Intelligent and Fast IRWA Algorithm based on Power Series and Particle Swarm Optimization

Joaquim F. Martins-Filho¹, Daniel A. R. Chaves¹, Carmelo J. A. Bastos-Filho² and Douglas O. Aguiar¹,
¹ - Department of Electronics and Systems, Federal University of Pernambuco, Recife-PE, 50740-530, Brazil
² – Department of Computing Systems, University of Pernambuco, Recife-PE, 52720-001, Brazil
Tel/Fax: +55 81 21268995, e-mail: jfmf@ufpe.br

ABSTRACT
In all-optical networks signals are transmitted through physical layer with no regeneration. Therefore, physical impairments along lightpath can severely reduce network performance. For this reason, many efforts have been made to develop impairment aware routing and wavelength assignment algorithms (IRWA) in order to mitigate the impairments effects, improving the network performance. In this paper we propose and analyze the performance of an adaptive impairment aware routing algorithm based on a set of chosen input network parameters. The cost function of this routing algorithm is based on a power series expansion. The routing algorithm, called Power Series Routing (PSR), is trained by an optimization technique called Particle Swarm Optimization. We show that this IRWA algorithm can learn during the training stage and adapt itself to the network conditions.

Keywords: All-Optical Networks, Noise, Optical Signal-to-Noise Ratio, Routing and Wavelength Assignment.

1. INTRODUCTION
In transparent all-optical networks there is no signal regeneration at intermediate nodes along the lightpaths. Therefore, the signals accumulate noise due to transmission impairments. For this reason, the routing algorithm should be aware of these physical penalties to fetch routes that minimize OSNR degradation due to optical noise. Recently, many efforts have been made to develop RWA algorithms that consider physical impairments (IRWA) [1]-[4]. The main goal of these approaches is to minimize the blocking probability by finding routes considering physical layer status. Although routing schemes based on optical impairments outperform the most common approaches, the use of these algorithms implies in higher computational complexity and longer computational time to obtain the route. Some algorithms have been proposed to achieve a good performance with a lower computational time [3].

In this paper we propose a method to build the link cost function based on a set of relevant network parameters. This is an important tool for routing algorithm design, since the parameters selection is a relatively easy task for a network specialist. Nevertheless, combining these parameters to obtain optimal network performance is a complex task. For this reason we propose an adaptive cost function for impairment aware routing, which we call PSR (Power Series Routing). We use PSR to provide the link cost for a lowest cost routing algorithm (e.g. Dijkstra’s algorithm). The PSR training is performed by the particle swarm optimization technique.

2. POWER SERIES AND ALGORITHM DESCRIPTION
In this section we present a method to determine a link cost function for network routing. The proposed approach consists of three steps: first, a number of input variables for the cost function are chosen by a network specialist. Then, the cost function is written in terms of a series of functions. And finally, an optimization algorithm is used to find the series coefficients that minimize the network blocking probability.

In this paper, we focus our analysis in the series that make use of a set of orthogonal polynomials. Assuming the continuity of the function and its derivatives, the expansion can also be done for a multivariable function. It is well known that one can find the coefficients of the expansion by means of derivatives (multivariable Taylor’s series). However, this approach works only for a function with derivatives. Nevertheless, one can find the coefficients of the expansion by a non analytical procedure. Considering this, we used the proposed approach to build an adaptive cost function for impairment aware routing, which we call power series routing (PSR).

The first step is to choose the input variables for the cost function. In optical networks the information about link length, link availability and number of hops have high correlation with noise accumulated along the lightpath. Excessive noise can damage the signal transmission quality as the link length increases, and higher gains must be provided by the optical amplifiers to compensate for the losses. Therefore, more ASE noise is added by optical amplifiers in the lightpath. Link usage has impact in amplifier saturation and ASE noise generation, since the amplifier gain and noise figure depends on the total input signal power [5]. Besides, as the number of hops increases, more crosstalk noise is added in intermediate nodes. Therefore, these elementary network parameters could be used to build a simple routing scheme, instead of using the noise information, obtaining similar network performance as for schemes that use optical noise information to compose the cost function [3]. For these reasons, we choose as input variables for the cost function two simple network parameters: normalized link availability and normalized route length.
The second step is to describe the cost function in terms of a series, according to the number of network parameters chosen. Therefore, the link cost between nodes \( i \) and \( j \) can be expressed in a two variables form by:

