

# Self-Configuration and Self-Healing for Cognitive Optical Networks

Tania Regina Tronco<sup>1,2</sup>

<sup>1</sup>Fundação CPqD, Diretoria de Gestão da Inovação, Campinas-SP, Brasil, [tania@cpqd.com.br](mailto:tania@cpqd.com.br)

Mariana Massimino Feres, Amilcar Careli César, Mônica de Lacerda Rocha<sup>2</sup>

<sup>2</sup>Escola de Engenharia de São Carlos – USP, São Carlos-SP, Brasil, [monica.rocha@usp.br](mailto:monica.rocha@usp.br)

**Abstract**— In this article we propose a fuzzy controller, as an inference engine for cognitive optical networks, to take decisions about routing of new demands of lightpaths, considering physical layer impairments (Fuzzy Controlled-PLIARWA algorithm), self-configuration, self-healing and cross-layer optimization functionalities. The proposed algorithm has been tested in a metropolitan-scaled network. The preliminary results obtained are promising in terms of modularity, flexibility, and high processing speed, independency of underlying technology and scalability of the solution.

**Index Terms**— Cognitive Optical Networks, Adaptive Optical Networks, Self-Configuration, Self-Healing, GMPLS, WDM.

## I. INTRODUCTION

Optical networks have evolved continuously, increasing their capacity due to the growing of Internet traffic demand, thus setting the path for new approaches that allow them to become more dynamic and flexible. Such reconfigurable optical networks have the capability of being dynamically programmable by software [1] thus enabling, among other features: (1) configuration of different bit rates in the same optical transponder, (2) reconfiguration of the optical equipment parameters, depending on the quality of the transmission, (3) selection of different modulation formats and FEC (Forward Error Correction) algorithms and (4) dynamic adjustment of the optical channel bandwidth.

In this new scenario, the network control and management systems become more complex and may require the use of artificial intelligence (AI) techniques such as genetic algorithms, fuzzy logic, neural networks, ant colony optimization, etc., in order to adapt the network configuration for meeting the service requirements under variable conditions of the optical layer. By using cognition, the optical network can observe its own physical layer and also the MAC (Media Access Control) and network layers. This way, based on the collected information, the control plan can act, learn and optimize the network performance. Such architecture represents a new paradigm in terms of networking evolution, resulting in the so-called “Cognitive Optical Networks”. Fig. 1 illustrates the model of a cognitive network proposed by Thomas et al. [2], composed of three layers: (i) Reconfigurable network elements (“Software Adaptable Network”), (ii) Cognitive Process and (iii) End-to-End goals.

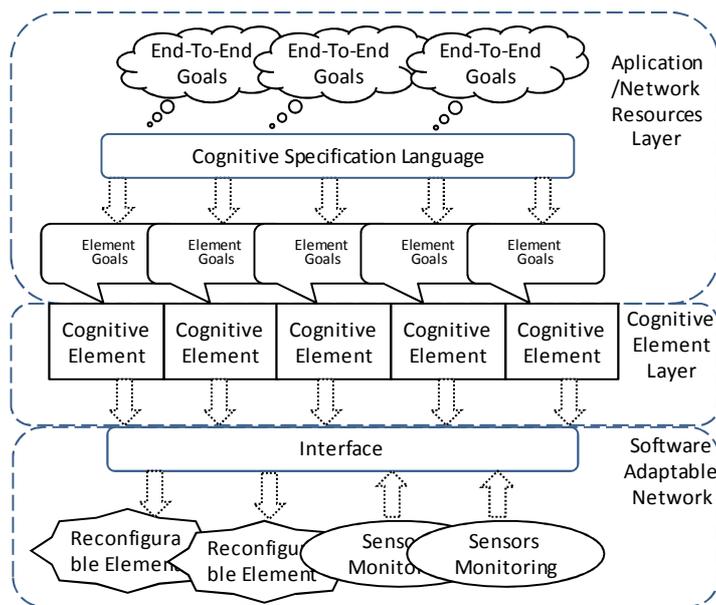


Fig. 1 - Cognitive Network Model, based on [2]

In Fig. 1, the software adaptable network consists, for example, of ROADMs (Reconfigurable Optical Add/Drop Multiplexers), tunable transponders, optical amplifiers, etc. The cognitive layer corresponds to an extension of the control plane employing artificial intelligence. Furthermore, spread throughout the network, real-time sensors monitor the physical layer. The end-to-end goals are related to high level targets, designed by network operators.

