doi:10.20944/preprints201901.0311.v1

Machine Learning Techniques to Detecting and Preventing Jamming Attacks in Optical Networks

Mounir Bensalem, Sandeep Kumar Singh, and Admela Jukan
Technische Universität Carolo-Wilhelmina zu Braunschweig, Germany
{m.bensalem, sandeep.singh, a.jukan}@tu-bs.de

***Abstract*—We study the effectiveness of various machine learning techniques, including artificial neural networks, support vector machine, logistic regression, K-nearest neighbors, decision tree and Naive Bayesian, for detecting and mitigating power jamming attacks in optical networks. Our study shows that artificial neural network is the most accurate in detecting out-of-band power jamming attacks in optical networks. To further mitigating the power jamming attacks, we apply a new resource reallocation scheme that utilizes the statistical information of attack detection accuracy, and propose a resource reallocation algorithm to lower the probability of successful jamming of lightpaths. Simulation results show that higher the accuracy of detection, lower is the likelihood of jamming a lightpath.**

I. Introduction

Optical networks are vulnerable to different types of attacks, including jamming, eavesdropping, and infrastructure attacks [1]. The jamming attacks in optical networks in particular are usually launched by inserting a relatively high power signal over either a frequency within the transmission window (*in-band jamming*) or out of the transmission band (*out-of-band jamming*) of legitimate data channels. Although both types of jamming are common and can disrupt communication or even steal information with the help of crosstalk mechanism, the out-of-band jamming is hard to detect under some circumstances, where the power of the jamming channel does not differ much than the legitimate channels. Furthermore, an out-of-band jamming signal at an erbium-doped amplifier (EDFA) can cause gain competition and starve legitimate signals from amplification. More importantly, a relatively high power signal in any of the channels in wavelength-division multiplexed (WDM) networks when pass through multiple EDFAs along its route can accumulate higher amplified spontaneous emission noise and fiber nonlinear interference noise, which lead to degradation of optical signal-to-noise ratio (OSNR) and bit error rate (BER) of in-band and out-of-band channels [2].

Previous work proposed attack prevention mechanisms, such as data encryption, quantum key distribution, and scrambling techniques [3]–[6]. While these methods are hard to break, attackers can still try to launch signal insertion and splitting attacks without being noticed. Therefore, an attack detection – in addition to prevention – is fundamentally important as well. In the literature, past schemes based on power detection, spectrum analysis, and optical time domain reflectometry methods have been proposed, but they have been shown useful for offline analysis, and are prone to errors under noisy condition and in large-scale networks. As the data rate of today's cloud-centric optical networks exceed 100 Gigabits per second, attack detection and prevention in real time have become a challenge. To this end, machine learning (ML) has recently gained significant attention as a statistical method and a viable solution to attack detection to address the issues of attack detection and prevention in real time and accurately [7]. In addition, ML and other artificial intelligence methods, have been used for various optical parameter monitoring, resource (re)allocation, and network reconfiguration [8]–[10].

In this paper, we study the effectiveness of various ML solutions to detection and mitigation of power jamming attacks in optical networks. We focus on combined detection as well as mitigation of the jamming attacks. For the detection part, we comparatively study the effectiveness of various ML techniques, such as artificial neural networks (ANN), support vector machine (SVM), logistic regression (LR), K-nearest neighbors (KNN), decision tree (DT) and Naive Bayesian (NB), to detecting out-of-band power jamming attacks in a WDM network. For the attack mitigation part, we utilize the probabilistic attack detection accuracy, and propose a scheme to reassign routes and wavelengths to lighpaths, such that there is low probability of continuous and future jamming or observing of legitimate data signals carried by authorized lightpaths. Our analysis shows that only a few number of reconfigurations (through resource reassignment process) are sufficient to secure lightpaths, while artificial neural networks (ANN) and support vector machine (SVM) are the most suitable for detecting power jamming attacks.

The rest of this paper is organized as follows. Section II presents the jamming model with ML techniques. Routing and wavelength reassignment for attack prevention scheme is presented in Section III. Section IV presents the data extraction method for a WDM network scenario, and evaluates its performance. Finally, we conclude the paper in Section V.

