Model for Energy Efficiency Increase of Metropolitan Optical Networks

Alexandru SMEUREANU[1], Stefan Daniel DUMITRESCU[2]
[1]Academy of Economic Studies
[2]Politehnica University of Bucharest
alexandru.smeureanu@gmail.com, dumitrescu.stefan@gmail.com

The paper presents a model used for improving the energy efficiency of dense fiber optical networks present in large metropolitan areas. By reducing energy consumption the OPEX (Operating Expenditures) is decreased offering thus competitive advantage to network operators. The proposed model uses a genetic algorithm approach in order to obtain viable network topologies that have the highest degree of robustness while minimizing energy consumption.

Keywords: *Optical Networks, Graph Theory Algorithms, Energy Efficiency*

1 Introduction

As a result of the rapid increase of Internet service demand, the infrastructure that supports it has grown intensively. Much of the development was made using incremental changes and without following a precise optimal network design. This often led to sub-optimal performance and high OPEX costs. However, approaches that reduce the OPEX costs by reducing power consumptions and increasing energy efficiency can be applied.

In [2] [4] it is estimated that the energy consumption for supporting the Internet infrastructure can be as high as 4% in countries with broadband connections. Similar energy studies have indicated that ITC are indirectly accountable for 2% of the CO2 emissions because of the greenhouse effect out of which 14.8% are the contribution of the network equipments that make up the infrastructure. For example, in [1] [4] it is shown that the average power consumption for an 100mbit SFP transceiver is around 3W while the consumption for a XenPak form factor transceiver is around 7W.

This paper presents a model for increasing energy efficiency while considering the resilience of the network for a metropolitan area last-mile internet service provider that operates a FTTB (Fiber-To-The-Building) infrastructure using SFP (Small Form-Factor Pluggable) as optical fiber transmitters and managed switches and routers.

2 Problem definition

Given an existing network infrastructure represented by the graph *G(E,V)* where edge set *E* represents the optical links in the network and vertex set *V* represents the nodes, with *E>>V*, an energy consumption value is associated with every link. This value depends on the Euclidian distance between the nodes and the bandwidth of the transmitters used:

$$E = \begin{cases} 3\,W,\ \ d < 2km \\ 7\,W\ 2km < d < 7km \\ 14\,W\ 7km < d < km \end{cases}$$

The goal is how to minimize the energy consumption of the network while ensuring the best possible robustness of the network. The metric used for robustness will have to take in account the effects of network cascading and improve over all availability of the network in case of a given series of events (network failures).

3 Problem solution

An efficient optical network is a network that uses the least amount of possible interfaces. Because all the network switches or routers are connected to end-users it is not feasible to shutdown nodes of the network.

The only possibility is to use the ability of managed network switches, routers and media converter shelter units to shutdown power to the optical transponders. As a result, unused, redundant or load balanced links can be

stopped and the traffic rerouted during off-peak periods in which they are not required. It is clear that by stopping these links the robustness of the network is greatly affected. A balance must be found to ensure both energy efficiency and reliability.
The proposed model uses a two stage algorithm. The first stage represents the generation of the admissible solution space which contains a relatively large set of networks solutions that optimizes the energy consumption.
The algorithm is:

1. Initialize a state vector that contains one element for each node of the network. Each element of v[] is set to *False*
2. Initialize a vector that will contain admissible solutions *as[]*
3. Initialize a minimum energy consumption (*mec*) with the result returned by the *Mec()* procedure
4. Initialize the current energy consumption cost with zero (*cec*=0)
5. Initialize the current graph (*cg*) with a graph that contains all the nodes and no links
6. Set the root node as the current node and mark in the state vector that the node is visited: *cn=root*, *v[cn]=True*
7. Call *Explore()* procedure

procedure Explore (v[],as[],cn,mec,cec,cg)