Other properties of those cognitive optical networks are self-configuration, self-optimization, self-healing and self-protection. Self-configuration means that the system automatically detects and configures new equipments, components and network connections, promoting the necessary adjustments to their incorporation into the network. The self-healing property means that the system detects fails and repairs problems that occur during the network operation, keeping it normal. In the self-optimization process, the system continuously monitors the network and performs rearrangements for resource optimization and performance. Finally, in the self-protection, the system quickly identifies attacks and vulnerabilities, running appropriate protective actions. These properties allow naming those networks as “autonomic”. The term “autonomic” comes from an analogy with the human autonomic nervous system and idealizes a telecommunication network in which managers are concerned with the high-level decisions that correspond to business rules, while the management and control systems meet these rules taking their own decisions, for example, self-configuring equipments, self-optimizing its parameters, self-recovering from failures and self-protecting against attacks or errors [3]. We consider the autonomic functionalities (self-\*) as part of the cognition concept, since they are essential to provide scalability to the architecture of the cognitive network.

Based on these characteristics, we propose a definition to cognitive optical network as: a network that should have the following capabilities, as essential:

1. Environment monitoring and context inference;
2. Learning, applied to all the network layers, retaining knowledge from interactions with the environment, storing this knowledge in a "knowledge base" and adopting different decisions/strategies at different times;
3. Decision, operation and planning: to make decisions, act on the system and plan future actions, based on:
  - Data collected from the environment;
  - Services requirements;
  - Profiles of network customers;
  - Policies provided by network operators;
  - Knowledge stored in the knowledge base.
4. Auto-adaptation to the environment changes in a reactive and proactive mode;
5. Self-configuration;
6. Self-healing;
7. Self-optimization;
8. Self-Protection;
9. Cooperation between network elements and between different network domains and
10. Interaction between different network layers (also named cross-layer design).

Currently, there are some cognitive optical network architectures in development, which we highlight, the following:

- CHRON project [4], an European Community (EC) project with funding from EC's Seventh Framework Programme (FP7/2007-2013) and
- COGNITION [5], a research line within High Performance Network Group in Department of Electrical and Electronic Engineering, University of Bristol, UK.

Nevertheless, each one of these architectures has its own architecture design, employing different cognitive systems with different goals to achieve, lacking a global standard for a unified architecture. The search for a suitable architecture for cognitive optical networks is characterized by challenges such as modularity, cross-layer design, scalability, information imprecision, selection of learning techniques, etc.

In such scenario, to overcome the challenges of cognitive optical network architecture, we propose the development in two phases:

Phase 1: Built an inference engine (Fuzzy Controller) using classic fuzzy logic to take decisions about new demands of connections (lightpaths), self-configuration, self-healing and cross-layer optimization.

Phase 2: Add learning capability to this engine, through Fuzzy Cognitive Map (FCM)

scheme. FCM is a tool for modeling human knowledge using the fuzzy systems, but with a structure similar to Artificial Neural Networks (ANN), which facilitates the processing of data and has training capacity and adaptation [6]. Fuzzy Cognitive Maps can be used to dynamically change the fuzzy logic modeling already defined in phase 1, according to learning and experience gained from the environment.

The use of fuzzy logic was inspired by the work of Baldo and Zorzi [7], in which fuzzy logic is employed to represent the cross-layer information and optimization in the network layers of cognitive radio networks. Baldo and Zorzi have shown the benefits in terms of modularity, low complexity, high processing speed, independent of underlying technology and scalability of using fuzzy logic in cognitive radios.

In this paper, we present the phase 1 of this development, the inference engine, called Fuzzy Controller, used for the representation of information about the physical, MAC and IP layers and also to take decisions about new demands of connections, through a PLIA-RWA (Physical Layer Impairments Aware – Routing and Wavelength Assignment) algorithm, self-configuration and self-healing of optical network resources and cross-layer optimization. This controller includes interactions between IP, MAC and physical (optical) layers (cross-layer design). In IP layer, the network congestion state is verified by an artificial neural network. In the MAC layer, the OTN (Optical Transport Network) technology parameters are considered and, in the optical physical layer, the bit rate requested by the clients and PLIs (Physical Layer Impairments) are taken into account.