II. Power Jamming Detection Model

A. Jamming Attack Model

We investigate the out-of-band jamming attacks in WDM networks which are primarily used to disrupt services and require physical access to a legitimate network access point. Thus, we define the power jamming attack as the ability of an unauthorized user to sneak into the network and access a light source (laser), and thereafter launch a signal with higher power than other authorized wavelength channels, inducing nonlinearity in the fiber and degradation of OSNR and BER of other

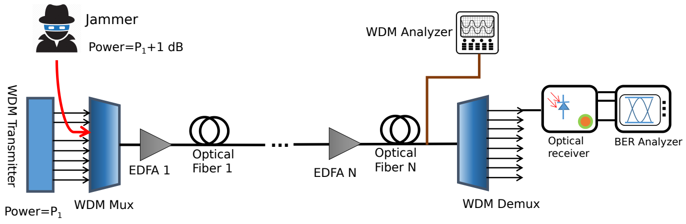

Fig. 1: A WDM transmission system under a jamming attack.

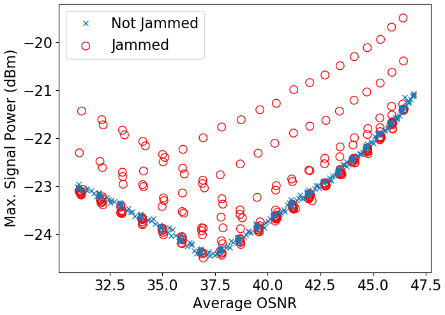

Fig. 2: Max. received power are shown against average OSNR.

neighboring channels. Generally, users can be classified based on their trust so that an attacker's location could be predicted, however in optical networks, localization of the attacks is a challenge. In our approach, we model the jamming attack, and then apply different ML techniques for the detection of jamming without localization of the attacker.

We distinguish between two types of user: authorized users, characterized by transmitter sources which generate optical signals with a defined signal power, and unauthorized users, characterized by transmitter sources that generate higher power jamming signals. Fig. 1 illustrates authorized and unauthorized users in an optical WDM system, where the authorized sources in a WDM transmitter send signals with power P_1, and an unauthorized signal is inserted by a Jammer (attacker) in the 3^{rd} channel with $+1$ dB higher power than the legitimate WDM channels. As shown in Fig. 1, signals are multiplexed using a WDM Mux, then amplified with a series of EDFAs during the transmission. At the end, optical receivers and BER Analyzers can be used for monitoring of received signals. A WDM Analyzer is connected before the WDM Demux to capture different metrics about all channels such as signal power, noise power, SNR or OSNR. Afterwards, these metrics can be analyzed to detect whether there is a jamming attack in the network using a detection algorithm.

To design a detection technique for the jamming attack, we first observed the impact caused by an attack on different parameters. Some metrics such as the signal power and OSNR react more to jamming attacks while some others react less, e.g., noise power. This study might give insights on how difficult the detection can be. Thus, let us first characterize the data samples obtained from a simulation corresponding to authorized (not jammed) and unauthorized (jammed) signals. In Fig. 2, we plot the maximum signal (channel) power measured by a WDM analyzer just before the WDM Demux against the average OSNR of channels. We observe that a high number of data points from authorized and unauthorized signals are overlapping. Therefore, a threshold method will not be an efficient solution to detect a jamming scenario.

ML techniques offer powerful statistical solution for decision making through a learning mechanism. In order to use ML algorithms for jamming detection, we need to first collect data for training of ML model classifiers. The data represent instances of the network in which a jamming attack can happen. For each instance, the input represent the measured features and the output corresponds to a binary variable indicating the presence of the attack. We collect the information about two features: the signal power and OSNR. We add a WDM Analyzer in the receiver side to monitor theses features.

Let's assume that an authorized WDM transmitter generates W different wavelengths with equivalent channel power $P_w = P_1, 1 \leqslant w \leqslant W$, and a Jammer inserts an unauthorized signal into an unused channel J $(1 \leqslant J \leqslant W, J \neq w)$ with a power $P_J = P_w + \epsilon$, where 1 dB $\leqslant \epsilon \leqslant 3$ dB. The received signals before demultiplexing corresponding to a w^{th} channel is denoted by $RS_w = (f_w^1, ..., f_w^m)$, where f_w^m is the value of the m^{th} feature of a received signal corresponding to the channel number w. Each received signal can be characterized by different features as described previously. In our study, we consider the signal power and OSNR as features. Therefore, we denote the received signal as $RS_w = (RP_w, ROSNR_w)$, where RP_w is the received power, and $ROSNR_w$ is the received OSNR of the channel number w. Now, let's consider that we have multiple instances (N_I) of the network in Fig 1. An instance of the network generates one data point i where $X_i = (RS_1, ..., RS_W)$ is the input data, and the output $Y_i = 1$ if there is a jamming and 0 otherwise. It is important to mention that various ML techniques utilized for detecting jamming attacks use the same dataset consisting of input samples, each with $2W$ feature vector, i.e., $X_i \in \mathbb{R}^{2W}, 1 \leqslant i \leqslant N_I$.

