# Real-Time Noise Identification in DSL Systems Using Computational Intelligence Algorithms

Farias, F. S., Borges, G. S., Roberto M. Rodrigues, Santana, Á. L., and Costa, J. C. W. A., *IEEE Member*
Universidade Federal do Pará, Belém, Brazil
{fabriciosf, gilvan, menegues, adamo, jweyl}@ufpa.br

***Abstract* — Despite the advances and improvements in the Digital Subscriber Line (DSL) technology, noise is still the main impairment. In special, far-end crosstalk, Radio Frequency Interference (RFI) and Impulsive Noise (IN) are of greatest concern and study. In DSL world, there are many noise mitigation techniques, but to know the impairment as a priori knowledge is a step necessary to apply the appropriate technique. In this paper we propose a new methodology for noise identification on real-time. Computational Intelligence (CI) algorithms are used in order to classify in real time the absence of noise or the predominance of IN, crosstalk or RFI. The algorithms are applied to a database composed by management information base (MIB) metrics. In order to ensure the database diversity, several DSL topologies using real cables were created and evaluated. In order to choose the best CI algorithm, a benchmarking was performed comparing the results achieved by naive Bayes, Bayesian belief networks and artificial neural networks based on backpropagation and on Radial Basis Function (RBF). The results demonstrate the potential use of CI for noise identification in DSL networks through MIB metrics and the most difficult noise to be identified is pointed. Tests indicate the RBF algorithm achieving the best result with 99.6% of accuracy.**

***Index Terms*— DSL, data mining, noise identification, real time systems, monitoring, network measurement, backhaul.**

## I. Introduction

DIGITAL *Subscriber Line* (DSL) technologies employ metallic cables commonly used in telephone systems for high-speed data transmission. There are today more than 1.3 billion copper phone line connections, which are important in the modern world of telecommunications, a growing 1/3 of them are now using DSL [1]. Despite the increasing use of fiber optics and architectures such as fiber-to-the-cabinet and fiber-to-the-home, DSL technologies for backhaul data transmission are still advantageous in terms of cost in last mile solutions, i.e., femtocells, domestic and/or corporate users.

The most recent DSL technologies under operation, Very-high-bit-rate DSL (VDSL) and VDSL2 operate in short links on the order of hundreds of meters, i.e., up to 300 meters providing maximum communication speed of 100 mbps. In general this communication is between the distribution cabinet and the user equipment or femtocell modems. Although the progress in the DSL technologies has been increasing, the noise impact is still the main concern.

There are several noises in communications, but in DSL: far-end crosstalk (crosstalk), Radio Frequency Interference (RFI) and Impulsive Noise (IN) are of greatest concerns of the study.

In an environment where several active DSL lines are in the same binder, like in residential deployments or backhauling purposes, it is crosstalk noise that severely restricts the system performance. Moreover, among many types of noise, the crosstalk noise is more predominant in DSL networks [2]. Besides, it is the major performance bottleneck for the large deployment number of DSL lines trying to achieve high-speed data rate [3], causing problems of signal loss, and decreasing the channel capacity.

In industrial areas and residential premises, IN becomes a relevant restricting factor. Furthermore, IN is the very complex and hard to be characterized; and, it also causes damages, such as loss of connection, decrease of link range and loss of service quality.

Finally, RFI is mostly caused by radio broadcast in low frequency, as well as interference in the hams range may cause non-linear distortions.

The bibliography related to noise identification generally focus on crosstalk. Perhaps the most natural way to identify crosstalk is from its perceptible impact on the Signal-to-Noise ratio (SNR) [4-5], which can be determined by retrieving information from customer premises equipment (CPE). Some methods employ the classical least-mean approach to estimate the dominant crosstalk users [6] or the crosstalk sources in a binder. Other methods provide real-time exploiting of signaling exchange on the training phase of a recently activated DSL line to determine the crosstalk function between that line and an existing operational one [5]. Computational Intelligence (CI) algorithms based on sampling and clustering have been proposed in order to estimate IN, as a step for cancelling [7-8].

Noise identification in DSL system has attracted a lot of attention due the significant benefits of having an accurate description from network limitation factors. At this time, it is noticed a lack of material about real-time noise monitoring in DSL networks. Following that trend, this paper proposes a new strategy for real-time noise monitoring (three different noises) in DSL networks. CI algorithms were applied in order to identify the predominant noise in the network.