The paper is organized as follows. In section II, the Fuzzy Controller (FC), its operation mode and benefits are presented. We also describe the process of translating all these information into a fuzzy representation, together with an appropriate fuzzy controller design. Next, in section III, we presented a FC-PLIARWA simulation and its results applied to a metropolitan network using the softwares Matlab and OptiSystem 9.0 [8]. Finally, in section IV the conclusions are highlighted.

## II. FUZZY CONTROLLER ARCHITECTURE

Fuzzy set theory differs from traditional set theory, where an element either belongs to a set or it does not. In fuzzy, a partial membership is allowed (i.e., an element can belong to a set only to a certain degree). This membership degree is commonly referred to as the membership value and is represented by a real value in the interval  $[0, 1]$ , where 0 and 1 correspond to full non-membership and membership, respectively. Commonly, triangular or trapezoidal functions are used as membership functions because of their simplicity [9].

A fuzzy controller system uses fuzzy logic reasoning to take decisions. Since the system inputs and outputs system are commonly crisp in nature, a fuzzification and defuzzification processes are needed in order to translate them to and from fuzzy representation, as depicted in Fig. 2. The term crisp is used to indicate variables having exact values, as opposed to the term fuzzy, which indicates a qualitative rather than quantitative method of representation [9].

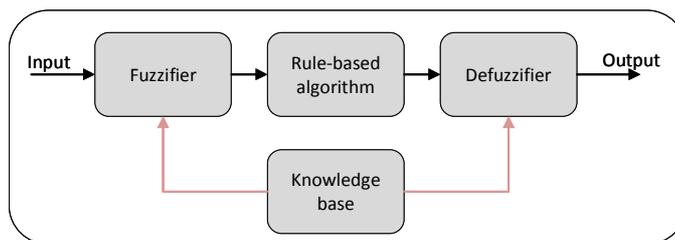


Fig. 2 – Basic Architecture of a Fuzzy Controller

In Fig. 2, the knowledge base defines the relationship between crisp input/output parameters and their fuzzy representation, using membership functions. From a practical point of view, each input/output variable is characterized by a set of linguistic attributes (“labels”) and, for each label, a membership function is defined. As an example, if SNR (Signal to Noise Ratio), in dB, is an input variable of the control system, in fuzzy logic, we can choose the attributes such as good or bad to represent SNR qualitatively and define a membership function of each one of these attributes.

The Rule-Based Algorithm is the heart of the fuzzy logic controller and is composed of a set of IF-THEN rules used to determine the value of the output variables.

The Fuzzy Controller we propose is shown in Fig. 3. It is implemented as a centralized engine that receives information from each network layer, of each network element, and exports parameters by all layers, of all network elements. Moreover, it interacts with the GMPLS (Generalized MultiProtocol Label Switching) control plane, distributed in each network element, taking decisions about routing and wavelength assignment.