B. ML Techniques for Attack Detection

Several ML algorithms can be used to detecting jamming attacks. In this paper, we compare the following six popular learning algorithms to detect jamming attacks in optical networks, and evaluate them under a network scenario to find the most accurate technique for an attack detection tool.

K-Nearest Neighbors (KNN): This algorithm is based on feature similarity to identify the corresponding class of the tested samples. A sample is assigned to a class that gets the vote of its K-neighbors. We consider our input data as pair of vectors $(X_i, Y_i)_{1 \leqslant i \leqslant N_I}$. Let $\| \cdot \|$ be the Euclidean distance on $\mathbb{R}^{2W}$, and $x' \in \mathbb{R}^{2W}$ be a test instance of a network which is to be classified into a jamming or non-jamming. The KNN runs through all the pair of data to compute the distance between x' and X_i then it reorders the distances and considers only the K nearest training samples (Ω the set of the K nearest samples) to estimate the conditional probability for each class $Pr(Y = 1|X = x') = \frac{1}{K} \sum_{i \in \Omega} M(Y_i = 1)$, where $M(\alpha)$ denotes an indicator function (1 if α is true and 0 otherwise).

Decision Tree (DT): This classifier uses a graph model in the form of a tree to introduce conditional statements in which each node represents a conditional test on an attribute to output a class represented by one of the leaves. The training process is used to build the tree that minimizes the classification error. In our case, the decision tree divides the feature space of dimension $2W$ in every node. To build the tree, we consider the Gini impurity measure which is computed as $I_G(p) = 1 - \sum_{i=1}^{C} p_i^2$, where C is the number of classes and p_i is the fraction of samples labeled with class i. The training is based on an iterative approach that aims to minimize the impurity to obtain the best split in each node of the tree. To classify a new sample, we need to process the tree from the root to the leaf, and assign the majority class of reached leaf.

Logistic Regression (LR): This model is mostly used for binary classification problems. It uses a linear equation to predict a real value, and applies a Sigmoid function to map the value between 0 and 1 so that it can output a binary class. Let p be the probability of a positive jamming attack in the network, and given as $p = \frac{e^y}{1+e^y}$, $y = \beta_0 + \beta_1 X_i^1 + ... + \beta_{2W} X_i^{2W}$, where β_0 is a bias term and $(\beta_i)_{1 \leqslant i \leqslant 2W}$ are the partial regression coefficients, and $\{X_i^j, 1 \leqslant j \leqslant 2W\}$ is a feature vector of a data sample X_i. To find the best value of each coefficients so that the classifier detects an attack, the following linear relation can be used: $y = \ln(\frac{p}{1-p})$. Then, the sum of squared errors is used to perform the optimization of coefficients.

Artificial Neural Network (ANN): This classifier is composed of computing units called neurons organized in layers to map an input vector to an output. The neurons can perform non linear transformations which give it the power to represent and generalize complex problems. In our case, we use a feed forward ANN with one class output. We used a Sigmoid activation function to model the non-linear decision boundary. Let the objective function be $-\sum_{O_h} p(O_h) log(q(O_h))$, where O_h is the output of a node from the previous hidden layer, p is the ground truth and q is the output of the neural network. We aim to minimize the objective function by updating the parameters at each node in the network using a solver such as gradient decent. Finally, a Softmax function is used in the output layer to decide the class.

Support Vector Machine (SVM): This is a discriminating classifier used to separate the space by hyperplanes to classify new data samples. For nonlinear classification, the SVM maps the data to a higher dimensional feature space, and perform a linear separation. The hyperplane is formulated as the following classification rule $f(X) = \langle w, X \rangle + b$, where w is a vector parameter, and b is a bias term. Here, X can be replaced by our input vector. The SVM aims to find a value of the parameters that maximize the geometric margin. We assign a data sample to a class based on the value of $f(\cdot)$, and $f(X_i) < 0$ represents a jamming attack in the network.