\[
 f(x_{i,j}, y_{i,j}) = \sum_{n_0=0}^{\infty} \sum_{n_1=0}^{\infty} b_{n_0,n_1} x_{i,j}^{n_0} y_{i,j}^{n_1},
\]

where \( x_{i,j} \) and \( y_{i,j} \) are, respectively, the link availability and normalized link length between the nodes \( i \) and \( j \). \( x_{i,j} \) is defined as:

\[
 x_{i,j} = \frac{\lambda_{i,j}^a}{\lambda_{i,j}^T},
\]

where \( \lambda_{i,j}^a \) and \( \lambda_{i,j}^T \) are, respectively, the number of unused and total number of wavelengths in the link between nodes \( i \) and \( j \). The normalized link length \( y_{i,j} \) is defined as:

\[
 y_{i,j} = \frac{d_{i,j}}{d_{\text{max}}},
\]

where \( d_{i,j} \) is link length between nodes \( i \) and \( j \) and \( d_{\text{max}} \) is the maximum link length in the network. Since it is not possible to have an infinite number of terms in Equation (1), one shall truncate the series in order to obtain an approximation with \( N \) terms:

\[
 f(x_{i,j}, y_{i,j}) = \sum_{n_0=0}^{N} \sum_{n_1=0}^{N} b_{n_0,n_1} x_{i,j}^{n_0} y_{i,j}^{n_1},
\]

One can note from Equation (4) that this function has a constant term, which can represent the hop cost.

The third step consists of using PSO to find the series coefficients that optimizes a network performance parameter. We used PSO because it achieves a better performance in high dimensionality problems than other optimization techniques (e.g. Genetic Algorithms). The optimization algorithm can be either used to maximize the network throughput or minimize network blocking probability. In this paper we find the \( b_{n_0,n_1,...,n_k} \) coefficients that minimize blocking probability as will be described in the next section.

It must be highlighted that one can include an arbitrary number of input parameters in order to build the cost function, including direct information about the physical impairments.

3. PARTICLE SWARM OPTIMIZATION

For the particle swarm optimization we used the algorithm recommended in [6], following the pseudo-code proposed therein. The particle velocities are updated using the constriction factor approach, the \( L_{\text{best}} \) swarm model was used as the swarm communication topology [6].

4. SIMULATION SETUP

Our simulation software uses the following steps. Upon a call request it selects an available wavelength from a list, using the first fit algorithm. The route is defined by a routing algorithm that uses one of the following weight functions: Shortest Path algorithm (SP), with physical length as the cost function; Least Resistance Weight (LRW) described in [7]; an algorithm that uses the total OSNR of the lightpath as the cost function (OSNR-R) proposed in [1]; and our proposal. The OSNR of each lightpath is evaluated using the same model used in [5].

This model considers the following impairments: ASE noise, amplifier gain saturation effect, saturation of ASE noise in EDFAs and homodyne crosstalk in optical switches. If the OSNR of the lightpath is above the pre-determined level \( (\text{OSNR}_QoS) \), then the call is established. Otherwise the call is blocked. Our algorithm also blocks a call if there is no wavelength available. The blocked calls are lost. For each network simulation a set of \( 10^7 \) calls are generated by choosing randomly (uniform distribution) the source-destination pair. The call request is characterized as a Poisson process. We assume circuit switched bidirectional connections in two different fibers and no wavelength conversion capabilities.