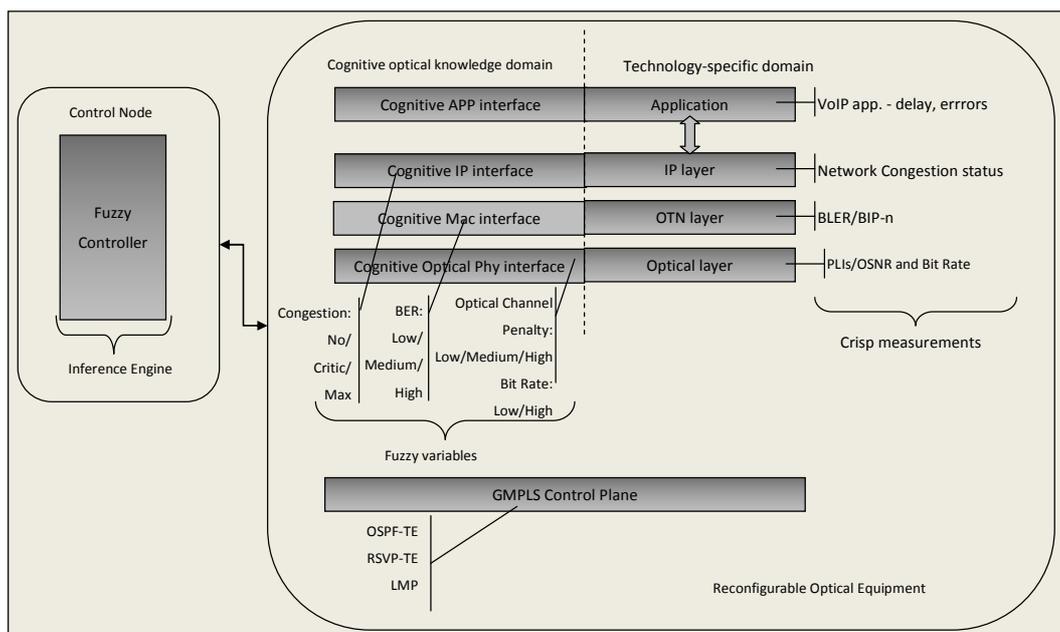


Fig. 3 – Fuzzy Controller for Cognitive Optical Architecture

GMPLS is an IETF standard [10] composed of three main protocols: a routing protocol with traffic engineering, in general OSPF-TE (Open Shortest Path First - Traffic Engineering), a signaling protocol, called RSVP-TE (Reservation Protocol - Traffic Engineering), and a protocol for managing the connections, called LMP (Link Management Protocol).

Since, in our proposal, the routing decisions are taken by the fuzzy controller, the OSPF-TE can be used only as a mean to share the availability of the nodes and links and the configuration parameters among the network elements, as proposed by [11].

After calculating the route by fuzzy controller, the signaling protocol RSVP-TE provides the path on all nodes belonging to the route. The selection of a route in the optical network involves the selection of nodes and links and also the wavelength to be used for transmission.

According to Baldo and Zorzi [7], the first step to build a Fuzzy Controller is to define, for each layer in the protocol stack, a set of variables and parameters and translate them to fuzzy variables using generic status, independent of the underlying technology.

In our approach, the physical layer exploits the knowledge of which bit rates are requested by the clients (i.e. 10 Gbit/s or 40 Gbit/s) and which is the PLIs value, related to the each lightpath. The bit rates are translated into a more qualitative fuzzy attributes: low (10 Gbit/s) and high (40 Gbit/s) and the PLIs values and its associated system penalty, measured in dB, are also translated into normalized fuzzy attributes: low ( $0\text{dB} \leq \text{penalty} \leq 1\text{dB}$ ), medium ( $1\text{dB} \leq \text{penalty} \leq 4\text{dB}$ ) or high ( $\text{penalty} \geq 4\text{dB}$ ). We consider PLIs linear (Chromatic Dispersion (CD), Polarization Mode Dispersion (PMD), Polarization Dependent Loss (PDL) and Crosstalk (XT)) and nonlinear (Self phase Modulation (SPM) and Cross Phase Modulation (XPM)).

In the MAC layer, we consider the counting of blocks error (Block Error Rate - BLER) in the OTN frames. The OTN frame includes a code, named BIP-n (Bit Interleaved Parity – level n) in the overhead field, which is used to block error rate supervision. This code is composed by n independent parity counting in n adjacent positions of each byte. The counting result is stored in an overhead byte in the OTN frame and sent in the transmission side. At the reception side, the counting is made again and compared to the received. In case of difference between the two values, a block errored is counted. The value of the variable n of the BIP-n depends of which OTN layer it refers. From BLER (Block Error Rate) counted by BIP-n, at each OTN layer, it is possible to infer the BER (Bit Error Rate), as demonstrated in [12]. Then, the BER value is translated into a fuzzy variable as: low ( $\text{BER} \leq 10^{-10}$ ), medium ( $10^{-10} < \text{BER} \leq 10^{-6}$ ) or high ( $\text{BER} > 10^{-6}$ ).

At IP layer, an artificial neural network type "Multilayer Perceptron with backpropagation algorithm" (MLP) [13] is employed to recognize IP network congestion patterns. The backpropagation algorithm is used to minimize the error of the system. The traffic conditions are evaluated during a VoIP application (Voice over IP - VoIP) over an established lightpath. The neural network configuration employed is shown in Fig. 4. It is composed by an input neural layer, a neural output layer and intermediate layer (hidden) neurons, located between the input layer and the output

layer. The neural input layer has six inputs (X1, X2, .... X6), each of which receives an input of the IP network traffic parameters such as: inter-packet delay, lost packet rate, number of lost burst, length of lost burst, jitter and probability of lost. The neural output layer has 4 outputs (S1, S2, S3 and S4) to indicate the traffic conditions in the network, as shown in Table 1.