Naive Bayesian (NB): It is a technique based on the Bayes' theorem considering that all features are independent, and each one of them contributes to the probability that defines the class variable. The probability of a jamming attack can be defined as $Pr(Y = 1|f_1, ..., f_{2W}) = \frac{Pr(Y=1)Pr(f_1,...,f_{2W}|Y=1)}{Pr(f_1,...,f_{2W})}$. As the denominator is not dependent on the class variable, we focus only on the numerator. As a result, we get $Pr(Y = 1|f_1, ..., f_{2W}) \backsimeq Pr(Y = 1)Pr(f_1, ..., f_{2W}|Y = 1) \backsimeq Pr(Y = 1)Pr(f_1|Y = 1)Pr(f_2, ..., f_{2W}|Y = 1, f_1)$, and so on. The training consists on computing the probability of attack given a particular feature.

III. Resource Reallocation for Attack Prevention

Once the attack is detected, the information about it can be collected and analyzed for the purpose of attack prevention. We define prevention not as an ability to keep a network free from attacks, but make it robust in presence of attacks. This robustness we define as the ability to minimize the number of channels affected by an attack. To this end, we propose to periodically change the optical resources allocated to maintain optical channels between a source-destination pair, called as lightpaths (LPs), using a resource reallocation strategy. The proposed attack prevention scheme is a proactive approach, in which the control and management planes can inquire about the network status regularly with mean interarrival time τ_r, and as a response the attack detection tool sends the likelihood of a network being attacked with probability P_a. When the information regarding the localization of attacks is provided by an attack detection tool, then it is easy to handle such unauthorized attacks. However, a network-wide attack monitoring is generally complex, and exact locations of attacks (i.e., unauthorized LPs) might be difficult to identify due to amplification noise, (de)multiplexing processes, power coupling, non-uniform transmission distances, etc. Thus, the attack prevention scheme presented here utilizes only the attack detection probability P_a, and based on this information it tries to reallocate optical resources to LPs, which could be authorized or unauthorized, such that an attacker or a group of attackers should not be able to jam or have unauthorized access (by means of crosstalk) continuously to authorized lightpaths.

Algorithm 1 describes our wavelength (re)assignment approach for WDM optical networks. We utilize a K-shortest paths routing and a first-fit (FF) wavelength assignment scheme, which assigns a first available wavelength to a LP request requiring a single wavelength (step 1). A wavelength is assigned with a continuity constraint, which means that a new

Algorithm 1 Attack Prevention Scheme

```
1: A LP request's arrival: Assign a first available wavelength on an ordered set of $K$ shortest paths. Block it when no wavelength assignment is possible.
2: Reallocation requested: Using the attack detection probability $P_a$, reassign wavelengths to all network LPs in a random fashion.
3: When reassignments of all LPs are not possible, then keep the original wavelength assignment, and scramble wavelengths assigned to LPs between each node pair.
4: Security Measure: Calculate the probability of jamming $P_j$, and a probability of jamming without an interruption $P_c$ using all LPs, and obtain the security metric by (1).
```

LP must be assigned to the same wavelength on each link of its traversing route. When a LP request does not find a continuous free wavelength then it is blocked. In step 2, we consider another type of request, namely a resource reallocation request, to review security of the network. It arrives regularly with mean time τ_r, and the network decides whether to perform reassignment of wavelengths or not with probability P_a. However, a randomized reassignment of wavelengths to LPs is not feasible always, since routing and wavelength assignment is known to be a NP-hard problem. In such situations, we keep the original wavelength assignment, and simply scramble wavelengths of LPs between each source destination node pair in step 3. Finally, to evaluate the proposed scheme, in step 4 we measure a security metric, which is described below.

