Packet Scheduling Using SVM Models in Wireless Communication Networks

Sovit Bhandari[1], Hong Ping Zhao[1], Hoon Kim[1], Pervez Khan[2], Sana Ullah[3]

[1] IoT and Big-data Research Center, Department of Electronics Engineering, Incheon National University, South Korea
[2] Department of Computer Science and IT, University of Malakand, Pakistan
[3] Department of Computer and Software Technology, University of Swat, Pakistan
sovit198@gmail.com, {hongpieng614, hoon}@inu.ac.kr, {pervaizkanju, sanajcs}@gmail.com

Abstract

This paper addresses a support vector machine (SVM) model-based packet scheduling in wireless communication networks. The linear SVM, as well as the polynomial SVM model-based proportional fair scheduling (SVM-PFS), are suggested. Moreover, their performances are analyzed for various system and machine learning parameters such as the average window, signal to interference ratio (SIR) and the degree of a polynomial. From the computer simulations, the performance measurements of PFS, such as, user selection fairness and user throughput are calculated, which proves that performance of SVM-PFS approaches to that of the conventional metric-based PFS in most cases, while the simulation results show that the polynomial SVM-PFS outperforms the linear SVM-PFS. It is also shown that the SVM-PFS performs better under higher SIR or larger average window.

Keywords: Packet scheduling, Support vector machine, Proportional fairness, Machine learning

1 Introduction

The goal of packet scheduling in wireless communication networks is to find efficient assignment between transmission channel and access users [1-3]. Most solutions to the problem of packet scheduling have been mainly based on either prescribed algorithm or iterative calculation. A proportional fair scheduling (PFS), proposed by Kelly [4], maximizes the logarithmic sum of users throughput is considered to be one of most practical scheduling policies in which it provides a good trade-off between system throughput and fairness [5-8]. PFS's for single and multi-carrier systems have been proposed and widely employed for wireless communication networks system performance evaluations [8-9].

Meanwhile, the need of machine learning (ML) techniques in network resource allocation has been motivated as high computing power are expected to be equipped in communication networks in near future. In [10], ML modules distributed in multiple layers in network architecture is proposed and the ML application to routing problems has been considered [11]. In [12], it is proposed to use ML to detect flows in each module of a packet-switched optical network (PSON). However, some issues such as feasibility study of ML adoption for specific scheduling algorithms, its effectiveness, and challenges still remain to be solved.

In this paper, the ML scheme that trains the scheduler in order to achieve PFS and its accuracy over PFS is suggested based on computer simulations. Amongst ML algorithms, the SVM classification rule is applied and its accuracy with various average powers of communication links, average window size, the range of timeslot, and the degree of a polynomial employed are analyzed. From the simulations, better qualities of ML schemes are achieved under a high degree of SVM, high SIR, and large average window.

2 System Model and PFS

2.1 System Model

The schematic diagram in Figure 1 shows the PFS which is done by a metric-based and ML-based approach. A system for downlink packet scheduling is considered, where the communication channels between the users and the radio access point to receive data from the network side are used. The channel information is assumed to be collected perfectly and the scheduler computes average throughput for each user. Based on those information, the scheduler assigns the user to the channel.

*Corresponding Author: Hoon Kim; E-mail: hoon@inu.ac.kr
DOI: 10.3966/160792642019092005016

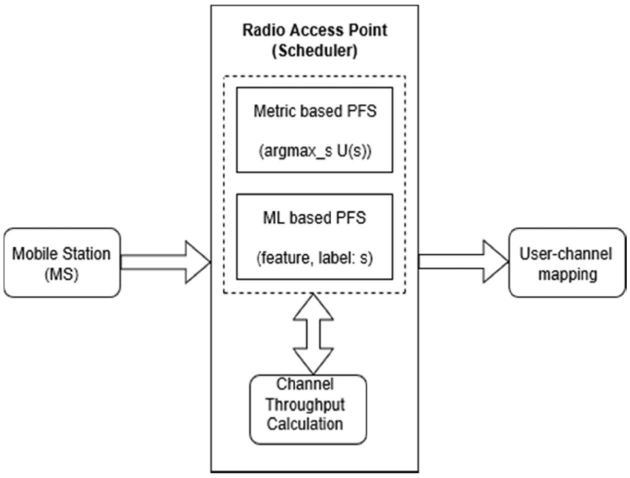

Figure 1. System model for PFS

2.2 PFS

Scheduling is a resource allocation scheme that select users for whom the data will be sent in the next transmission. The criteria of proportional fairness has been considered in wireless communication networks because it achieves a good trade-off between system throughput and fairness unlike round robin and maximum carrier-to-interference (CIR) method of scheduling. In this scheme, the user is selected on the basis of performance they exhibit while accessing the respective timeslot. For instance, the user for the current scheduling epoch is determined by the performance they showed in the previous timeslot as well as the channel capacity of a current timeslot.