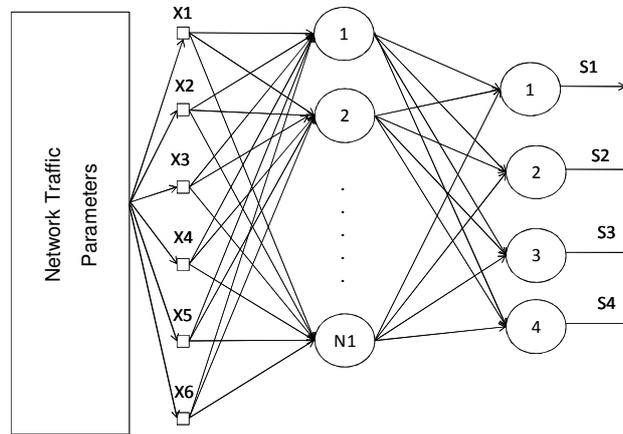


Fig. 4 – Basic Architecture of Neural Network

TABLE I. OUTPUT OF NEURAL NETWORK AND TRAFFIC CONDITIONS

S1	S2	S3	S4	Condition
0	0	0	0	No congestion
.	.	.	.	
0	1	1	1	Critical congestion
.	.	.	.	
1	1	1	1	Max congestion

The total set of network parameters samples were divided into two sets: a set of training patterns and a set of test patterns and three topologies were tested with different numbers of neurons at the intermediate layer: 3, 10 and 15. The network topology including 15 neurons at intermediate layer showed the best results for the probability of correct classification (75%). When more than 15 neurons are employed at intermediate layer, the probability remains constant (75%). Fig. 5 shows the mean squared error as a function of the number of epochs during the training phase.

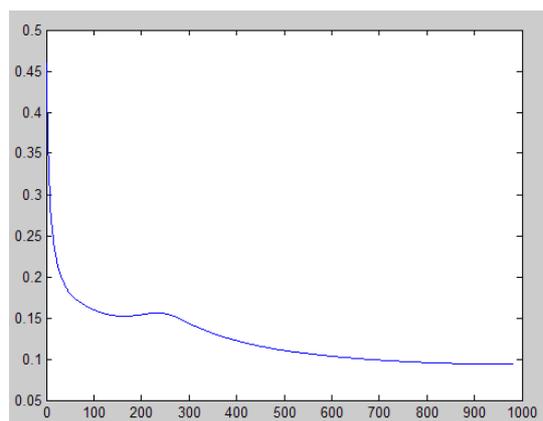


Fig. 5 – Mean Squared Error versus Number of Epochs

The neural network outputs are translated in qualitative fuzzy attributes such as: [0 to 10] – low network congestion and [11 to 16] – high network congestion.

#### FUZZY CONTROLLER - OPERATION MODE

The Fuzzy Controller can provide the following four main functionalities, and beyond:

1. FC-PLIA-RWA algorithm: it decides if a new lighthpath can be established, based on the current status of the network in terms of quality of transmission (QoT) and in the bit rate requested by the client.  
The fuzzy logic provides flexibility to the PLIA-RWA algorithm in accepting more or less connections, when compared with PLIA-RWA algorithms using crisp thresholds (non-fuzzy). Using FC-PLIA-RWA, the telecom operators can choice and adapt the fuzzy functions values dynamically, according to their policies.
2. Self-Configuration: it decides if an established lighthpath should be reconfigured, changing modulation formats, increasing or decreasing link capacities, connecting or disconnecting nodes at the given topology, based on low QoT, congestion scenarios, new user's demands for more bandwidth or policies of the operators.
3. Self-healing: it decides if an established lighthpath should be switched to a backup lighthpath, in case of link/node failures or low QoT (Quality of Transmission).
4. Cross-layer Optimization among the network layers: it provides joint optimization of all network layers.