Let us assume that the reallocation requests arrive with exponentially distributed mean interarrival time τ_r. If we assume that the holding time of LPs is also exponentially distributed with mean τ_c, then it can be easily shown that the number of arrivals of reallocation requests during the lifetime of an authorized LP is $\frac{\tau_c}{\tau_r}$. However, a reallocation process is only performed with the attack detection probability P_a, which means the average number of reconfigurations performed during the lifetime of the LP would reduce to $N_r = \frac{\tau_c}{\tau_r} P_a$. As N_r number of reconfigurations of the LP during its lifetime τ_c can be divided into $N_r + 1$ parts, each part is approximately given as $\Delta t = \frac{\tau_c}{N_r+1}$ [6]. Assuming that one or more attackers can jam or observe authorized LPs in their proximity, that means to left and right of unauthorized LPs. The amount of successfully jammed data of an authorized LP with a data rate of b bits per unit time in the first interval is $b\Delta t \times P_j$, where P_j is the probability that a LP is jammed in an interval Δt. Furthermore, in each of the following N_r intervals, the amount of successfully jammed data of the LP is $b\Delta t \times P_j \times P_c$, where P_c is the probability that the LP is jammed without an interruption at each reallocation. We can obtain these probability terms (P_j and P_c) by simulating all (re)allocation instances in an event simulator. Therefore, the amount of data successfully jammed in the LP's lifetime is $b\Delta t \times P_j \left[1 + N_r \times P_c\right]$. Moreover, the amount of data carried by the same LP during its lifetime is $b \times \tau_c = b\Delta t(N_r + 1)$. Thus, we define a security metric by finding the probability of unsuccessful jamming of an authorized LP, which is given by one minus the ratio of successfully jammed data and total carried data per authorized LP, and it is obtained by Eq. (1).

$$\Lambda_j = 1 - P_j \frac{1 + \frac{\tau_c}{\tau_r} P_a P_c}{1 + \frac{\tau_c}{\tau_r} P_a} \qquad (1)$$

IV. Numerical Analysis

A. Application Scenario and Data Extraction

To generate an experimental dataset, we use an Optisystem software to simulate the detection problem in a WDM network scenario shown in Fig. 3, in which the received signals come from different light sources and propagate through different fiber paths. The WDM network is composed of 4 WDM transmitters with 8 channels each. The starting channel frequency is 193.1 THz and the frequency spacing is equal to 100 GHz. Each transmitter is connected to a WDM Mux with 1 dB insertion loss. The power of the multiplexed light signals coming from the first and fourth WDM transmitters is amplified with EDFAs of 5 m length, and two EDFAs of 2 m are used to amplify the power of the light signals coming from the second and third WDM transmitters. We set the noise center frequency of each EDFA to the frequency corresponding to central channel of the multiplexed wavelengths. Thereafter, the amplified signals coming from EDFA 1 and EDFA 4 are connected to an optical fiber of 50 km length with 0.28 dB/km attenuation loss. The light from the fiber 1 is combined with the amplified light from EDFA 2 using a power combiner. Similarly, we combine the lights coming from fiber 2 and EDFA 3. After each combiner, the lights are amplified using an EDFA of 5 m length then they travel a 70 km optical fiber. Afterward, we combine the two lights with a power combiner to get a 32 multiplexed channels. Finally, we amplify the resulting light which travels an optical fiber of 50 km before being demultiplexed by a WDM Demux and connected to receivers. We designate an optical transmitter as a Jammer, which inserts unauthorized signals with higher power in one of the channels of one of the 4 WDM Mux. A WDM Analyzer is used to extract the data related to each channel before the WDM Demux. As explained in Sec. II, we focus on two features – the signal power and OSNR, which are measured by the WDM Analyzer. The power of each transmitter is set to a value between -22 and 0 dBm so that the received power of channels are close to each others. The power of transmitters is varied in the defined range and more power is given to the signals which travel longer paths (relative to the 1st and 4th transmitters). We use an Optisystem simulator to simulate various instances of authorized and unauthorized data samples, where each instance of generating a data sample takes around 20 to 25 minutes. The transmission power is varied for generating 3 types of datasets that store different percentage of authorized and unauthorized samples: (50%–50%), (70%–30%) and (90%–10%). The number of data points in each dataset is 300. The dataset is divided into training and testing sets with proportion of 75% and 25% respectively.

B. Performance

In this subsection, we first evaluate the overall accuracy of different detection algorithms (ANN, DT, KNN, SVM, LR and NB) and analyze its two main components: True Positive (TP) rate as the probability of correct detection of jamming attacks, and True Negative (TN) rate as the right recognition of the normal behavior. We present the accuracy of each ML algorithm by averaging training and testing results obtained with 20 different seeds. All algorithms are implemented in Python and use Scikit-learn library. We use a fully connected neural network with one hidden layer, and Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) algorithm as an optimizer. We consider only one neighbor for KNN, and the inverse of regularization strength is set to 0.1 in LR. Moreover, we considered the Gaussian NB, and use the default parameters provided by scikit-learn implementation of SVM

