WDM networks," in *2009 International Conference on Industrial and Information Systems (ICIIS)*, 2009: IEEE, pp. 83-88. doi: 10.1109/ICIINFS.2009.5429885
- [6] S. Chakraborty, A. K. Turuk, and B. Sahoo, "OBS network blocking probability prediction using ensemble technique," in *2020 IEEE International Symposium on Sustainable Energy, Signal Processing and Cyber Security (iSSSC)*, 2020: IEEE, pp. 1-6. doi: 10.1109/iSSSC50941.2020.9358862
- [7] Y. Wang, T. Chen, and N. Zhou, "Space-based Optical Burst Switching Assembly Algorithm Based on QoS Adaption," in *2019 IEEE 11th International Conference on Communication Software and*

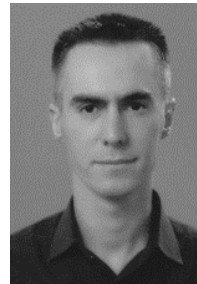
- Networks (ICCSN)*, 2019: IEEE, pp. 101-105. doi: 10.1109/ICCSN.2019.8905264
- [8] B. T. F. Fernandez and C. Sreenath, "Burstification threat in optical burst switched networks," in *IEEE proceeding of International Conference on Communication and Signal Processing*, 2014, pp. 1666-1670. doi: 10.1109/ICCSN.2014.6949804
- [9] Y. Chen and P. K. Verma, "Secure optical burst switching: Framework and research directions," *IEEE Communications magazine*, vol. 46, no. 8, pp. 40-45, 2008. doi: 10.1109/MCOM.2008.4597102
- [10] Y. Coulibaly, A. A. I. Al-Kilany, M. S. Abd Latiff, G. Rouskas, S. Mandala, and M. A. Razzaque, "Secure burst control packet scheme for Optical Burst Switching networks," in *2015 IEEE International Broadband and Photonics Conference (IBP)*, 2015: IEEE, pp. 86-91. doi: 10.1109/IBP.2015.7230771
- [11] M. P. Fok, Z. Wang, Y. Deng, and P. R. Prucnal, "Optical layer security in fiber-optic networks," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 725-736, 2011. doi: 10.1109/TIFS.2011.2141990
- [12] A. Rajab, C. Huang and M. Al-Shargabi, "Decision Tree Rule Learning Approach to Counter Burst Header Packet Flooding Attack in Optical Burst Switching Network", *Optical Switching and Networking*, Vol. 29, pp. 15-26, July 2018.
- [13] R. Alshboul, " Flood Attacks Control in Optical Burst Networks by Inducing Rules using Data Mining", *IJCSNS International Journal of Computer Science and Network Security*, Vol. 18, February 2018.
- [14] S. Kavithal, M. Hanumanthappa and A. Syrien, "Evaluation of Optical Burst Switching (OBS) Using Various Classification Techniques", *National Conference On Contemporary Research and Innovations in Computer Science (NCCRICS)*, Dec. 2017.
- [15] V.N. Uzel and E. Saraç Eşsiz, "Classification BHP Flooding Attack in OBS Network with Data Mining Techniques." In *International Conference on Cyber Security and Computer Science (ICONCS'18)*, 2018, pp. 134-137.
- [16] J. Platt: Fast Training of Support Vector Machines using Sequential Minimal Optimization. In B. Schoelkopf and C. Burges and A. Smola, editors, *Advances in Kernel Methods, Support Vector Learning*, 1998.
- [17] J.G. Cleary, , L.E. Trigg, "K*: An Instance-based Learner Using an Entropic Distance Measure", *Proceedings Twelfth International Conference on Machine Learning*, Tahoe City, California, 1995, pp. 108-114.
- [18] M. Z. Hasan, K. M. Hasan, and A. Sattar, "Burst Header Packet Flood Detection in Optical Burst Switching Network Using Deep Learning Model," *Procedia Computer Science*, vol. 143, pp. 970-977, 2018. doi: 10.1016/j.procs.2018.10.337
- [19] M. Hola, L. Scholtz, L. Ladanyi, and J. Mullerova, "Modeling of optical burst switching networks," *Proc. SPIE 10976, 21st Czech-Polish-Slovak Optical*

Conference on Wave and Quantum Aspects of Contemporary Optics, 2018, 1097610. doi: 10.1117/12.2518407

BIOGRAPHIES



EBRU EFEOGLU is currently an Assistant Professor at Kutahya Dumlupinar University Software Department. She received her B.S. degree in Geophysics Engineering from Kocaeli University and Management Information Systems from Anadolu University, Turkey. She received her Ph.D. degree in Computer Engineering from Trakya University, Turkey in 2021. She has authored several papers in international conference proceedings and SCI-Expanded journals. Her research interests include machine learning and data mining, and their applications in various research domains.



GURKAN TUNA is currently a Professor at the Department of Computer Programming at Trakya University, Turkey. He is also the head of the graduate program of Mechatronics Engineering at the same university. His current research interests include wireless networks, wireless sensor networks, mobile robots, multi-sensor fusion, and smart cities.