A Set of Tools to Generate Intelligent Agents for Fault Prediction in Optical Rerouting

Carlos Hairon R. Gonçalves[1], Carlos Eduardo A. Vieira[2], Mauro Oliveira[1] and Rossana M. C. Andrade[2]

[1]Laboratório Multiinstitucional de Redes de Computadores e Sistemas Distribuídos
Centro Federal de Educação Tecnológica do Ceará (LAR/CEFET-CE)
Av. 13 de Maio, 2081, Benfica, Fortaleza, Ceará, Brasil, 60.531-040.

{hairon,mauro}@lar.cefet-ce.br

[2]Universidade Federal do Ceará, Departamento de Computação (DC/UFC)
Campus do Pici, Bloco 910, Fortaleza, Ceará, Brasil, 60.455-760.

{carllos,rossana}@lia.ufc.br

Abstract. *One of the issues that constrains the use of GMPLS protocol in DWDM links is the slow rerouting provided by GMPLS over DWDM networks.The 1+1 protection mechanism is an approach to solve it, where a backup link is employed as mirror to provide an immediate fault recovery. However, such solution is costly since a given channel must exclusively be allocated. Intelligent agents can execute the task of monitoring an optical link of a network, and aiming to predict anomalous situations, so that pro-active measures can be taken before faults occur. Thus, this paper presents a set of tools for generating such intelligent agents using Artificial Neural Networks. The use of intelligent agents leads to an algorithm for fault prediction with few computation resources and low complexity.*

1 Introduction

Software Agents have been used in different kinds of applications like network management, automation and data systems. When software agents use artificial intelligence techniques, they are called intelligent agents. Such agents can predict situations as traffic increase or link failure allowing pro-active measures before the problem gets complicated. The use of Intelligent Agents make possible to provide a fast rerouting mechanism for GMPLS (*Generalized Multi-Protocol Label Switching*) over DWDM (*Dense Wavelength Division Multiplexing*) networks.

This paper is then organized as follows. Section 2 presents related works to fault prediction in optical links. In section 3 the set of tols is described. First, the simulated optical system is described in section 3.1. Then, in section 3.2 the ANNs (*Artificial Neural Networks*) used in this work are described. Section 3.3 discusses not only the results for each ANN proposed in 3.2 but also an intelligent agent prototype. Finally, in section 4presents our conclusions and future works.

2 Related Works

Kartalopoulos [1] presents a book that deals with faulty in DWDM networks exclusively. Carmen Mas [2] develops an algorithm for location of faulty WDM systems. Both works propose techniques of location and treatment of faults, but not prediction.

About rerouting in GMPLS networks, John Park [3] considers a hybrid model for fault protection. In this approach, backups paths are selected when the main path is established, but the reservation of resources is only made when the faulty situation occurs. To deal with the occurrence of multiple faults a dynamic mechanism for fault recovery of the link is proposed in [3].

FIR (*Full Information Restoration*) is a rerouting algorithm for GMPLS networks introduced in [4]. The differential of the FIR is to share among different network nodes information about the available bandwidth for backup paths in order to optimize the use of such paths. Thus, for each link of the network, a master node keeps information on the bandwidth that needs to be replaced in case of link failure as well as information on which restoration link should be used if the faulty situation occurs. When a restoration link is selected, a message is sent from the source to the destination node through the restoration path with information about the main link in the main path. However, the proposed solution is valid only for single failures.

The main goal of [5] is to show an environment, called RENATA2, for generating ANNs that predicts failures and helps the GMPLS rerouting. For the rerouting, during the establishment of the main path, the election of a fixed alternative path should be made without reservation of resources.

When comparing these solutions, it is important to stress whether a prediction mechanism is used or not. Other features that should be taken in consideration are: recovery time, detection and recovery of multiple failures, consumed resources and algorithm complexity. Table 1 shows a summary of the related works mentioned in this section.

Solutions	**Prediction Mechanism**	**Recovery Time**	**Multiple Failure**	**Consumed Resources**	**Complexity**
RENATA2	Best Case	Imperceptible	Inefficient	Low	N^2 (prediction)
	Worst Case	Low			
Resilience in GMPLS	Not Present	Low	Eficient	Low	N^2 (rerouting)
FIR	Not Present	Low	Inefficient	Low	Not Mentioned

Table 1 - Comparison of Optical Rerouting Mechanisms.