This paper is structured as following: section II presents the DSL topologies considered; section III shows the Knowledge Discovery in Database (KDD) steps applied in this paper; section IV presents the results obtained with the

This work was supported in part by the CNPq, CAPES, UFPA and Ericsson Telecomunicações Research and Development Center S.A. Brazil.

978-1-4799-1089-2/13/$31.00 ©2013 IEEE

analyzed algorithms; and the last section presents the results obtained in this study.

## II. TOPOLOGY ARCHITECTURES

Figure 1 shows the setup used during the network measurements. The information generated from the measurements was taken by the MIB, which are a set of data returned by the modem and DSL Access Multiplexer (DSLAM). The MIB metrics acquire the necessary information for the network management. Such metrics are managed by the Simple Network Management Protocol (SNMP), which is an application layer protocol responsible for the network management, allowing its performance evaluation and specific settings modification.

During the DSL measurements phase, MIB metrics (attributes) were collected every 15 seconds, resulting in approximately 11 hours of measurements. The database was built from measurements made of different network topologies varying the noise power in order to ensure the diversity in data.

The network measurements were done using copper wires, which are represented by the dotted lines. The continuous lines are Ethernet cables used to transmit the data from the traffic generator/analyzer (TGA) to the DSLAM or central office and from the DSL modem or customer premises equipment to TGA. TGA is hardware that generates triple-play broadband traffic.

The dashed lines connecting Ethernet cables from the DSLAM to Switch are used for transmitting data under test and managing the test itself. PC1 and PC2 are connected computers that manage the tests.

The topologies were created in order to cover the short representative links of a VDSL2 network [9], topologies with 0.4 mm and 0.5 mm gauges and 50, 150 and 450 meters lengths were used.

Table I summarizes the twenty-four topologies created. Six are reserved for measurements without noise injection, whereas the other eighteen uses noise injection to simulate DSL networks under disturbances influence.

TABLE I
REPRESENTATION OF TOPOLOGIES AND NOISES.

| Loop | Noise Type | Length (meters) |
|---|---|---|
| Loop 1 (0.4 mm) | Crosstalk | 50, 150 and 450 |
| | IN | |
| | RFI | |
| | No Noise | |
| Loop 2 (0.5 mm) | Crosstalk | |
| | IN | |
| | RFI | |
| | No Noise | |

In order to build a database for noise identification task, predefined noise masks from International Telecommunication Union (ITU) [10] recommendations were used. The noises were inserted in the DSLAM side using the noise generator (NG) represented in Figure 1.

The NG calculates the noise sample of the noise profiles that are available on the generator. A noise profile injected by NG has the power spectral density (PSD) description of crosstalk, IN and RFI noise.

In order to guarantee database diversity the noise power was varied. Firstly, IN was injected with 0 dBm. Secondly, RFI with -54 and -44 dBm were injected separately. Third, Crosstalk was injected and varied in following powers -25.3, -23.4, -22.4, -21.6, -21.1, -20.6, -20.2, -19.8, -19.5, until -19.3 dBm. Finally, it was measured an experiment without noise injection by NG, named No Noise.

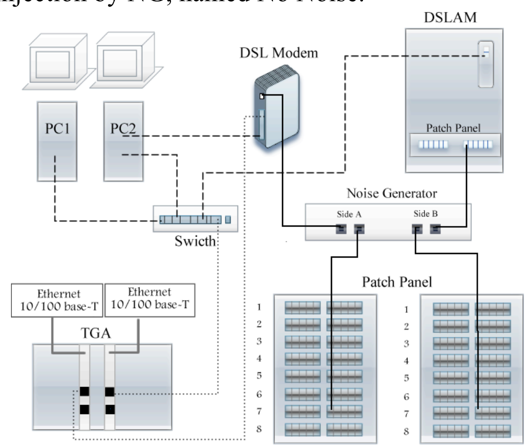


Fig. 1. Topology and devices used on this work.

## III. KDD APPLICATION

Figure 2 presents the steps of the KDD [11] process applied to this study. It ranges from the database creation to the extraction of knowledge by the application of a CI or statistical technique in the data mining [12] process.