1. For each link that is bounded by the current node:
   1.1. If there is no value unmarked in vector *v[]* then
      1.1.1. If the current min. energy consumption is equal to the min. energy consumption
         Add to the admissible solution the current graph
   1.2. If opposite node is not marked as visited in the state vector then
      1.2.1. Set the current node as the opposite of the current node
      1.2.2. Mark current node as visited v[cn]=True
      1.2.3. Add the link to the current graph
      1.2.4. Call procedure Explore(v[],s[],mec,cec,cg)
      1.2.5. Remove the current link from the current graph
      1.2.6. Mark current node as visited v[cn]=False

procedure Mec()*// Determine minimum optimum energy consumption*

1. Initialize a consumption vector(*cv*) and visited state vector(*vv*) that contains one element for each node of the network. The distance is set to infinity for all *dv* elements and with *False* for each element in *vv*.
2. Set the minimum consumption to zero
3. While there still are nodes for which *vv[node]* is *False*
   3.1. Select the node that has the lowest value in the consumption vector(*cv*) as *cn*
   3.2. Set *v[cn]=True*, add the consumption of the link to the minimum consumption.
   3.3. Recalculate the consumption vector that contain the needed energy to connect the a node to the already connected network
4. Return the minimum consumption

The minimum energy consumption (Mec()) for a given network in the previously specified optimization problem conditions is obtained by applying the minimum spanning tree algorithm on the network topology graph in which the cost of a edge is associated with the energy consumption of the optical transmitters used on that link [3].
The second stage represents the heuristic search in the admissible solution space for the solution which leads to the most robust network topology. It can be shown that two different solutions that have the minimum energy consumption can have different robustness. For example:

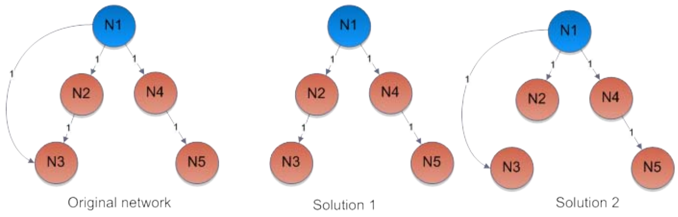

Fig. 1. Minimum energy configurations topology example

Both solutions minimize the energy consumption which in this case is 4W but Solution 1 is less robust than Solution 2 since in this case there is a higher probability that node 3 will be down more often. The probability of node 3 in solution 1 to be down is:

$$P_d^{N3} = P_f^{N1} + P_f^{N2} + P_f^{N3}$$

P_d^{N3} -Probability of node *N3* to be down

P_f^{Nx} -Probability of node *Nx* to fail

which is higher than the one in solution 2:

$$P_d^{N3} = P_f^{N1} + P_f^{N3}$$

In order to quantify the robustness differences in topologies, a special function is used that takes into account the equipments' cascading effect and also overall availability of the network. Several network failure events are randomly generated and used in all test cases to preserve test consistency. All network events have the same duration in time, are not concurrent and affect only one node at a time. However, because of the network topology, a failure can affect more than one node. During a network failure if a node has no path towards the exit router, it is considered to be down. The robustness of the network is measured as the number of operational nodes divided by the number of network events and divided by the number of nodes.

$$A(x) = \begin{cases} 1, x \text{ connected to the exit node} \\ 0, \text{if not} \end{cases}$$

A(*) the availability function
x the current node

$$I_r = \frac{\sum_{i=1}^{NE}\sum_{j=1}^{N} A(j)}{NE * N}$$

I_r the network robustness indicator
NE the total number of events
N the total number of nodes

Because the robustness function defined in the above format is not an additive value it is impossible to formulate an exact algorithm to determine the most robust network.

The first step in implementing a genetic search algorithm is to associate concepts like individual, inheritance, selection, crossovers, mutations, population and chromosomes from biology to elements from computer science optimization problems.

In the current situation an individual is a feasible solution of the optimization problem which is a sub network of the original network that minimizes the energy consumption and still connects all the nodes. Each solution is stored as a vector containing the individual names of all the links that make up the solution graph.

By analogy a population is defined as set of feasible solutions. The procedure explore from the previous step is responsible for generating a initial set of feasible solutions which becomes the initial population.

Inheritance represents the act of copying most of the network topology structure of one solution while alternating a few of the links in order to produce another solution.

The genetic searching algorithm is an iterative process. Each iteration is called a generation. In each generation the most robust solutions in the population (the fittest individuals) are selected to help obtain other better solutions (breed). Also in each generation in order to prevent the growing of the memory needed to store the population the least robust solutions are discarded (less fit individuals die).