In order to achieve more network nodes and to decentralize the rerouting decision, Intelligent agents can be combined with [5]. Such agents execute the task of monitoring an optical link of a network to predict anomalous situations, so that pro-active measures can be taken before faults occur. In this paper, we present a set of tools, which we detailed in the next section, to generate intelligent agents.

3 Intelligent Agent Generating Tools

Simulation tools like *ns* are available in the literature to help research in computer networks. However, there is a need for tools that help the simulation of a network faults and the generation of intelligent agents for optical networks. This paper presents then a set of tools, as follows. *ns* (*Network Simulator*) [6] is used to simulate IP, MPLS and GMPLS. JNNS (*Java Neural Network Simulator*) [7] is responsible for training the ANNs used in the intelligent agent. Besides *ns* and JNNS, five other tools are presented in this paper: GDP (*Disturbance Generator*), MSPD (*Data Preparation and Selection Module*), IT (*Training Interface*) and MGA (*Agent Generator Module*) [8].

3.1 Optical Link Simulation

A simulation of an optical system is necessary for getting *examples data base* that are used as knowledge database for the learning process of the neural networks. Such neural networks can be used as intelligent machine for software agents.

Since *ns* not only is one of the most used simulators in the literature but also is an open source software, we consider *ns* as part of the set of tools presented in this paper. However, *ns* does not provide a simulation of an optical physical layer that is necessary for generating the knowledge database mentioned. For that reason, GDP tool is proposed in this work for the simulation of faulty links, adding this functionality to *ns*. This tool simulates one link (transmitter, optic fiber and receiver) and the behavior of the physical layer. The optical link simulated by GDP works in the *full-duplex* mode. One side of the link has a semiconductor laser transmitter and on the other side there is a semiconductor photo receptor. The configuration parameters of these equipments had been chosen according to the DWDM system presented in [9].

On the other hand, *ns* [6] is responsible for the simulations of the upper layers: data link and network layers. The IP protocol is used in the network layer, and the MPLS is used between the data link and network layer. At the time of this paper writing, we could not find a *ns* plug-in that simulates a GMPLS network. Then, MNS (*MPLS Network Simulator*), which adds MPLS functionality to *ns*, is applied. The use of MNS does not affect the results for two reasons: the optical link is simulated by GDP, and GMPLS is a generalization of MPLS.

In the simulated network, two LSPs (*Label Switched Path*) are established: the LSP 1000 and the LSP 2000. The main LSP is the LSP 1000 where the traffic normally flows; the LSP 2000 is an alternative route to be used in case of a fault occurs. One LSR (*Label Switched Router*) is the *traffic protector*, which is defined in the MNS as responsible for monitoring the simulated link and warning about the need for rerouting. We call this link in the rest of the paper as the *main link*.

The main link supports 10 Gbps and the others links support up to 12 Gbps [10]. Node 0 is a IP router and also has the function of traffic generator, while node 7 is the final destination of the data generated by node 0. The traffic generated in the simulation is a 6MB burst of CBR UDP packages with 64KB each. The transmission assumes the following throughput values: 2, 4, 6, 8, 10 and 12 Gbps depending on the simulation.

3.2 ANN Training for Optical Intelligent Agents

The knowledge data base generated by the simulation is used in the training process for each type of neural network. Such data base is compose of three sets of *training patterns* (term used in the JNNS [7]) with each set stored in a separeted file (treina.pat, valida.pat and testa.pat) that has a defined purpose [5]. The MSE (*Mean Square Error*) was chosen to verify if MLP network is behaving correctly because is the most used [8].

MSPD not only is an interface between *ns* and IT but also does the analysis of the generated data by the ANNs. IT, which uses JNNS, and MGA are responsible for how the agent should act and interact with the network protocols when a faulty situation occurs.

The MSPD tool is responsible for adjusting the *log* files generated in the *ns* simulations. These files will compose the knowledge database for training and validating the neural networks. The Training Module, which uses the JNNS [7], is responsible for the training and validation of the neural networks developed in this paper.