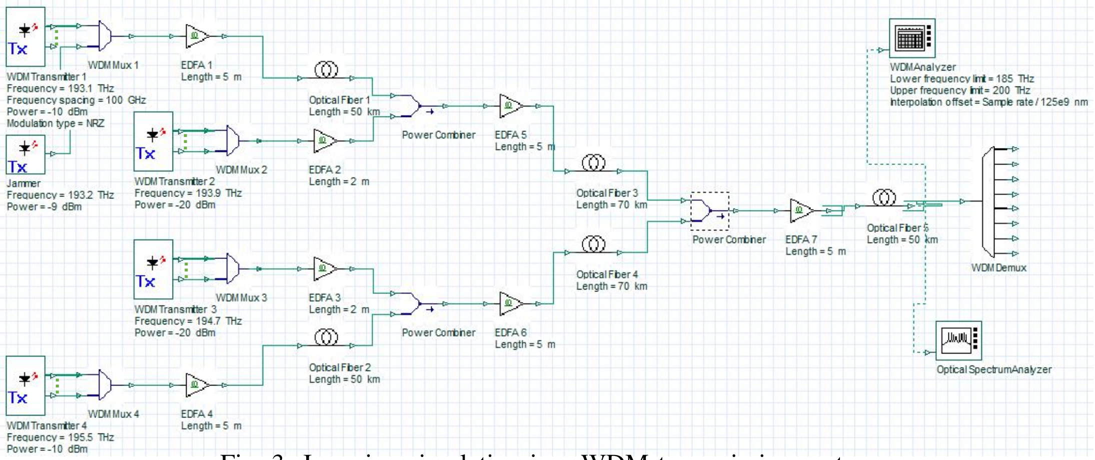

Fig. 3: Jamming simulation in a WDM transmission system.

and DT. Finally, this subsection evaluates the performance of the reallocation scheme.

Fig. 4 depicts the performance of different algorithms in terms of accuracy, TP rate and TN rate with equal proportion of authorized and unauthorized data samples, i.e., 50-50% sampling. The results show that the accuracy is higher when we use only power as a feature, since the attack is realized by a Jammer which inserts a higher power to disrupt the authorized channels by causing nonlinearities in the fiber. Therefore, the signal power is enough to model the jamming attack in optical networks and the use of another feature (e.g., OSNR) or both features (power and OSNR) confuses the detection algorithms. When we use the signal power as a lone feature, we see in Fig. 4 that ANN and LR outperforms other algorithms with an accuracy of 100%. KNN and DT show an accuracy of 85%. SVM has an accuracy of 80% which is a bad performance compared to the previous algorithms. The NB technique is the worst candidate for detecting jamming attacks with an accuracy of 60% and a TN rate equal to 55%.

In Fig. 5 we compare the performance of ML algorithms considering two other datasets of 70-30% and 90-10% sampling, where the received power is used as the only feature. We observe that ANN outperforms all other algorithms in different scenarios with an accuracy equal to 98%. NB is still the worst technique for the detection problem in terms of accuracy but in the case of 90-10% sampling it gives better results than other algorithms in terms of the TN rate which makes it a possible candidate for a hybrid detection design. The accuracy of all other algorithms (DT, KNN, SVM and LR) is around 95%, and their TN rates slightly decrease for 70-30% sampling and sharply decrease in the case of 90-10 % sampling. For SVM, the TN rate is equal to 46% for 70-30% sampling and 51.68% for 90-10% sampling. In general, ANN is the only algorithm in the evaluated scenario that has a stable performance with different sampling strategies. For other algorithms, the TN rate is lower than 80% in 90-10% sampling which could be due to insufficient amount of unauthorized data being used for training. Furthermore, from Figs. 4 and 5 we observe that the accuracy of ML algorithms could increase when the amount of authorized samples is increased in the dataset. The reason is that the TP rates increase with increasing proportions of authorized samples (from 50% to 90%). On the other hand, the TN rate decreases sharply for most of ML algorithms when the percentage of unauthorized signals is decreased in the dataset (from 50-50% to 90-10%). When considering the TN rate as a metric, again ANN shows the best performance with a high TN rate equal to 100% for 50-50% and 70-30% sampling, and equal to 75% for 90-10% sampling. However, the superior performance of ANN comes with a higher training complexity as compared to other evaluated ML techniques.

