Metropolitan fiber optical network planning model based on self organizing neural networks

ALEXANDRU SMEUREANU
Economic Informatics
The Bucharest Academy of Economic Studies
Calea Dorobantilor nr. 15-17, Bucharest
ROMANIA
alexandru.smeureanu@gmail.com
STEFAN DANIEL DUMITRESCU
Computers and Information Technology
Politehnica University of Bucharest
Splaiul Independentei nr. 313, Bucharest
ROMANIA
dumitrescu.stefan@gmail.com
LIVIU ADRIAN COTFAS
Economic Informatics
The Bucharest Academy of Economic Studies
Calea Dorobantilor nr. 15-17, Bucharest
ROMANIA
liviu.cotfas@ase.ro
ANDREEA DIOSTEANU
The Bucharest Academy of Economic Studies
Calea Dorobantilor nr. 15-17, Bucharest
ROMANIA
andreea.diosteanu@gmail.com

Abstract: The paper presents a model for metropolitan optical network design that relies on self organizing neural networks and graph theory adapted algorithms. The resulted network is a two layered hierarchical network; the first layer represents the core of the network and has a ring topology while the second layer has a tree topology with a maximum depth constraint. The model is aimed towards providing a close to optimal solution for large problem configurations that are unsolvable in polynomial time.

Key-Words: neural networks, graph theory, large networks

1 Introduction

The paper presents the development of an optimum optical infrastructure planning model based on self organizing neural networks. This system has been added as a functionality into an preexisting Network Management System[1].

The strong demand for Internet services in the recent years has had a great impact in the development of communication networks. Because rapid growth was, and still is, based on ad-hoc decisions taken in the absence of a development plan, in time it has lead to rather poor network architectures (incurring efficiency and reliability issues) and other inherited design problems.

The current trend of technology in the last decades has shown optical networks are the preferred choice in infrastructure development both for the core of the Internet and last mile connections in metropolitan area networks. The advantages of using fiber optics in metropolitan area networks is that it has virtually no constrain on cable length. O the other side, active terminator equipments are expensive.

The introduction of Voice and Video over IP (VoIP and VVoIP) communications and IPTV service has increased the need for reliable and robust network designs. One of the best ways to increase the reliability and robustness of a network is to introduce redundancy.

Redundancy can be achieved in several ways in optical networks.

2 Redundancy in IP networks

The best way to ensure redundancy in IP networks is to have two different physical (and geographical) optical circuits between each node. In this situation, because the two links start in the same device and end in another,

ISBN: 978-960-474-281-3

they can be aggregated at Layer 2 level using a link aggregation or port trunking technologies such as EtherChannel [2]. The advantage in this case is that the two links can be used simultaneous and bandwidth is cumulated. If one of the links fails there is very little down time (less than 1 second). However this solution is not cost effective because is doubles both the number of links and interfaces needed for every device in the network.

The less expensive solution is to use ring topologies together with Layer 2 protocols or Layer 3 protocols to do load balancing and reroute traffic when one link fails. However, these protocols have higher convergence time in case of a topology problem and the network is still vulnerable if two distinct incidents occur in the same time. This will be reflected further on.

Giving the economical constraints of the technical solutions, the Internet Service Providers mainly rely on ring topologies and use optical cables that enhance reliability and offer high bandwidth capabilities. In order to improve redundancy, the network is split into several regions that have important devices interconnected in a ring.

The goal of our model is to obtain a near-optimal solution to connect several locations within a city, respecting a given network hierarchy and minimizing the cost expressed as geographical distance (i.e. minimizing cable length).

3 Problem statement

Given a set *V* containing geographical locations that need to be interconnected and the maximum number of sub-networks *n*, it is requested to generate an optimum graph *G(V,E)* representing the network infrastructure, where *E* represents a set of links between locations in *V*, and *G(V,E)* has the following properties:

- $\exists G'(V',E')$ which is a sub graph of *G(V,E)* where $|V'| = n$ (| | cardinal operator) and $|E'| = n$ and *E'* forms the Hamiltonian circuit.
- $\exists G_i''(V_i'',E_i'')$ where $0 \leq i < n$ and:
  - $E_i'' \cap E_j'' = \emptyset$ where $i \neq j, 0 \leq i,j < n$
  - $|E_i'' \cap E'| = 1$ where $0 \leq i < n$
- Each $G_i''(V_i'',E_i'')$ is a minimum spanning tree of the complete graph formed with the V_i'' where $0 \leq i < n$
- $n << |V|$ (cardinal operator)
- $E_i'' \cap E' = \{(x_1, y_1)\}$ where $d\left((x_1,y_1),\left(\overline{x_{V_i''}},\overline{y_{V_i''}}\right)\right)$ is minimal.
- The sum of all the Euclidian distances of the links (edges) in E is minimal.

The definition of the problem is focused at this point only on the economical constraints.

The network architecture that is expected as a solution has the following design:

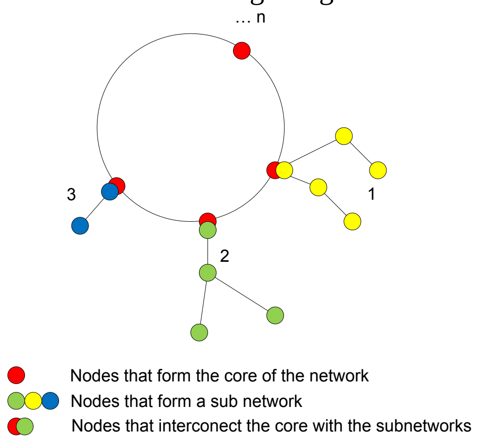

Fig. 1 Proposed network architecture

The resulted network will have a two layered hierarchy in which the first layer will represent a core network that will have a ring topology respecting the minimum cost constraint. The second layer is formed by *n* sub-networks each having a tree topology.