In order to obtain new better solutions two genetic operators are applied: mutation and crossover. The crossover operator permits the combination of information for two individuals with the purpose of obtaining new individuals. There are several types of crossover techniques most important being: normal, two point crossovers, cut and splice. However in this model the normal crossover and two point crossovers are used because of the length constrain of the solution vector. This is due to the fact a number of links that make up the network must is constant and if one more link is added the minimum energy condition is not met anymore.

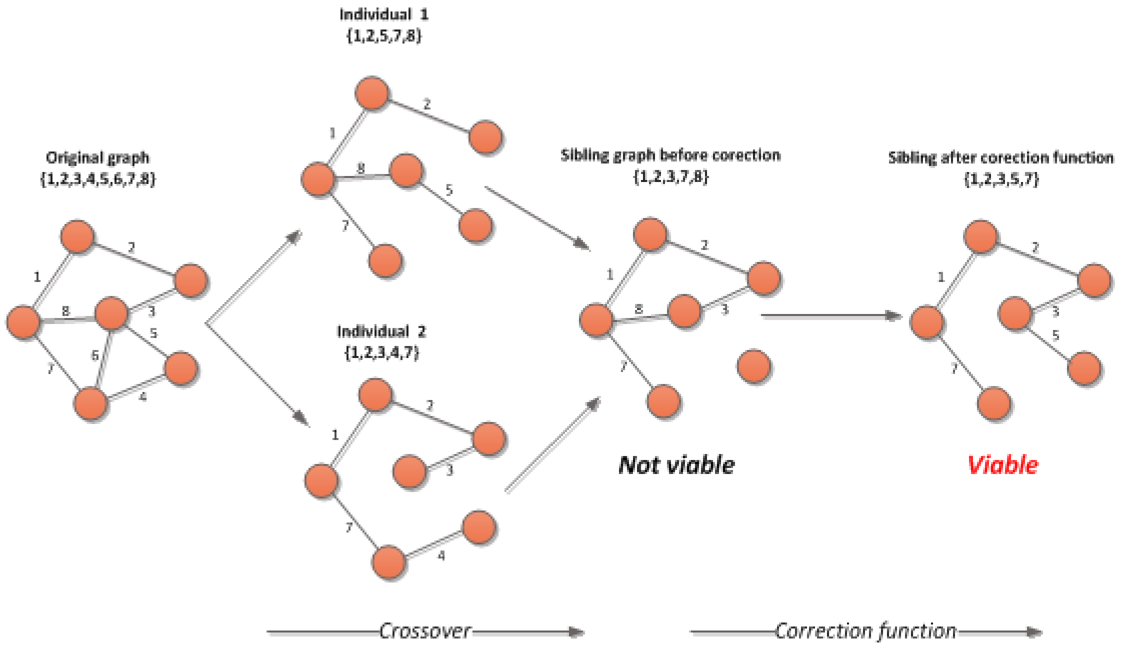

Fig. 2. Crossover between two admissible solutions

Because the network solution proposed by the resulted individual needs to connect all the nodes a special correction function is used. This function eliminates the loops and double links resulted from the application of the crossover operator, by removing them from the network topology and replacing them with links connecting the nodes that where left isolated nodes. In case of a loop the nodes originating from the dominating parent (individual that gave the most links) are preferred to be eliminated.

For better understanding in Figure 2 a crossover is presented between two individuals that represent two optimum energy efficient networks that originate from the same original network. The two individual chromosomes are represented by the sets written bellow the name that contains the identification numbers of the links that make up the proposed network solution. During the double crossover procedure presented in Figure 3 sibling is created however it is not viable because a node is left isolated as it can be seen in Figure 2.