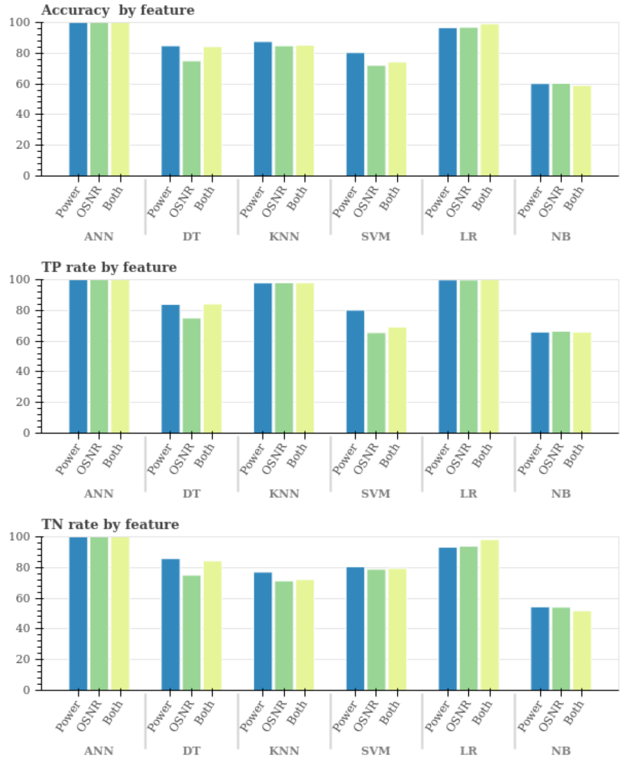

Fig. 4: Detection accuracy per algorithm for 50-50% sampling.

Finally, we present the effectiveness of our attack prevention mechanism in Fig. 6, which illustrates the effect of the mean

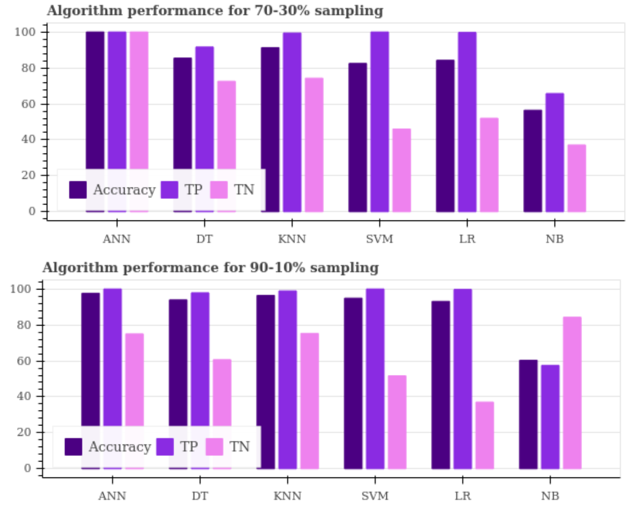

Fig. 5: Performance of algorithms with different sampling.

reallocation time (τ_r) and detection accuracy (P_a) on the security of lightpaths (Λ_j), as defined by Eq. (1). This result is obtained in a well known 14-node NSFNET topology with two shortest paths per node pair, and the number of wavelength channels per link is 80. We generated two types of lightpath (LP) requests: authorized and unauthorized. The total LP requests are uniformly divided into node pairs, and they are generated with exponential mean 200 requests per unit time, and their holding times are exponentially distributed with mean $\tau_c = 10$ time units. Furthermore, the requests (authorized and unauthorized) are classified into $10 : 1$ ratio. Remember that, as discussed in Sec. III, unauthorized LPs are used by attackers to jam the data being carried by the authorized LPs over the network. On the other hand, the reallocation requests are generated with exponential mean τ_r, and it is varied as shown in the horizontal axis in Fig. 6. We obtained the probability of jamming per reallocation interval (P_j) as the ratio of number of jammed interval per authorized LP to the total intervals through a discrete event simulation. Similarly, the probability of jamming without interruption in a reallocation interval is calculated by the ratio of the number of instances where reallocation effect is absent (i.e., a LP is jammed without interruption) to the total reallocation instances. We use these probability terms in Eq. (1) to obtained the security metric Λ_j.