In the step of the database building, measurements without noise injection were made, as well as with crosstalk, impulsive and RFI noise injection.

The metrics selection step was done to withdraw incomplete data from database. A correlation analysis was applied in the preprocessing step in order to remove registers and attributes correlated. Finally, the reduced database was normalized in the transformation step. After the initial KDD steps, only the following attributes were used as input on the four algorithms:

- adslAtucCurrSnrMgn and adslAturCurrSnrMgn are the noise margin as seen by the ATU-Central (ADSL Termination Unit-Central) and ATU-Remote, respectively;
- adslAturCurrOutputPwr is the total output power measured and transmitted by the ATU-Remote.
- adslAtucChanInterleaveDelay defines the mapping (relative spacing) between subsequent input bytes at the interleave input and their placement in the bit stream at the interleave output.
- adslAtucChanPerfCurr1DayUncorrectBlks is the count of all blocks received with uncorrectable errors on the channel during the current day;
- adslAtucPerfCurr1DayESs is the count of errored seconds during the current day;

• adslAtucChanPerfCurr1DayCorrectedBlks is the count of all blocks received with errors that were corrected on the channel during the current day;

• adslAtucPerfXCurr1DayEcs and adslAturPerfXCurr1DayEcs are the error count during the time in seconds as seen by the ATU-Central and ATU-Remote, respectively;

• adslAtucPerfCurr1DayUasL and adslAturPerfCurr1DayUasL are the number of seconds which lines are unavailable as seen by the ATU-Central and ATU-Remote, respectively;

• adslAturPerfCurr1DayLprs is the count of the number of seconds when there was loss of power during the current day;

• adslAtucPerfCurr1DayInits is the count of the line initialization attempts in the day.

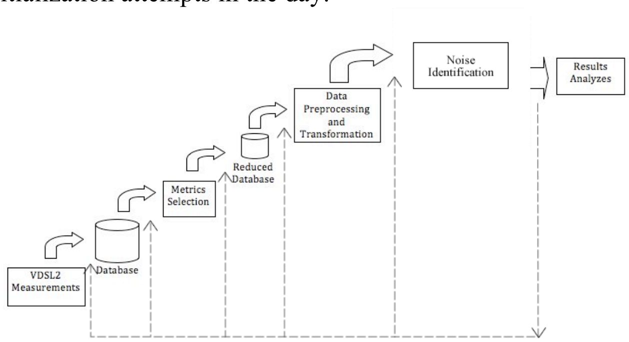


Fig. 2. Flowchart depicting the KDD process extended to our application.

In selection and preprocessing steps 40, 77, 83 and 40 registers were discharged from No Noise, Crosstalk, IN and RFI respectively. This reduction resulted in a database of 2654 samples, being 244, 1933, 117 and 360 from No Noise, Crosstalk, IN and RFI, respectively.

Finally, the data mining algorithms were applied and the results interpretations were conducted during the step of result analysis.

Four classification algorithms were tested: Naïve Bayes (NB), Bayesian Networks (BN), Artificial Neural Networks (ANN) based on Radial Basis Functions (RBF) and Backpropagation (BP) [13-14] for benchmarking purposes.

## IV. RESULTS

In this section the best results are presented and discussed. The result analysis in Table II was done in four phases. The first analysis regards the relative performance of the algorithms for detecting the presence of noise. The second analysis regards weighted average of accuracy among the kind of noises present on the experiments. The third analysis presents the accuracy percentage during the best simulation of each algorithm. The final analysis presents the most difficult noise to be identified.

The confusion matrix (Table II) shows the classification accuracy of the algorithms used. The left columns show the algorithms and the desired output (Actual Noise Environment). The rows represent the output after tests from each algorithm.

For detection of noise purposes, the Table II can be simplified as shown in Table III. In Table III crosstalk, IN and RFI are considered as Noise and No Noise is kept. The main diagonal indicates the correct detection of the noise environment (true positive and true negative) while the associated secondary diagonal indicates the misdetection (false positive and false negative).