The default optical parameters used in our simulations are: amplifier output saturation power \( P_{\text{Sat}} = 19 \text{ dBm} \), transmitter output power \( P_{\text{t}} = 0 \text{ dBm} \), input optical signal-to-noise ratio \( \text{OSNR}_{\text{in}} = 40 \text{ dB} \), optical signal-to-noise ratio for QoS criterion \( \text{OSNR}_{\text{QoS}} = 23 \text{ dB} \), optical filter bandwidth \( B_{\text{c}} = 100 \text{ GHz} \), channel spacing \( \Delta f = 100 \text{ GHz} \), fiber loss coefficient \( a = 0.2 \text{ dB/km} \), multiplexer loss \( L_{\text{MUX}} = 3 \text{ dB} \), demultiplexer loss \( L_{\text{DEMUX}} = 3 \text{ dB} \), switch loss \( L_{\text{Switch}} = 3 \text{ dB} \), amplifier noise factor that corresponds to \( \text{NF} = 5 \text{ dB} F_0 = 3.162 \), noise factor model parameter \( A_1 = 100 \text{ noise factor model parameter } A_2 = 4W \text{ [5]} \), switch isolation factor \( \epsilon = -41 \text{ dB} \). Amplifier gains are set to compensate link losses. We used the well known network topology of Finland that is often used as a benchmark.

We used the following PSO parameters in our simulations: 50 particles, 500 interactions, velocity update parameters \( c_1 = c_2 = 2.05, \epsilon_1 \) and \( \epsilon_2 \) random numbers with uniform distribution in the interval \([0,1]\), Constriction factor \( \varphi = 0.72984 \), PSO search space interval \([-1,+1]\), maximum and minimum velocity equals to \(+1\) and \(-1\).

The network parameters, physical layer parameters and devices characteristics were set for two different situations. In the first scenario \( (S_1) \) the blocked calls are mainly due to OSNR degradation along the lightpath, i.e., the blocked calls are due to crosstalk and amplifiers impairments. Blocking due to lack of available wavelength is negligible. In the second scenario \( (S_2) \) the blocked calls are due to both lack of wavelength and OSNR degradation. The main difference between \( S_1 \) and \( S_2 \) is the number of total wavelengths available in each
link. In $S_1$ scenario we set the number of available wavelengths to 36 in order to provide very low call blockings due to lack of wavelengths. In the $S_2$ situation we decreased the number of available wavelengths to 21.

5. RESULTS

The first step before the assignment of the Equation (6) as a cost function for routing is to find the optimum values for the $b_{0,n_1}$ parameters. We have performed a search in $b_{0,n_1}$ space using PSO. The search was done using network load of 80 Erlangs. We propose to optimize for higher network loads since it is the worst case. The goal of this search is to minimize the network blocking probability (BP). In order to evaluate the fitness for a given particle, each network was simulated for a set of $10^5$ calls. The returned blocking probability $BP$ is assigned as the fitness value for this particle. We call these network simulations as offline training process since it should be done prior to network operation.

As it was discussed before, we chose two variables as input parameters for PSR cost function: link availability $x_{i,j}$ and normalized link length $y_{i,j}$. Using the best parameters $b_{0,n_1}$ found by PSO, we can plot the link cost as a function of $x_{i,j}$ and $y_{i,j}$ in terms of level curves, for both $S_1$ and $S_2$ as shown in Figure 1-a and 1-b, respectively. One can note that in both cases the link cost is high for long distances and for low link availabilities (white regions in graph) and the cost is low for short distances and high link availabilities (black regions in graph), as expected. Comparing Fig. 1-a and Fig. 1-b it is clear that the cost functions obtained for each scenario are different. It means that PSR is able to find different cost functions for different network scenarios and for this reason it can optimize the network performance. Thus, the PSR algorithm has the ability to learn with the changes in the network characteristics, in this case, with the change in the number of available wavelengths.

![Figure 1](image_url)

---

Since we have found a link cost function (Figures 1-a and 1-b) we can assign it as the network link cost and evaluate de network performance of the proposed routing scheme. We compared the PSR against three other cost function reported in literature: SP, LRW, OSNR-R. These algorithms were chosen for comparison due to following reasons: SP is simple and most largely used cost function for routing comparison purposes; LRW is an algorithm capable of finding less congested routes and, for this reason, leads to an improved network load distribution; and OSNR-R is a routing scheme that uses the physical impairments information for the routing procedure. Figure 2-a shows the blocking probability as a function of total network load for these four different algorithms for $S_1$. One can note that our proposed PSR far outperforms the results obtained using either SP or LRW algorithms. Furthermore, when compared with the IRWA approach (OSNR-R), PSR has better network performance in terms of blocking probability. It means that PSR is capable to reach the high performance of the IRWA approach not evaluating directly the impairments in real time. The impairment information was considered in the offline (training) stage only. Performing the same analysis for $S_2$, which is show in Figure 2-b, one can see that PSR also far outperforms either SP or LRW algorithms and achieved a quite similar performance to the OSNR-R algorithm.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Blocking Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSR</td>
<td>Low</td>
</tr>
<tr>
<td>SP</td>
<td>High</td>
</tr>
<tr>
<td>LRW</td>
<td>Medium</td>
</tr>
<tr>
<td>OSNR-R</td>
<td>High</td>
</tr>
</tbody>
</table>