It is known that the PFS should maximize the sum of logarithmic user utility, which can be represented as below when the user utility is defined by average throughput of the user.

$$PFS = \arg\max_s \Sigma_i \log R_i^{(s)} \tag{1}$$

Where, $R_i^{(s)}$ is the average throughput of user i expected by scheduler s, which can be calculated by

$$R_i^{(s)} = \frac{(T-1)R_i' + r_i^{(s)}}{T} \tag{2}$$

Where T is the averaging window, R_i' is the average throughput at the previous timeslot, and $r_i^{(s)}$ is the instantaneous channel capacity for user i at the current timeslot if scheduler s assigns that very user. The value of the instantaneous channel capacity $r_i^{(s)}$ becomes 0 for the user i which is not assigned by the scheduler s at that respective timeslot.

3 PFS Schemes

Out of many scheduling schemes, we are going to discuss only two methods of PFS scheduling, namely, metric-based scheduling and ML-based scheduling. The metric-based scheduler determines the user-channel mapping based on scheduling metric computations. Whereas, the ML-based scheduler utilizes the features and produces the output for user-channel mapping without metric computations.

3.1 Metric-based PFS

In metric-based PFS, metric computations are done to determine the user-channel mapping. In Figure 1, s means a unique scheduling policy which maps users to channels identically and assigns users for the communication channel. PFS that provides a good trade-off between throughput and fairness is assumed to be the best scheduling policy. For single carrier wireless communication systems, the scheduler can be identified by the user index, and it is simply implemented by comparing the user-by-user metric as given below [8]:

$$PFS = \arg\max_i \frac{r_i^{(i)}}{R_i'} \tag{3}$$

The above PFS equation for single carrier system shows that the selection of user for the next timeslot depends upon the performance they exhibited in the previous timeslot. It is a non-bias scheme of scheduling. It tries to select the user which is not selected in the previous timeslot. For illustration, at N timeslot, PFS tries to select the user which has comparatively less value of average throughput (R_i) than other users in N-1 timeslot, provided that the all the user exhibit same throughput $(r^{(1)})$ at the N timeslot.

3.2 ML-based PFS

In the ML-based PFS, an SVM model is selected to quantify the performance of the ML approach while dealing with PFS. SVMs have been popular for data classification [13]. An SVM often maps data to a high dimensional space and then employs Kernel techniques [14]. In a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other. An SVM model is a representation of the examples as points in space, mapped in a way that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall. This kind of SVM classification is referred to as large margin classification or hard margin classification. There are two main issues with hard margin classification. The first issue associated with hard margin classification is that it only works if the data is linearly separable, and the second one is that it is quite sensitive to outliers.

To mitigate the problem associated with hard margin classification, it is preferable to use a more flexible SVM model, such as soft margin classification. The objective of soft margin classification is to find a good balance between large street width and small margin violations. In Scikit-Learn's SVM classes, we can control this balance by changing the value of C hyperparameter. A smaller value of C leads to a wider street but leads to more margin violations, while the larger value of C leads to fewer margin violations but ends up with a smaller margin. The value of C must be chosen in such a way that it provides a good tradeoff between the width of the street and the margin violation [12].

3.2.1 Linear SVM

Linear SVM is one type of ML algorithm. It classifies the data with the linear approach. It works perfectly for the data which can be classified linearly i.e. data with few numbers of features. Linear SVM tries to find a separating hyper-plane between two classes with the approach of keeping maximum gap in-between. A hyper-plane is a set of x points in d-dimension which satisfies the following equation:

$$w^T + b = 0 \quad \textbf{(4)}$$

$$h(x) = w^T + b \quad \textbf{(5)}$$

where w is the weight vector having the d-dimension, $h(x)$ is the hyper-plane function and b is the scalar denoting the bias. Initially, the values of w and b will be unknown to us. We have to find the values of w and b in such a way that the separating hyper-plane $h(x)$ can separate the two different classes of data as correctly as possible [14].