The block diagram of Fig. 6 shows the operation of the FC-PLIARWA algorithm. Initially, the network topology is collected and, if a new connection is requested, a route is calculated between the source and destination nodes using the Dijkstra (OSPF) algorithm. Then, based on the PLI and penalty correspondent stored in the knowledge database, and in the bit rate requested, the system decides if the route should be accepted or not. In affirmative case, a set of wavelengths is assigned to it. In negative case, the connection is blocked.

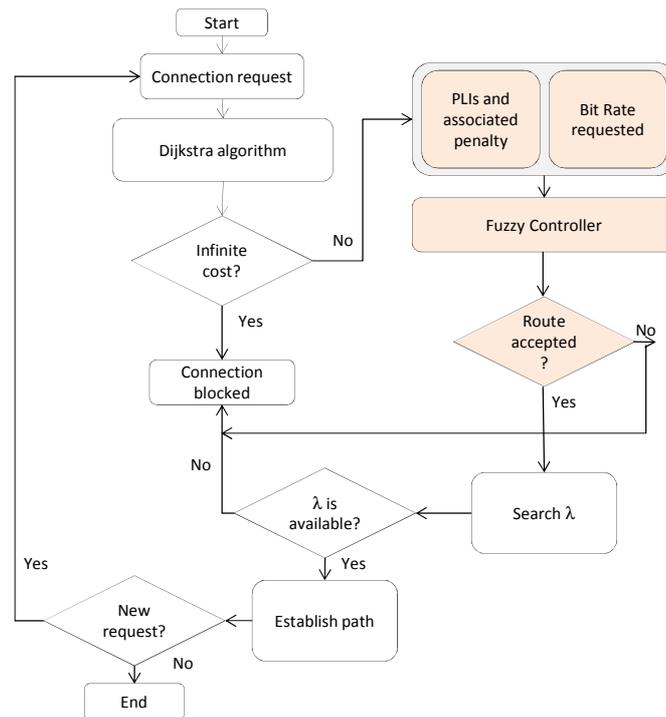


Fig. 6 – FC- PLIARWA Operation Mode

The FC-PLIARWA rules are shown in Table II.

TABELA II. FC-IARWA RULES

Bit Rate and Penalty		R
Bit Rate	Penalty	
Low	Low	1
Low	Medium	1
Low	High	0
High	Low	1
High	Medium	0
High	High	0

In Table II, the variable R shows when the connection request can be accepted (R=1) or not (R=0), depending on the bit rate and penalty values.

The FC-PLIARWA Penalty fuzzy membership function is shown in Fig. 7.

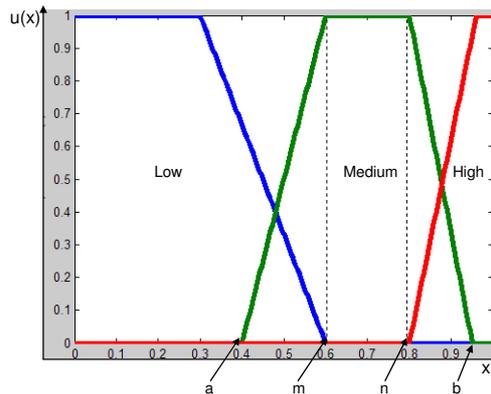


Fig. 7 – Penalty Membership Function

For self-configuration, self-healing and optimization, we use a similar approach. For example, we associate the increment or decrement of link capacities (bit rate) to the fuzzy variables LCIncr and LCDecr, respectively. The value of these variables is determined as the output of the fuzzy controller using the following rule set:

**Scenario 1: QoT:**

IF (penalty is high) AND (bit rate is high)  
 THEN (LCDecr=1 and LCIncr=0)

**Scenario 2: Users demand more bit rate:**

IF (penalty is low) AND (bit rate is low) AND (network congestion is low) AND (user request more bandwidth)  
 THEN (LCDecr=0 and LCIncr=1)

**Scenario 3: High Congestion:**

IF (R=0) AND (penalty is low)  
 THEN (LCDecr=1 and LCIncr=0)

where R=0 means connection request rejected.

In self-healing, we associate the decision to switch or not for a backup lightpath to the fuzzy variables SwBack and NSwBack, respectively. The value of these variables depends of the each scenario: (1) link/node failures or (2) low QoT (Quality of Transmission) and is determined as the output of the fuzzy controller using the following rule set:

**(1) Link/Node Failure:**

In this case, the approach is:

1. Simulate a link or node failure and search for an alternative lightpath;
2. Evaluate wavelength and path availability;
3. Evaluate physical layer impairments (penalty);
4. If penalty is low then => switch for the alternative lightpath.