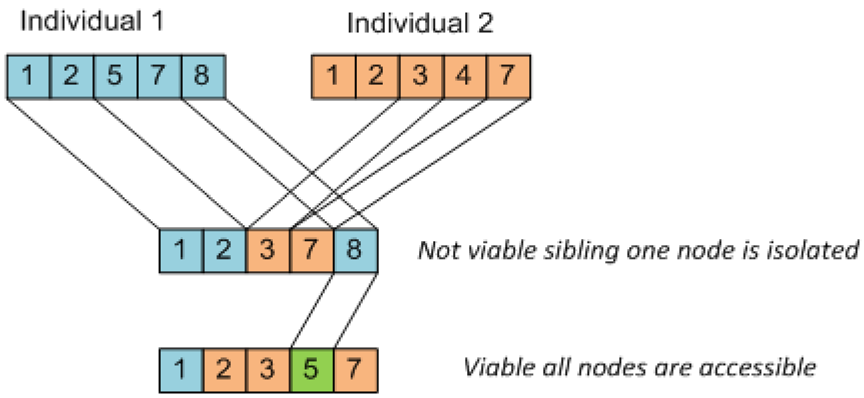

Fig. 3. Chromosomal representation of the crossover

The correction function will be then used to obtain a viable sibling. The function will eliminate the link 8 because it is inherited from the dominating individual and replace it with link 5 which will allow the reconstruction of the network.

The mutation operator randomly changes a part of the information in one individual. Because of the problem constrains mentioned previously in the current model the mutation algorithm removes one link from the topology replaces it with another link. The solution is then checked for viability and if it is not viable the correction function also used for crossover is applied.

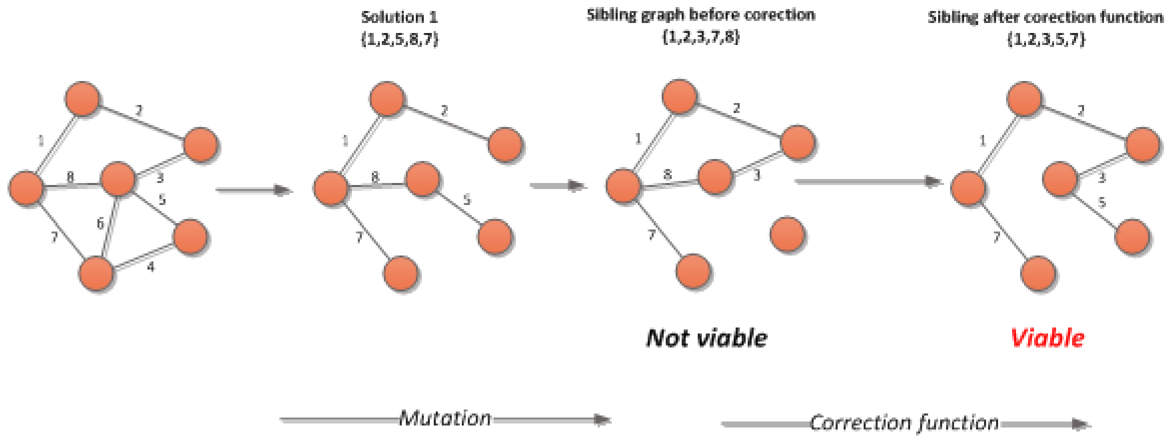

Fig. 4. Mutation example for network topologies

In Figure 4 similar the crossover operator the resulted sibling is not viable so the correction function is applied. As a result link 8 is replaced with link 5.

The algorithm stops when the resulted population meets a predefined fitness requirement or when the evolution process has stagnated over multiple generations indicating that the algorithm has converged towards a stable solution.

This step is performed by using genetic algorithms. The algorithm steps are:

1. Initialize the population starting from the admissible solutions generated in stage 1 of the algorithm
2. Evaluate the fitness of the population by calculating the robustness of each individual in the population
3. Until the robustness of the fittest individual in the population doesn't change within ε do:
   3.1. Select the most robust solutions for networks and produce cross-over and mutation operations on them

3.2. Evaluate whether the resulted networks are still connected and they have an energy consumption equal to the minimum energy consumption solutions. If these requirements are not met the solution is discarded.
3.3. Evaluate the robustness of the solution and replace the solutions from the population which are less robust

4 Evaluation

The performance evaluation of the model needs to be done from three points of view:

1. Economical benefits reflected by the energy consumption savings
2. Time of the algorithm to converge as a function of the network complexity degree:

$$NC = m - n$$

NC network complexity degree
m number of links in the network
n number of nodes in the network

3. The robustness of the model proposed solution versus the initial robustness average of the admissible solutions that make up the initial population of the genetic algorithm.