Although linear SVM classifiers are efficient and work surprisingly well in many cases, many datasets are not even close to being linearly separable. In this kind of scenario, it is advisable to use a non-linear SVM algorithm i.e. polynomial SVM.

3.2.2 Polynomial SVM

Polynomial SVM is used when the data cannot be separated linearly. It maps data into a higher plane and constructs a hyper-plane to separate the data. When there are multiple features, polynomial SVM is capable of finding relationships between features, unlike linear SVM method. This kind of ML-algorithm is used when the data has a large number of features. For example, if there were two features x and y, and with the polynomial feature of degree 3, this would not only add the features x^2, x^3, y^2, and y^3, but also the combinations xy, x^2y, and xy^2 It should be noted that the polynomial feature of a degree d transforms an array containing n features into an array containing $\frac{(n+d)!}{n!d!}$ features, where $n!$ is the factorial of n, equal to $1 \times 2 \times 3 \times \ldots n$. We should keep in mind about the combinatorial explosion of the number of features [12].

With this paper, it is going to be verified that the polynomial SVM approach of classifying data is better than the linear SVM approach of classifying data, provided that the data depends on the more than two features.

4 Performance Evaluation

4.1 Simulation Parameters

To compare the performances of the metric-based PFS solution and the ML-based PFS, some system scenarios are assumed. In ML-based PFS, linear as well as polynomial SVM method is taken into account to find the better result. A single transmission channel is considered in our simulation. A single channel is shared by two users, the statistics of user signal to interference ratio (SIR) is assumed to be independent of Rayleigh fading. The channel capacity is calculated by using Shannon's theorem i.e. $\log_2(1+SIR)$. The model of classification used for ML in this paper is SVM. The ML features include channel capacity, average throughput, and the ML label is scheduled user in each scheduling time slot. To evaluate the accuracy of the ML algorithm, the output of the ML-based scheduling scheme is compared to metric-based PFS. The time slots taken for simulation is 10,000 for various channel average powers. The training and test data sets for ML are partitioned into the former and the latter half of timeslots. In our simulation, for the training data sets, 1-5000 timeslot is taken, whereas, for test data sets, 5001-10000 timeslot is taken. The Table 1 given below summarizes the simulation parameters and their values which are taken into account.

Table 1. Simulation parameters

Parameters	Value
Number of users	2
Number of channel	1
Channel variation	Rayleigh
Channel capacity	$\log_2(1+SIR)$
Average window timeslots (T)	10,20
ML model	SVM (linear, polynomial)
Number of timeslots	10,000
Test data sets	1-5000
Training data sets	5001-10000

4.2 Simulation Results

4.2.1 PFS Characteristics

In this section, the output results of the proportional fairness metric, user selection ratio and user throughput fairness are displayed for the linear as well as the polynomial SVM-PFS.

Figure 2 shows the labels of linear SVM based PFS versus the prediction for user selection with respect to the ratio of the proportional fair (PF) metric of user-2 to user-1. In the y-axis, “0” means user-1 selection, whereas “1” means user-2 selection. The metric-based PFS selects user-2 if the ratio of PF metric of user-2 to user-1 is greater than 1. For linear SVM-based PFS, there are some points where the user-2 is not selected for the ratios greater than 1, similarly, user 1 is not selected for the ratio less than 1.

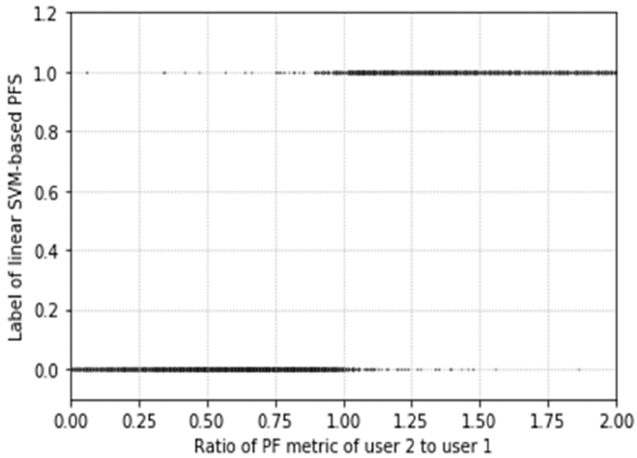

Figure 2. Label vs. ratio of PF metric of user 2 to user 1 for linear SVM method

Figure 3 shows the labels of polynomial SVM based PFS versus the prediction for user selection with respect to the ratio of the PF metric of user-2 to user-1. In the polynomial SVM based PFS method, we can see that, there are only a few points where the user-2 is not selected for the ratios greater than 1 and ‘vice-versa’.