To predict a faulty link, the neural network should consult the previous 20 collected samples and calculate a new BER (*Bit Error Rate*) value of the physical layer. In each simulation of the network a *log* is generated with several parameters: *link status, current traffic, incoming packets, lost packets* and the *BER*. The link status parameter represents whether the link is working or it is in a faulty situation.

For each 100ms a collection of the parameters in the simulation was made. The ANNs receives the parameters and suggests the need of rerouting in the next 100ms. In case it verifies the possibility of a faulty situation, the neural network still has 100ms to carry through the rerouting *before* the fault occurs. Accordingly to the simulations the average time necessary for the rerouting in the proposed topology is approximately 25.67ms.

Three types of MLP (*Multi-Layer Perceptron*) neural networks [11] had been developed in this work called ANN1, ANN2 and ANN3. AN1 uses parameters related with the physical equipment and predicts a faulty situation in the physical layer; ANN2 uses parameters related with the traffic of the optical link (except the link status); and ANN3 uses the same parameters as the ANN2, however it tries to get the ANN1 results. These three types of neural networks are candidates to act as inference machine of the intelligent agent. A 20-20-1 topology using *backpropagation* is appropriate for the three proposed neural networks as well as for training purposes.

3.3 Predicting Faulty Links

The parameters for ANN1 should be the values related with the physical layer, which is related only with GDP. *Ns* will generate logs to create the knowledge data base. The knowledge data base for ANN1 contains the last nineteen BER values and the current BER. This time window was empirically determined. The result of ANN1 is the BER value for the next 100ms. Thus, the BER values should be collected in the instants: *t-0,19, t-0,18, t-0,17..., t-0,3, t-0,2, t-0,1* and *t*. However, when using the neural network, it was obtained the value of the BER in the time *t+0,1*, where *t* indicates time in milliseconds. In the real world is not possible to directly measure the BER of a system in operation [12], but an alternative could be to amplify the optical signal ($\overline{p}_{rec}$) before the photo receptor receives the signal and calculate the BER [12].

ANN1 presents good behavior [8] in relation to the MSE validation considering all the points of the `valida` data base (0,0437). The ANN1 performance to the `treina` data base (0,2612) can be considered as reasonable behavior [8]. The complete set of values of the MSE can be seen in Table 2. In addition, the ANN1 achieved good results in the peak points, which are points that need to be predicted in the simulation because a gradual transition of status occurs from normal status to faulty situation of the simulated link.

Type of Error	**Results**		
Training MSE	0,2612		
Validation MSE	0,0437		
Peak Points MSE	Peak Points	δ	MSE
`Treina.res x Treina.pat`	118	$0,2 \geq \delta \geq 0,01$	0,0407
	171	$0,2 \geq \delta \geq 0,005$	0,1628
	229	$0,2 \geq \delta \geq 0,01$	0,0407
`Valida.res x Valida.pat`	16	$0,2 \geq \delta \geq 0,01$	0,0437
	25	$0,2 \geq \delta \geq 0,005$	0,1816
	37	$0,2 \geq \delta \geq 0,01$	0,3228
`Testa.res x Testa.pat`	20	$0,2 \geq \delta \geq 0,01$	0,0401
	28	$0,2 \geq \delta \geq 0,005$	0,1389
	35	$0,2 \geq \delta \geq 0,01$	0,2586

Table 2 - ANN1 Results.

The parameters of ANN2 are: link status, current traffic, received packages, lost packages and BER, which are collected in: *t-0,4, t-0,3, t-0,2, t-0,1* and *t*, where time *t* is given in milliseconds. The ANN2 indicates BER in the instant *t*. It presents excellent results for the BER inference of the physical layer in the present time. The MSE presents good results in relation to the total set of the training parameters and the SPPs (*Soft Peak Points*) [8].

The values of the two errors that have been calculate demonstrates the possibility to infer the physical layer BER of the behavior of the computer network traffic. However, this type of network did not get satisfactory results when it is used to predict failure.

This type of network presents good behavior [8] in relation to the MSE validation considering all the points of the `valida` data base (0,0105). The ANN2 performance in the `treina` data base (0,0150) can also be considered as good behavior [8]. Further informations and MSE values can be found in [8].