The economical benefits are in a direct relation with the energy consumption savings. The energy consumption is in turn in direct relation with the number of links that can be safely shut down during off peak hours. The complexity degree itself can reveal a maximum threshold for energy savings. The proof lies in the fact that the smallest network possible that still connects all nodes is a tree and in a tree with n nodes there are exactly n links. This means that if no bandwidth requirement on a link is exceeded then the most energy efficient network would in fact be a tree.

In order to estimate the amount of energy savings we can also define a maximum power saving forecast which is equal to the network complexity degree multiplied by the maximum consumption of the optical media convertors. This function overestimates the effective energy savings but offers a good estimate for network topologies with $NC < 30$.

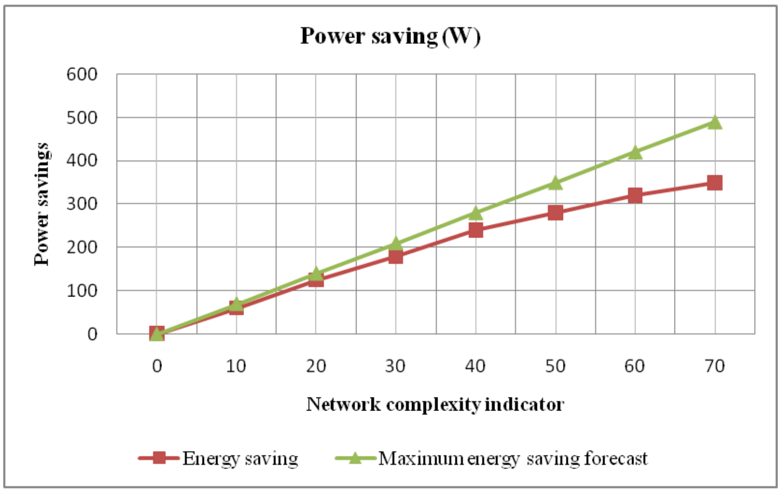

Fig. 5. Power consumption saving

Figure 5 represents the dynamics between the power savings and maximum energy saving forecast as a functions of the network complexity. In order to obtain the energy savings the power savings need to be multiplied with the time in which the network operates in the reduced form.

The algorithm convergence time can be hard to estimate. Some of the algorithms involved in the model can be predicted quite well. For instance the robustness function call has the following complexity:

$$CI_r = N * O(n)$$

n number of nodes in the network

N number of random generated events

The complexity is given by the fact that there are N repetitive event and in all of them in order to know how many nodes are functional a depth first traversal is needed. From this complexity estimation we can estimate how fast is a call to this function. However for other components such as the genetic algorithm search estimating the convergence time is not possible. In order to estimate the convergence time several simulations have been done (see Figure 5).

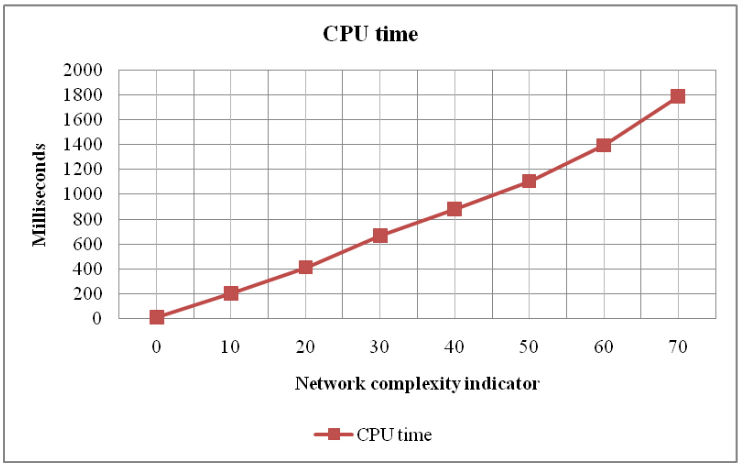

Fig. 6. CPU Time needed as a function of network complexity

Similar to the previous simulation, a fixed network made up of *n* nodes was used. Within this network the number of links *m* was gradually validated from *n* to *n*+70. The time needed to solve the solutions was presented in Figure 6. The figure shows an optimistic linear dependency of CPU time to network complexity.

The robustness of the proposed solutions was evaluated using the same network topologies used in the previous setups. For reference, the robustness of the initial network is also presented.