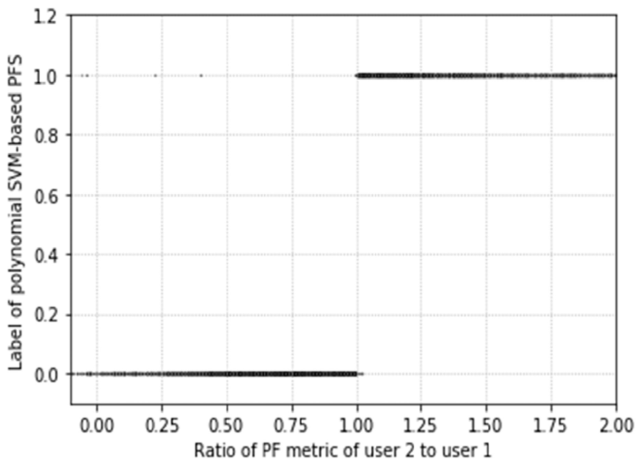

Figure 3. Label vs. Ratio of PF metric of user 2 to user 1 for polynomial SVM method

If we compare the figures i.e., Figure 2 and Figure 3, we can say that the polynomial SVM based PFS method selects the users more accurately than the linear SVM based PFS method.

Figure 4 and Figure 5 show the user selection rate of PFS scheme using linear as well as polynomial SVM, respectively. The respective results show that the user selection rate between different users using PFS scheme is almost similar i.e., around the marginal line of 0.5, even if we change the value of average SIR. This results also show that metric based PFS scheme as well SVM-PFS scheme display similar output in terms of fairness.

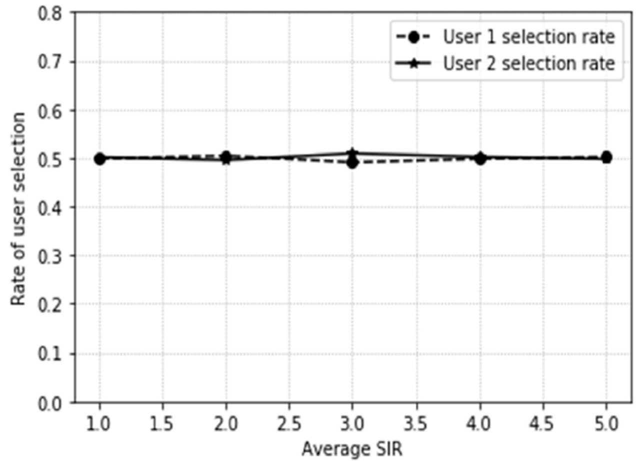

Figure 4. User selection rate with increasing SIR using linear SVM model of ML

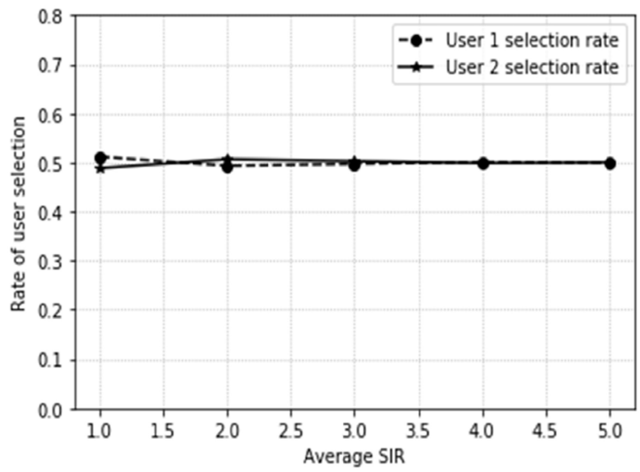

Figure 5. User selection rate with increasing SIR using polynomial SVM model of ML

The graph in Figure 6 illustrates the throughput ratios between different users using the linear SVM model of ML. Similarly, the graph in Figure 7 illustrates that the throughput ratios between users using the polynomial SVM model of ML.

The graphs in Figure 6 and Figure 7 portraits that the throughput ratios between different users in the PFS scheme are almost similar for different values of SIR, irrespective of ML model used.