The rules are:

IF (link failure=1) AND (penalty is low)  
THEN (SwBack=1 and NSwBack=0)

(2) Low QoT:

In this case:

1. Evaluate PLIs at optical layer, BER, at OTN layer, and network congestion, at IP layer;
2. If penalty is high but BER or network congestion is high then => non switch for the backup lightpath.
3. If penalty is high but BER and network congestion is low, then => switch for the backup lightpath.

The rules are:

IF (penalty is high) AND (BER is high) OR (network Congestion is high)  
THEN (SwBack=0 and NSwBack=1)

IF (penalty is high) AND (BER is low) AND (network congestion is low)  
THEN (SwBack=0 and NSwBack=1)

In cross-layer optimization, we can, for example, activate an enhanced FEC (EnFEC) at OTN layer if the penalty at the optical layer is high.

Then, the rule is:

IF (penalty is high) AND (BER is high)  
THEN (EnFEC=1)

In this system, we can continually add new parameters and rules easily and the algorithm consumes low time of processing.

### BENEFITS

The fuzzy controller presented here is modular, scalable and allows independent implementation of the layers themselves, while representing cross-layer information using generic status variables ('good, bad, low, medium'). Fuzzy logic is intrinsically suited to represent the likelihood of information, is independent of the underlying technology and it requires low computational resources, allowing fast and consistent decisions.

### III. SIMULATIONS AND RESULTS

We simulated a metropolitan network and the FC-PLIARWA algorithm using Matlab and OptiSystem 9.0 tools. The network topology is illustrated in Fig. 8 (a). It comprises networks with 8 nodes in a mesh topology. Each node includes a reconfigurable add/drop multiplexer (ROADM) with 8 wavelengths spaced by 100 GHz. Each link comprises a fiber length of at least 45 km, one dispersion compensating fiber (DCF) module centered at 1550 nm and a preamplifier. The pattern of connections is illustrated in Fig. 8 (b).

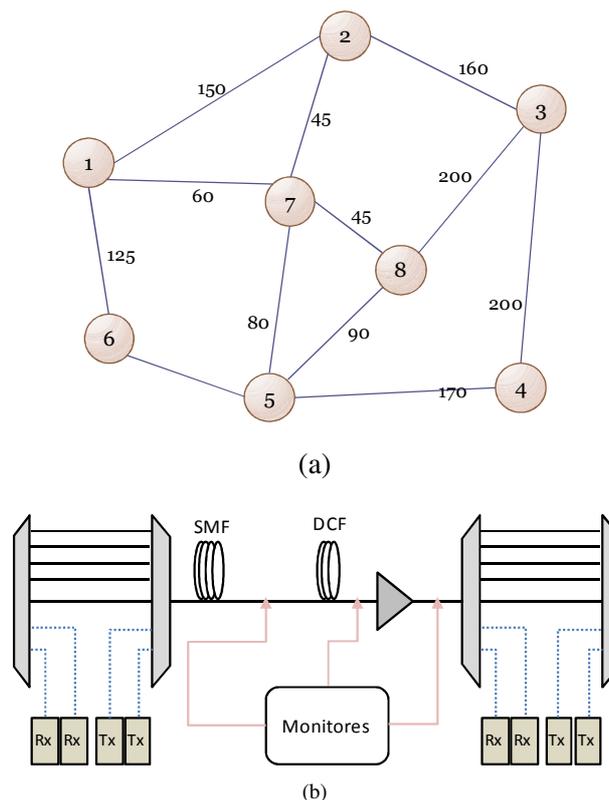


Fig. 8 – Network Topology and Link Model

A knowledge database stores the PLIs values resultant from simulations of the links when passing through the ROADMs.

In order to test the FC-PLIRWA, implemented in Matlab, a traffic generator was used to create 20000 connections with 1000 Erlangs each one, and the bandwidth requested by the user is allocated and released, according to this generator. As a connection is accepted, the traffic generator decrements the bandwidth requested of the link bandwidth. When the total bandwidth of the link is exceeded or when no longer a wavelength available, the connection is blocked.