Since we have found a link cost function (Figures 1-a and 1-b) we can assign it as the network link cost and evaluate de network performance of the proposed routing scheme. We compared the PSR against three other cost function reported in literature: SP, LRW, OSNR-R. These algorithms were chosen for comparison due to following reasons: SP is simple and most largely used cost function for routing comparison purposes; LRW is an algorithm capable of finding less congested routes and, for this reason, leads to an improved network load distribution; and OSNR-R is a routing scheme that uses the physical impairments information for the routing procedure. Figure 2-a shows the blocking probability as a function of total network load for these four different algorithms for $S_1$. One can note that our proposed PSR far outperforms the results obtained using either SP or LRW algorithms. Furthermore, when compared with the IRWA approach (OSNR-R), PSR has better network performance in terms of blocking probability. It means that PSR is capable to reach the high performance of the IRWA approach not evaluating directly the impairments in real time. The impairment information was considered in the offline (training) stage only. Performing the same analysis for $S_2$, which is show in Figure 2-b, one can see that PSR also far outperforms either SP or LRW algorithms and achieved a quite similar performance to the OSNR-R algorithm.

PSR and OSNR-R routing algorithms have quite similar performance in terms of blocking probability. However, we must also compare the time spent by these approaches to solve the RWA problem for each call. We used an Intel® Core™2 Duo @2.13 GHz with 3GB of RAM computer to perform this comparison. The results for the average time spent to solve the RWA per call, performing 50000 calls, are shown in Table I. The PSR algorithm solves the RWA problem 9.4 times faster than OSNR-R. This occurs due to the offline training based on the physical impairments evaluation. In the OSNR-R algorithm, as well as in other physical impairment based algorithm, these calculations occur during the online (call by call) solution of the RWA problem. Table I also shows that PSR is up to 1.25 times slower than LRW. This small difference should be due to the simple mathematical formula of the LRW function, which involves just a single division operation. We did not consider the SP algorithm for computation time analysis since it has a fixed routing table.
Figure 2. Network blocking probability as a function of network load for the LRW, SP, OSNR-R and PSR algorithms in: (a) $S_1$ scenario and (b) $S_2$ scenario

Table I. Average spent to solve RWA per call in $S_1$ scenario.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Time</th>
<th>Normalized Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>LRW</td>
<td>0.078 ms</td>
<td>0.0975</td>
</tr>
<tr>
<td>PSR</td>
<td>0.098 ms</td>
<td>0.11</td>
</tr>
<tr>
<td>OSNR-R</td>
<td>0.886 ms</td>
<td>1.00</td>
</tr>
</tbody>
</table>

6. CONCLUSIONS

In this paper we proposed a systematic form to build the link cost function based on a set of relevant network parameters. We applied the proposed scheme to build an adaptive cost function (PSR) for impairment aware routing in all-optical networks. The proposed PSR is based on simple network parameters such as link availability, link length and hop count. Since PSR indirectly takes into account the network physical impairments we demonstrated that it outperforms or, in worst case, provides similar performance to other algorithm that use OSNR degradation as a weight function. Moreover, the computation time for our weight function was 9.4 times faster than for the OSNR based one, for the network simulation conditions used.

It must be highlighted that the proposed weight function does not rely on online physical impairments evaluation to infer about signal noise in the network. Therefore, it is not mandatory to perform complex evaluations to obtain values for optical noise based weight functions. However, PSR requires an offline simulation to store the awareness of physical impairments in the series parameters. This characteristic of a priori knowledge brings to our weight function a drastic reduction in the computation time for real time routing decision as compared to noise based approaches.

REFERENCES