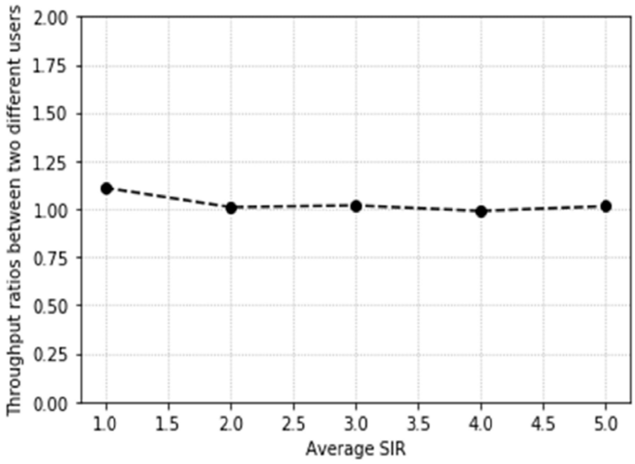

Figure 6. Throughput ratio between two different users for different values of SIR using linear SVM model of ML

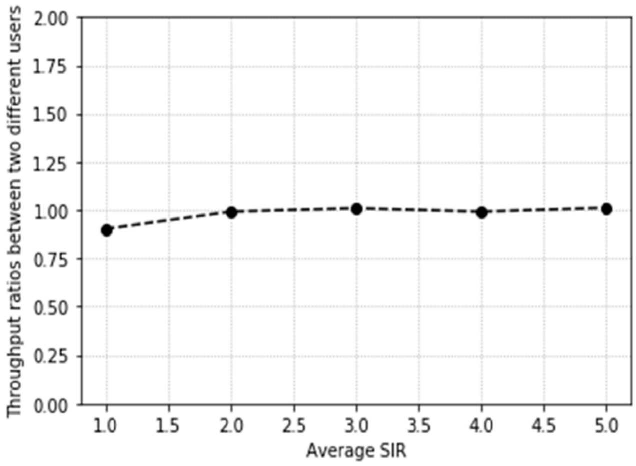

Figure 7. Throughput ratio between two different users for different values of SIR using polynomial SVM model of ML

4.2.2 Average Time Window

The outputs of the SVM-PFS are shown here for two different values of average time window *T*.

Figure 8 shows the accuracy of the linear SVM ML-based scheme compared to metric-based scheme for average window *T* of 10. The ML accuracies are plotted for different values of SIR. The performance of the suggested ML-based PFS scheduling almost approaches to that of the conventional PFS in most cases. The accuracy is improved as the channel average power gets high. This is because higher channel capacity provides more reliable data input to the ML algorithm, which results in fewer errors.

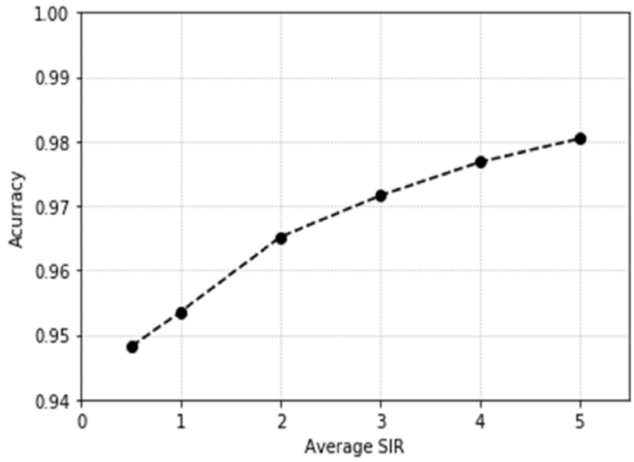

Figure 8. The accuracy of linear SVM ML-based scheme compared to metric-based scheme (T=10)

Figure 9 shows the accuracy of the polynomial SVM ML-based scheme compared to metric-based scheme for *T* of 10. It is notified that more accurate scheduling in terms of PFS is achieved by the polynomial SVM ML-based scheme. This is because polynomial components in the features can be reflected by the polynomial SVM ML-based scheme, which results in a smaller error.