TABLE II
CONFUSION MATRIX OF BP, RBF, BN AND NB.

| Algorithms | Actual Noise Environment | Algorithm's output | | | |
|---|---|---|---|---|---|
| | | No Noise | Crosstalk | IN | RFI |
| BP | No Noise | 244 | 0 | 0 | 0 |
| | Crosstalk | 1 | 1932 | 0 | 0 |
| | IN | 0 | 7 | 110 | 0 |
| | RFI | 6 | 0 | 0 | 354 |
| RBF | No Noise | 244 | 0 | 0 | 0 |
| | Crosstalk | 1 | 1931 | 1 | 0 |
| | IN | 0 | 0 | 117 | 0 |
| | RFI | 8 | 0 | 0 | 352 |
| BN | No Noise | 244 | 0 | 0 | 0 |
| | Crosstalk | 31 | 1902 | 0 | 0 |
| | IN | 0 | 0 | 117 | 0 |
| | RFI | 2 | 0 | 0 | 358 |
| NB | No Noise | 244 | 0 | 0 | 0 |
| | Crosstalk | 1 | 1901 | 1 | 30 |
| | IN | 0 | 0 | 117 | 0 |
| | RFI | 9 | 30 | 0 | 321 |

Regarding the noise detection error (false negative), all algorithms performed well. However, BP, RBF and NB achieved the false negative rate lower than 0.5% (0.26 %, 0.33 % and 0.37%, respectively) while BN performed relatively worse – i.e., 1.24%. From Table III, it is also evident that algorithms do not have a conservative behavior since none of “No Noise” data was wrongly assigned as a noisy environment (false positive 0% or true negative 100 %).

In summary, RBF, BP and NB were the algorithms that performed the best noise detection.

Regarding True Positives (TP) and False Positive (FP) Rate, the results for each algorithm are presented in Table IV. It is important to notice that in order to calculate the mean detection accuracy, the number of samples must be considered, i.e., a weighted computation must be done. Crosstalk was considered the more relevant class due the quantity of samples in the database while IN was the opposite.

RBF and BP algorithms achieved the best TP Rate, with very similar rates. For the studied case the difference is not significant to choose BP over RBF.

RBF algorithm achieved the best TP Rate and the minor FP Rate being the best algorithm comparing with the other three. BP achieved a result close to RBF, but worse. However, as shown in Table III, BP classified seven noise samples as No Noise, when RBF classified nine. For the studied case the difference is not significant to choose BP over RBF. NB and

BN did not achieve better accuracy when compared with BP and RBF.

TABLE III
CONFUSION MATRIX BINARY OF BP, RBF, BN AND NB.

| Algorithm | Actual noise environment | Algorithm's classification | |
|---|---|---|---|
| | | No Noise | Noise |
| BP | No noise | 244 | 0 |
| | Noise | 7 | 2403 |
| RBF | No noise | 244 | 0 |
| | Noise | 9 | 2401 |
| BN | No noise | 244 | 0 |
| | Noise | 33 | 2077 |
| NB | No noise | 244 | 0 |
| | Noise | 10 | 2400 |

When benchmarking the results, all algorithms achieved accuracy rates greater than 97%. However, among the four simulated algorithms, the RBF and BP are highlighted with accuracy of 99.6% and 99.4%, respectively. On the other hand, NB and BN achieved least expressive results, with 97.32% e 98.75% only. In summary, RBF and BP were the algorithms that performed the best noise identification.

TABLE IV
WEIGHTED AVERAGE OF ACCURACY BY THE TP RATE AND FP RATE.

| Algorithms | Weighted Average of Accuracy | |
|---|---|---|
| | TP Rate | FP Rate |
| BP | 99.5% | 0.7% |
| RBF | 99.6% | 0 |
| BN | 98.8% | 0.1% |
| NB | 97.3% | 3.3% |

Analyzing among the noises, all No Noise samples were correctly classified using the algorithms. Crosstalk classification achieved best results with neural network and NB algorithms. Only BP misclassified IN samples. Finally, due its low power, RFI samples were mistaken with No Noise in all algorithms.

Among the four desired output No Noise was well classified with 0% of error, while RFI, IN and Crosstalk achieved an error average by all algorithms of approximately 3.81, 1.49 and 0.46 percent, respectively.

## V. CONCLUSION

Noise identification is a necessary step in order to choose the most appropriate technique to mitigate noise.