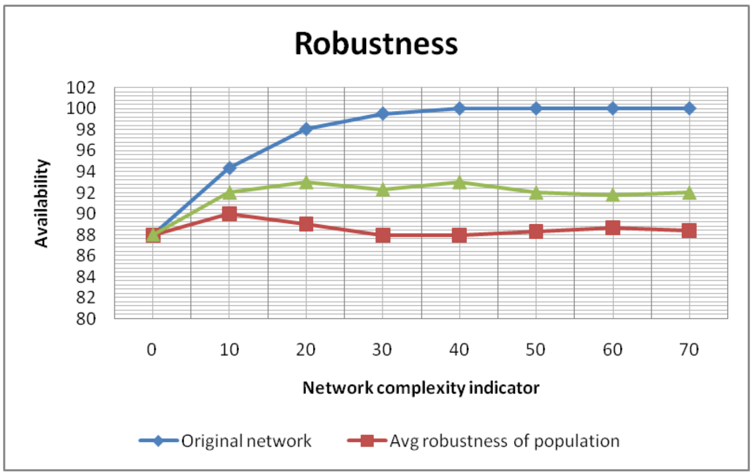

Fig. 7. Robustness performance analysis

The results of the simulation presented in Figure 7 show that there is a dramatic decrease in the robustness of the network when links are removed. However this is expected and has to be assumed by the internet service provider. The comparison between the initial population and the solution provided by the genetic algorithm search shows that there is a constant improvement over the whole network complexity range.

5 Conclusions

By analyzing the current trends it is clear that the Internet is still undergoing an extensive growth. It is mandatory that future network planning especially for last mile connections take into the consideration the possibility of reducing energy consumption and thus reducing CO2 emissions.

Also a reducing OPEX costs could also represent the key to a competitive advantage on a market in with the margin for profit is becoming thinner and thinner because of competition.

The proposed algorithm shows good results especially for Internet Service Providers that have a large number of redundant links. Energy savings can reach 6,5% for simulations ran based on real network topologies.

Acknowledgments

Some parts of this article are results of the projects: 1. „Doctoral Program and PhD Students in the education research and innovation triangle”. This project is co funded by European Social Fund through The Sectorial Operational Programme for Human Resources Development 2007-2013, coordinated by The Bucharest Academy of Economic Studies (Project no. 7832, “Doctoral Program and PhD Students in the education research and innovation triangle, DOC-ECI”). 2. POSDRU, financing contract POSDRU/6/1.5/S/19.

References

[1] B. G. Bathula, J. M. H. Elmirghani, “Green Networks: Energy Efficient Design for Optical Networks”, *Sixth IEEE International Conference on Wireless and Optical Communications Networks* (WOCN2009), April 2009.

[2] J. Baliga, K. Hinton, R. Tucker, “Energy consumption of the Internet”, *COIN-ACOFT*, Melbourne, Australia, Jun.2007

[3] T. Cormen, Ch. Leiserson, R. Rivest, C. Stein “Introduction to Algorithms”, Third Edition, MIT Press, 2009

[4] C. Pettey„Gartner Estimates ICT Industry Accounts for 2 Percent of Global CO2 Emissions*”*, http://www.gartner.com/it/page.jsp?id=503867

[5] M. Pickavet, W. Vereecken, S. Demeyer, P. Audenaert, B. Vermeulen, C. Develder, D. Colle, B. Dhoedt and P. Demeester , “World wide energy needs for ICT: The rise of power-aware networking”, in Proceedings *IEEE ANTS*, Dec.2008, pp.1-3.

Alexandru SMEUREANU graduated Politehnica University of Bucharest, Automatic Control and Computers Faculty, Computer Science Department and Academy of Economic Studies, the Faculty of Cybernetics, Statistics and Economic Informatics. He is currently a PhD student in the field of Economic Informatics at University of Economics. His interests range in the software programming, network management, GIS systems, vector graphics and embedded devices programming.

Stefan Daniel DUMITRESCU graduated Politehnica University of Bucharest, Automatic Control and Computers Faculty, Computer Science Department. He is currently a PhD student in the field of Information Extraction at Politehnica University of Bucharest. His interests range in the IT applications with accent on online development, information extraction, knowledge representation and human-computer interaction. He is also interested in project management and skilled in network technologies.