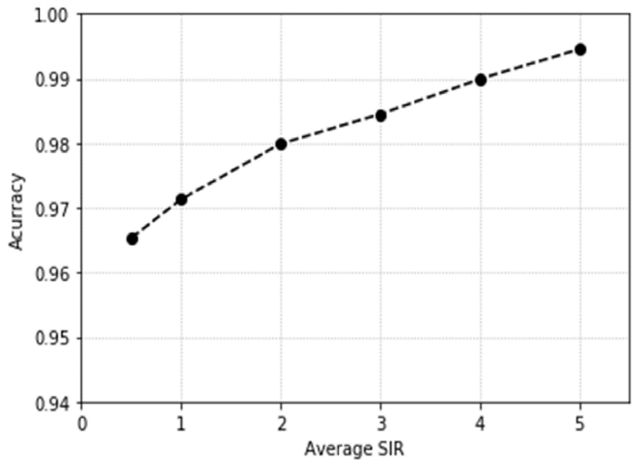

Figure 9. The accuracy of polynomial SVM ML-based scheme compared to metric-based scheme (T=10)

Similarly, Figure 10 shows the accuracy of the linear SVM ML-based scheme compared to metric-based scheme for *T* of 20. If we compare Figure 8 and Figure 10, we can say that the higher value of *T* provides better output.

Likewise, Figure 11 shows the accuracy of the polynomial SVM ML-based scheme compared to metric-based scheme for *T* of 20. If we tally Figure 9 and Figure 11, we can say that the simulated output in Figure 11 is better than in Figure 9.

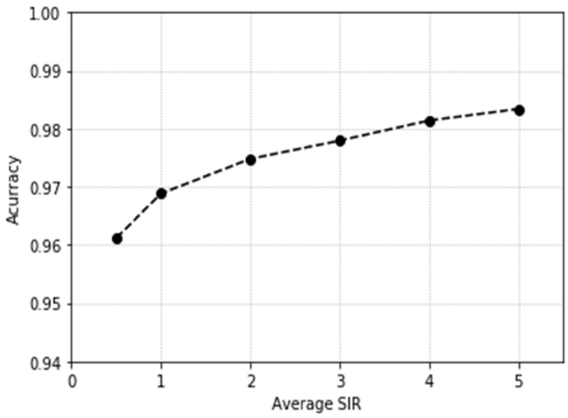

Figure 10. The accuracy of linear SVM ML-based scheme compared to metric-based scheme (T=20)

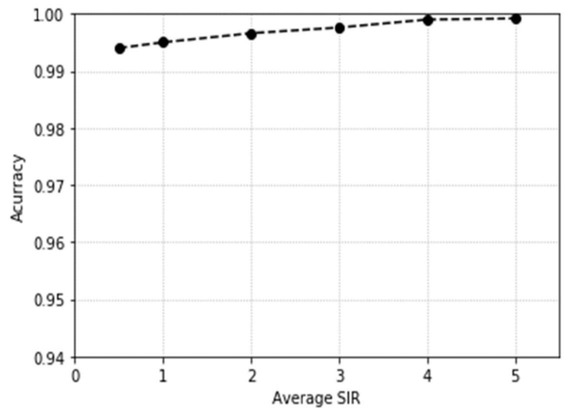

Figure 11. The accuracy of polynomial SVM ML-based scheme compared to metric-based scheme (T=20)

4.2.3 Number of Training Samples

The outputs of SVM-PFS for different values of the training samples are presented here.

Figure 12 shows that the accuracy of linear SVM based ML for the different number of the training samples i.e. 10, 100, 1000, and 10000, respectively. The ML accuracy is in the increasing trend when the training sample is increased by ten folds and almost saturates after the certain value of the training sample.

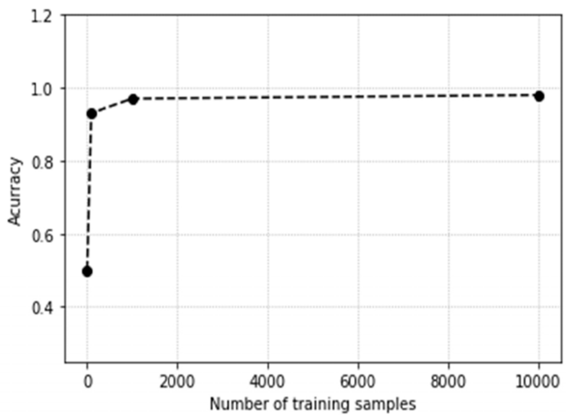

Figure 12. Linear SVM based ML accuracy for different number of training samples

Likewise, Figure 13 shows that the accuracy of polynomial SVM based ML for the different number of training samples. If we compare the outputs as seen in Figure 12 and Figure 13, we can draw a conclusion that the polynomial SVM-PFS provides more accurate result than the linear SVM-PFS, irrespective of the number of the training samples used.