It was demonstrated that is possible to monitor and to identify real-time noise using information from the application layer provided by MIB metrics. Also, it was shown that is possible to have a generalized solution based on CI algorithm to identify noise independently of topology.

Through KDD steps were possible to decrease the error rate and improve the classification results.

The methodology does not interfere in the data transmission services deployment (as it uses information strictly measured by the central office and residential modem). In addition, the methodology is not dependent on the physical modeling of communication channel. It dependents only on information obtained in the application layer.

The results demonstrate the high potential of CI algorithms for monitoring purposes. The best technique, the RBF, achieved 99.6% of accuracy. RBF produced the best results in all the comparisons.

The RFI showed as the most difficult type of noise to identify by CI algorithms due to its low power level. The RFI is the most likely to be confused with a measure without noise injection due to its low impact in the network.

The information about the predominant noise can be used not only for its estimation and mitigation, but also for a better management and development of new strategies to improve the DSL deployment. This strategy can be expanded to different networks that make use of MIB metrics.

## REFERENCES

[1] Cioffi, J.M.; , "Lighting up copper [History of Communications]," *Communications Magazine, IEEE* , vol.49, no.5, pp.30-43, May 2011
[2] D. Zanatta Filho, R. R. Lopes, R. Ferrari, M. B. Loiola, R. Suyama, G. C. C. P. Simões, and B. Dortschy. "Achievable rates of DSL with crosstalk cancellation", *European Trans. on Telecommunications*, vol. 20, pp. 81-86, 2009.
[3] S. M. Zafaruddin, S. Prakriya, and S. Prasad, "Performance Analysis of Zero Forcing Crosstalk Canceller in Vectored VDSL2", *IEEE Signal Process. Letters*, vol. 19, pp. 219-222, April 2012.
[4] M. Guenach, J. Louveaux, L. Vandendorpe, P. Whiting, J. Maes, and M. Peeters. "Performance analysis of the signal-to-noise ratio assisted crosstalk channel estimation for DSL systems", i*n Proc. of IEEE International Conf. on Communications 2009*, June 2009.
[5] N. Papandreou, and T. Antonakopoulos, "Far-end crosstalk identification method based on channel training sequences", *IEEE Trans. on Instrumentation and Measurement*, vol.54, no.6, pp. 2204-2212, Dec. 2005.
[6] M. L. Gujrathi, J. Homer, I. Vaughan, L. Clarkson, R. Cendrillon, and M. Moonen, "A Detection Guided Normalized Least-Mean-Squares Adaptative Partial Crosstalk Canceller for Multi-User DSL Enviroments", *IEEE Signal Process. Letters*, vol. 16, no. 6, pp. 489-492, June 2009.
[7] Al-Naffouri, T.Y.; Quadeer, A.A.; Al-Shaalan, F.F.; Hmida, H.; , "Impulsive noise estimation and cancellation in DSL using compressive sampling," *Circuits and Systems (ISCAS), 2011 IEEE International Symposium on* , vol., no., pp.2133-2136, 15-18 May 2011.
[8] Al-Naffouri, T.Y.; Quadeer, A.A.; Caire, G.; , "Impulsive noise estimation and cancellation in DSL using orthogonal clustering," *Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on* , vol., no., pp.2841-2845, July 31 2011-Aug. 5 2011 .
[9] P. Golden, H. Dedieu, and K. Jacobsen, "Fundamentals of DSL Technology" (1 ed.), Auerbach Publications, July 2004.
[10] ITU-T Recommendation G.993-2. "Series G Transmission Systems and Media, Digital Systems and Networks", Feb-2006.
[11] U. Fayyad, G. P. Shapiro, and P. Smyth, "From Data Mining to Knowledge Discovery in Databases", *AI Magazine American Association for Artificial Intelligence*, Fall 1996.
[12] I. H. Witten and E. Frank. "Data mining - practical machine learning tools and techniques". St. Louis, Elsevier, 2005.
[13] Finn V. Jensen and Thomas D. Nielsen, "Bayesian Networks and Decision Graphs", Springer Verlag, Second Edition, 2007.
[14] Simon O. Haykin, "Neural Networks and Learning Machines", Prentice Hall, Third Edition, 2008.