The MGA (Agent Generator Module) should automatically made available software agents in Java language so that they can be executed in different platforms including java enable routers. For this generation, the generator receives the topology, the weights and the *bias* of an ANN developed in the JNNS that will constitute the intelligent machine. Using this machine, an intelligent agent is generated to monitor equipment and/or traffic of the optical network capable of indicating a possible error, of interacting with the network protocols or executing some automatic corrective action.

4 Conclusions and Future Works

The most common way to verify if an ANN has reached its goals is observing the MSE. For fault prediction in an optical link, a good behavior of the MSE is related with all data in the knowledge database. The ANN should also present good results in relation to the SPPs [8], which are points of an eventual transition between normal and faulty stats of the link.

The set of tools presented in this article demonstrates that is possible to develop an intelligent agent that foresees or infers faulty situations in an optical network. Three types of ANNs (type 1, 2 and 3) has been developed, helped by this set of tools, as one solution to the IP+GMPLS on DWDM rerouting problem. It is possible to use information about the traffic (ANN2), as well as information related with the physical equipment (ANN1) to predict faulty links. However, ANN3 did not achieved good results.

Using our set of tools, the developer can quickly generate an intelligent agent. With the intelligent agent developed in this article it is possible to predict faulty situations in an optical link. One disadvantage is the lack of data of a real optical network to train the ANNs. Another possibility would be a hardware implementation of the desired ANNs in the equipment but changes on the ANNs are costly. As future work, a study of the consequences of a false prediction pointed by the agents should be made as well as the use of others type of neural networks like Self Organizing Maps and Fuzzy Logic should be investigated.

5 References

[1] Stamatios V. Kartalopoulos. *Fault detectability in DWDM : toward higher signal quality & system reliability*. New York: IEEE Press, 2001.

[2] MAS, CARMEN, THIRAN, PATRICK. An Efficient Algorithm for Locating Soft and Hard Failures in WDM Networks. *IEEE Journal*, 18(10):1900–1911, 2004.

[3] JOHN T. PARK. Resilience in GMPLS Path Management: Model and Mechanism. *IEEE Communications Magazine*, 42(7):128–135, July 2004.

[4] LI, G.; WANG, DONGMEI; K.C.; DOVERSPIKE, R. Efficient Distributed Restoration Path Selection for Shared Mesh Restoration. *IEEE Communications Magazine*, 11(5), 2003.

[5] Gonçalves, C.H.R.; Oliveira, M.; Andrade, R.M.C.; Castro,M.F.;. Applying Artificial Neural Networks for Fault Prediction in Optical Network Links. *Lecture Notes in Computer Science*, 3124:654–659, January 2004.

[6] PARC (Palo Alto Research Center), LBNL (Lawrence Berkeley National Laboratory) and UCB (University of California). ns2 (network simulator).

[7] JavaNNS: Java Neural Network Simulator. Available at `http://www-ra.informatik.unituebingen.de/software/JavaNNS/`, March 2005.

[8] GONÇALVES, C.H.R. Utilizando Redes Neurais Artificiais para Predição de Falhas em Links de Redes Ópticas. Master's thesis, (Mestrado em Ciência da Computação) - Departamento de Computação, Universidade Federal do Ceará, Fortaleza, 2003.

[9] Use of Altamar Optical Amplifiers with DWDM - APPLICATION NOTE. Available at `http://www.altamar.com/products/appnotes/DWDM.pdf`, December 2002.

[10] Scientific Atlanta Muliplexer/Demultiplexer for DWDM Systems - USER MANUAL. Available at `http://tvccatalog.tvcinc.com/Scientific\_Atlanta/IO106069dwdm.pdf`, December 2002.

[11] S. HAYKIN. *Redes Neurais Princípios e Prática*. 2.ed. Porto Alegre: Bookman. 900p., 2001.

[12] G. P. Agrawal. *Fiber-optic communication systems*. Fiber-optic communication systems, 2nd ed., by Govind P. Agrawal. New York, NY: Wiley, 1997, "A Wiley-Interscience publication", ISBN 0471175404., 1997.