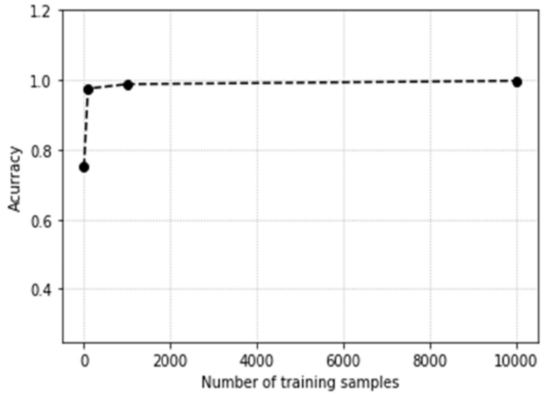

Figure 13. Polynomial SVM based ML accuracy for different number of training samples

4.2.4 Number of Polynomial Degrees

The simulated result of the SVM-PFS is shown here for the different values of the polynomial degree.

Figure 14 shows that the accuracy of polynomial SVM is in increasing trend if we increase the polynomial degree. After the polynomial degree of 4, ML accuracy slides down for the polynomial degree 5, but there is slight improvement for the polynomial degree of 6. This unusual behavior of the accuracy of ML algorithm is due to increment in the complexity as the increase in the degree of polynomial value. The low value of accuracy for a low polynomial degree is due to the underfitting, whereas, low value of accuracy for a high polynomial degree is due to overfitting. Here, overfitting is a condition when your model performs well on training data but predicts poorly due to the cross-validation metrics. If the model performs poorly on training data as well as test data, then it is said to be underfitting.

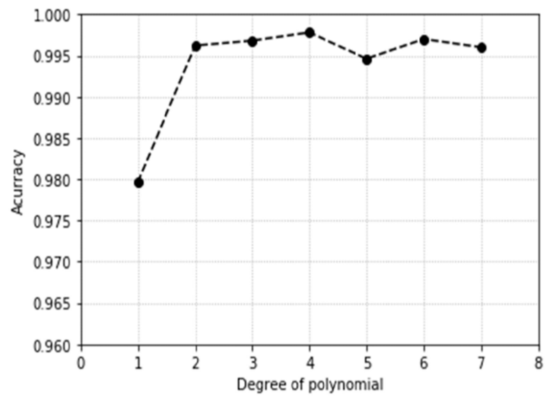

Figure 14. The accuracy of ML with respect to degree of polynomial employed

5 Conclusion

This letter addresses two approaches of PFS, metric-based PFS, and ML-based PFS. In this paper, the SVM model is considered as an ML model. By deploying the SVM model, the accuracy of the linear and the polynomial SVM ML-based PFS's are measured. From the computer simulations, it is found that the performance of the ML-based PFS scheduling almost approaches to that of the conventional PFS in most cases. Likewise, from the obtained results we can say that the polynomial SVM method predicts better than the linear SVM method. The effects of simulation parameters on the accuracy are also investigated, for instance, the ML-based PFS predicts more accurately for a higher value of T. Likewise, the prediction made by the linear and the polynomial SVM model varies according to the timeslot. It is also found that the prediction made by the linear and the polynomial SVM model increases as the values of the timeslot is increased. However, the prediction value almost saturates after a certain limit of training samples. The number of training samples can be increased up to a certain limit due to the limiting nature of computer performance.

Moreover, within the polynomial SVM method, the prediction value can vary depending upon the degree of a polynomial function we have used in our simulation. For the data we have employed in our program, we found that the polynomial SVM method predicting better for the higher degree of a polynomial. It is also notified that the degree of a polynomial which can predict better varies according to the number of features that are present in the system and the relationship between them.

For the future works, the performance of PFS for multiple channels under different ML simulation environments needs to be addressed considering different simulation parameters using high performing capacity computers.

Acknowledgements

This work was supported by Incheon National University Research Grant in 2013.

References

[1] R. Kwan, C. Leung, J. Zhang, Multiuser Scheduling on the Downlink of an LTE Cellular System, *Research Letters in Communications*, Vol. 2008, pp. 1-4, May, 2008.

[2] M. Zulhasnine, C. Huang, A. Srinivasan, Efficient Resource Allocation for Device-to-Device Communication Underlaying LTE Network, *2010 IEEE 6th International Conference on Wireless and Mobile Computing, Networking and Communications*, Niagara Falls, Ontario, 2010, pp. 368-375.