The blocking probability of connections is defined by:

$$\text{Blocking Probability} = \frac{(\text{Number of accepted connections})}{(\text{Total number of connection requests})} \tag{1}$$

Aiming to prove the flexibility of the FC-IRWA, two simulations were executed with different thresholds for the fuzzy membership function of the penalty (Fig. 7). In first one, the threshold values used were: for low penalty: a=0.0, m=0.0, n=0.2 and b=0.4; for medium penalty: a=0.2, m=0.4, n=0.6 and b=0.8 and for high penalty: a=0.6, m=0.8, n=1.0 and b=1.0. In this case, the blocking probability obtained was 0.21. In second case, the threshold values were: for low penalty: a=0.0, m=0.0, n=0.5 and b=0.6, for medium penalty: a=0.2, m=0.4, n=0.7 and b=0.8 and for high penalty: a=0.5, m=0.8, n=1.0 and b=1.0. In this case, the blocking probability obtained was 0.11. The results confirmed that using fuzzy logic a network operator can adjust the blocking probability according to their policies,

just adjusting the fuzzy membership functions.

We also tested the fuzzy controller functionality of decreasing the bit rate in high congestion state (high blocking probability). The results showed a lower blocking probability when the bit rate was decreased, compared to that one without this process. The blocking probability decreased from 0.21 to 0.03.

Preliminary results demonstrated the flexibility and low complexity of the fuzzy system as foreseen. As future works, new parameters and rules can be easily added to the system, proofing its scalability.

#### IV. CONCLUSIONS

This paper presented a proposal of a fuzzy controller to take decisions about PLIA-RWA, self-configuration, self-healing and cross-layer optimization. Preliminary results showed the feasibility of this solution due to its flexibility, scalability and independency of underlying technologies. The next steps of this work consist in add other parameters, functionalities and learning capability to the system, aiming to build a complete architecture for cognitive optical networks. Further simulations are also being carried for comparison between control planes with and without the fuzzy system proposed and also between fuzzy system and fuzzy cognitive map.

#### REFERENCES

- [1] A. Wang, j. Yu, "Cognitive Optical Networks: Key Drivers, Enabling Techniques and Adaptive Bandwidth Services", IEEE Communications magazine, janeiro de 2012.
- [2] R. Thomas, "Cognitive Networks" - Dissertation submitted to the Faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Computer Engineering, June 15, 2009.
- [3] Z. Movahedi et al., "A Survey of Autonomic Network Architectures and Evaluation Criteria", IEEE Communications Surveys and Tutorials, Vol. 14, No. 2, Second Quarter 2012.
- [4] CHRON Project, <http://www.ict-chron.eu/>.
- [5] G. Zervas, D. Simeonidou, "Cognitive Optical Networks: Need, Requirements and Architecture", Proceedings ICTON 2010.
- [6] M. Glykas, "Fuzzy Cognitive Maps - Advances in Theory, Methodologies, Tools and Applications", ISBN 978-3-642-03219-6, Springer-Verlag Berlin Heidelberg, 2010.
- [7] N. Baldo and M. Zorzi, "Fuzzy Logic for Cross-Layer Optimization in Cognitive Radio networks", IEEE Communications magazine, April 2008.
- [8] [http://www.optiwave.com/products/system\\_overview.html](http://www.optiwave.com/products/system_overview.html), accessed in March, 21, 2013.
- [9] G. Klir and B. Yuan, "Fuzzy Sets and Fuzzy Logic: Theory and Applications, Prentice-Hall PTR, 1995.
- [10] IETF RFC 3945 – "Generalized Multi-Protocol Label Switching (GMPLS) Architecture", October 2004.
- [11] D. Siracusa et al., "A control plane framework for future cognitive heterogeneous optical networks." International Conference on Transparent Optical Networks (ICTON), 2012.
- [12] T. Tronco, "Avaliação da Precisão de BER Obtida Através dos Códigos Corretores de Erro de Sistemas SDH Ópticos e Rádio", VI Simpósio Brasileiro de Microondas e Optoeletrônica, Belém, PA, Brasil, 1994. Vol. 1. p. 295-300.
- [13] Silva, I. N. et al., "Redes Neurais Artificiais para Engenharia e Ciências Aplicadas". Editora Artliber, 2010.