In Fig. 6, we see that the data security of LPs against jamming decreases exponentially with the mean reallocation time τ_r. Furthermore, when the mean reallocation time is equal to the mean lifetime of authorized LPs, i.e., $\tau_r = \tau_c = 10$, the security decreases only by 3% for detection accuracy of 100%, which means even one reconfiguration in the lifetime of a lightpath is enough to secure 97% of lightpaths against jamming attacks. However, with lower attack detection accuracy and with increasing number of unauthorized lightpaths in the network, the security against the jamming attacks decreases. Thus, the attack detection and resource reallocation are very important to prevent attackers from jamming legitimate data channels in optical networks.

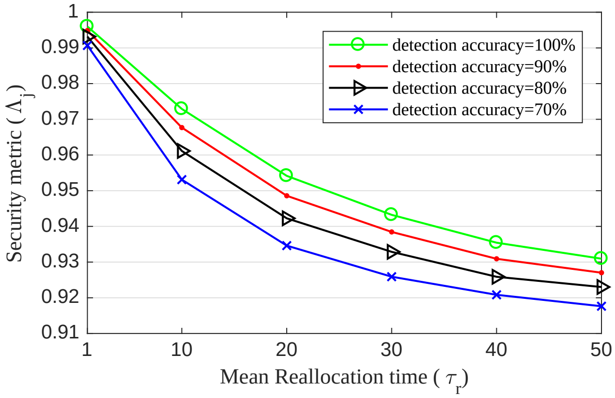

Fig. 6: Unsuccessful jamming probability in NSFNET.

V. Conclusion

In this paper, we studied the effectiveness of various machine-learning techniques: artificial neural networks, support vector machine, logistic regression, K-nearest neighbors, decision tree and Naive Bayesian, for detecting power jamming attacks. Moreover, we provided a comparative and detailed analysis of these techniques using different metrics, such as attack detection accuracy, true positive rate and true negative rate. We find that a fully connected neural network is the most accurate approach in detecting out-of-band power jamming attacks. Furthermore, we apply a resource reallocation scheme to take a preventive action against jamming attacks, and which is shown to improving data security in optical networks.

References

[1] M. Medard, S. R. Chinn, and P. Saengudomlert, "Attack detection in all-optical networks," in *OFC, Technical Digest*. IEEE, 1998.

[2] Y. Huang *et al.*, "Dynamic mitigation of edfa power excursions with machine learning," *Optics express*, vol. 25, no. 3, pp. 2245–2258, 2017.

[3] P. R. Prucnal, M. P. Fok, Y. Deng, and Z. Wang, "Physical layer security in fiber-optic networks using optical signal processing," in *Communications and Photonics Conference and Exhibition (ACP), 2009 Asia*, vol. 2009. IEEE, 2009, pp. 1–10.

[4] X. Hu *et al.*, "Chaos-based partial transmit sequence technique for physical layer security in ofdm-pon," *IEEE Photonics Technology Letters*, vol. 27, no. 23, pp. 2429–2432, 2015.

[5] Y. Li, N. Hua, Y. Song, S. Li, and X. Zheng, "Fast lightpath hopping enabled by time synchronization for optical network security," *IEEE Communications Letters*, vol. 20, no. 1, pp. 101–104, 2016.

[6] S. K. Singh, W. Bziuk, and A. Jukan, "A combined optical spectrum scrambling and defragmentation in multi-core fiber networks," in *Communications (ICC), IEEE International Conference on*. IEEE, 2017.

[7] Y. Li, N. Hua, Y. Yu, Q. Luo, and X. Zheng, "Light source and trail recognition via optical spectrum feature analysis for optical network security," *IEEE Communications Letters*, vol. 22, no. 5, pp. 982–985, 2018.

[8] J. Mata, I. de Miguel, R. J. Durán, N. Merayo, S. K. Singh, A. Jukan, and M. Chamania, "Artificial intelligence (ai) methods in optical networks: A comprehensive survey," *Optical Switching and Networking*, 2018.

[9] J. Thrane, J. Wass, M. Piels, J. C. Diniz, R. Jones, and D. Zibar, "Machine learning techniques for optical performance monitoring from directly detected pdm-qam signals," *Journal of Lightwave Technology*, vol. 35, no. 4, pp. 868–875, 2017.

[10] S. K. Singh and A. Jukan, "Machine-learning-based prediction for resource (re) allocation in optical data center networks," *Journal of Optical Communications and Networking*, vol. 10, no. 10, pp. D12–D28, 2018.