[3] A. Pokhariyal, K. I. Pedersen, G. Monghal, I. Z. Kovacs, C. Rosa, T. E. Kolding, P. E. Mogensen, HARQ Aware Frequency Domain Packet Scheduler with Different Degrees of Fairness for the UTRAN Long Term Evolution, *IEEE Vehicular Technology Conference*, Dublin, Leinster, 2007, pp. 2761-2765.

[4] F. P. Kelly, Charging and Rate Control for Elastic Traffic, *European Transactions on Telecommunications*, Vol. 8, No. 1, pp. 33-37, January-February, 1997.

[5] S. B. Lee, I. Pefkianakis, A. Meyerson, S. Xu, S. Lu, Proportional Fair Frequency-Domain Packet Scheduling for 3GPP LTE Uplink, *IEEE INFOCOM 2009*, Rio de Janeiro, Brazil, 2009, pp. 2611-2615.

[6] A. Jalali, R. Padovani, R. Pankaj, Data Throughput of CDMA-HDR a High Efficiency-High Data Rate Personal Communication Wireless System, *Proceeding of IEEE Vehicular Technology Conference-Spring*, Tokyo, Japan, 2000, pp. 1854-1858.

[7] J.-H. Jeon, J.-T. Lim, Delay Controlled Proportional Fair Scheduling in Rayleigh Fading Wireless Channel, *IET Communications*, 2012, Vol. 6, No. 17, pp. 2816-2824, November, 2012.

[8] H. Kim, Y. Han, A Proportional Fair Scheduling for Multicarrier Transmission Systems, *IEEE Communications Letters*, Vol. 9, No. 3, pp. 210-212, March, 2005.

[9] S. Teerapittayanon, B. McDanel, H. Kung, *Distributed Deep Neural Networks over the Cloud the Edge and End Devices*, https://arxiv.org/abs/1709.01921.

[10] A. Valadarsky, M. Schapira, D. Shahaf, A. Tamar, *A Machine Learning Approach to Routing*, https://arxiv.org/abs/1708.03074.

[11] L. Wang, X. Wang, M. Tornatore, K. Kim, S. Kim, D. Kim, K. Han, B. Mukherjee, Scheduling with Machine-Learning-Based Flow Detection for Packet-Switched Optical Data Center Networks, *Journal of Optical Communications and Networking*, Vol. 10, No. 4, pp. 365-375, April, 2018.

[12] A. Géron, *Hands-on Machine Learning with Scikit- Learn and TensorFlow: Concepts, Tools, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*, O'Reilly Media, 2017.

[13] S. Dong, W. Liu, D.-D. Zhou, Y.-C. Qi, NSVM: A New SVM Algorithm Based on Traffic Flow Metric, *Journal of Internet Technology*, Vol. 16, No. 6, pp. 1005-1014, November, 2015.

[14] Y. Chang, C. Hsieh, K. Chang, M. Ringgaard, C. Lin, *Training and Testing Low-degree Polynomial Data Mappings via Linear SVM*, http://jmlr.csail.mit.edu/papers/v11/chang10a.html.

Biographies

Sovit Bhandari received his Bachelor's degree from Kathmandu University, Nepal, in 2016. He is currently working as a Graduate Research Assistant at Incheon National University, South Korea. His research interests include networking, 5G networks, wireless communications, machine learning and internet of things.

Hong Ping Zhao received his Bachelor's degree from Jilin Jianzhu University, China, in 2016. He is currently working as a Graduate Research Assistant at Incheon National University, South Korea. His research interests include machine learning, big data and networking.

Hoon Kim has been working as an Associate Professor at Department of Electronics Engineering, Incheon National University, South Korea, since 2008. His research interests include radio resource management, optimization techniques, 5G mobile communication systems, machine learning, and internet of things. He is a member of KICS, IEIE, IEEE, and IEICE.

Pervez Khan has been working as an Assistant Professor at Department of Computer Science and Information Technolgy, University of Malakand, Pakistan, since March 2018. His research interests include wireless communications, wireless sensor networks, wireless ad-hoc networks, wireless body area networks, 5G networks, machine learning, fuzzy logic and MAC protocol design.

Sana Ullah is currently working as an Assistant Professor at Department of Computer and Software Technology, University of Swat, Pakistan, and is also serving as a Senior Researcher at Gyeongsang National University, South Korea. He is currently an editor for the Wiley SCN, JIT, IJAACS, JOMS and KSII TIIS Journals.