The input parameter *n* defines the number of sub-networks that need to be created. Usually, the higher the number of sub-networks, the more resilient the network as a whole becomes.

The proposed problem can be regarded as two different sub-problems. The first is to group the locations in clusters that define a geographical region and to solve the minimum spanning tree problem (MST) within this region. The complexity and quality of the grouping is strictly related to the clustering algorithm and for the moment we assume it being polynomial. The complexity of the MST is at most O(nlogn). The second sub-problem is to solve a Hamiltonian circuit, also known as the Traveling Salesman Problem (TSP)[3] for the nodes selected to be part of the metropolitan ring. The formulation of the TSP problem is simple: find the minimum-length tour trough N cities by visiting each city exactly once. Although the problem seems simple and humans are known to offer good solutions even for small-medium size ones [4], the problem is NP-Complete [5] and thus unsolvable in polynomial time at the moment. The overall solution of the optimum problem, assuming that the clustering algorithm is ideal and offers the optimum clustering solution, is still going to be NP-Complete because of the TSP problem.

4 Problem solution using self organizing neural networks

4.1 Self organizing neural network

Self organizing neural networks are categorized as unsupervised learning neural networks. They are designed to discover correlations in the input data and to form categories. This type of neural network is inspired by the property of the human brain to self organize upon receiving inputs presenting similar patterns. In literature neural networks and especially associative neural networks have been used to solve optimum problems [6]. The general approach in these problems as well as in our situation is to define an energy function similar to the potential energy in physics. The stability of the entire system is determined by the energy function. Similar to physics, the lower the energy of the system, the more stable it is. All the imposed restrictions of the optimum problem need to be reflected in this function. The tendency of the system to shift towards a lower energy configuration during the simulation is the only mechanism leading to a feasible solution that in this situation minimizes the cost. Although this approach can be applied in several combinatorial problems like the ones presented in this paper, it doesn't guarantee that an optimum solution will be obtained.

4.2 Solving the partitioning problem

The general cluster analysis is composed out of two important steps: determination of the number of clusters needed and the classification of the data into clusters. In the proposed model the cluster analysis focuses on the second part. A self organizing neural network algorithm is used to group the locations in geographical regions that later are interconnected to form a given number of sub-networks which is an input parameter of the model.

The self organizing neural network consists of an input layer and the Kohonen layer. The Kohonen layer is usually designed as a two-dimensional arrangement of neurons that maps an N-dimensional input to two dimensions, preserving topological order, but for the purpose of identifying cluster membership, we use a one-dimensional Kohonen layer.

The input layer of neurons is fully connected to the Kohonen layer. The Kohonen layer computes the Euclidean distance between the weight vector or more intuitively the current location of the neurons in the 2D space and the input pattern. This is done for all the neurons in the network. The closest neuron in terms of Euclidian distance is selected winner and is activated together with the other neighbor neurons.

The Kohonen layer of our model is a one-dimensional array of neurons, indexed by i where $0 \leq i < n$. The index indentifies a specific location and also indicates the neighbor relation between two nodes by the absolute difference between the two indexes.

Because self organizing neural networks are trained by an unsupervised competitive learning algorithm, a process of self organization takes place [7]. The neuron that was declared winner updates its weights and migrates towards the current input location. The migrated distance is equal to the distance between the neuron and the current location multiplied by the learning rate. Similar to the winner neuron, the neighbor neurons are also updated with the same value multiplied by a penalty function also known as the neighborhood function.

$$d(i, w) = |i - w|$$

d	absolute index distance function
i	the coordinates of current location
w	the winner neuron index

$$f(d, \theta) = \begin{cases} e^{-(d\theta)^2} \; where \; d < n/10 \\ 0 \end{cases}$$

$f(*,*)$	the neighborhood function
n	the total number of sub networks
θ	neighbor influence loss coefficient

In one epoch all the locations are used as input of the neural network once. The network will enter another epoch if at least one neuron has changed its location in the last epoch with a distance $\Delta d > 0.1$. This is also known as convergence condition.

During the self organization process, the weights of the neurons asymptotically converge towards the values of cluster centroids. These weights are used afterwards to partition the locations into geographical regions. All the locations from one region will be interconnected by the same sub-network. When the neural network has converged, the weights of each neuron represent the coordinates of a cluster centroid. Each location is then associated with the cluster whose centroid is the closest.

The clustering algorithm is the following:

1. Initialize n neurons with random position distributed around the geometrical center of the locations$(\overline{x_V}, \overline{y_V})$. Initialize learning rate $\mu = 0.6$, energy loss coefficient $\theta = 0.1$, epoch counter $i = 0$ and $\alpha = 0.12$
2. Initialize a list that contains all the locations V and perform a random shuffle
3. Initialize a list with the initial location of each neuron
4. For each location in the list
    - For each neuron select the neuron w closest to the current location
    - Adjust the neighbor neurons keeping in mind the index distance from the current neuron in the hierarchy
5. Increment epoch

ISBN: 978-960-474-281-3

$$i = i + 1$$

6. Adjust the neighbor influence loss coefficient by multiplying it with the learning rate.
$$\theta_i = \mu * \theta_{i+1}$$
7. For each neuron calculate the distance Δd covered during the last epoch using the list saved in step 3.
    - If $\Delta d < 0.1$ go to step 2.
8. For each location in *V* find the closest neuron and assign the location to the cluster represented by the neuron

4.3 Solving the optimum ring topology

For solving the second part of the problem the following neural network architecture is used:

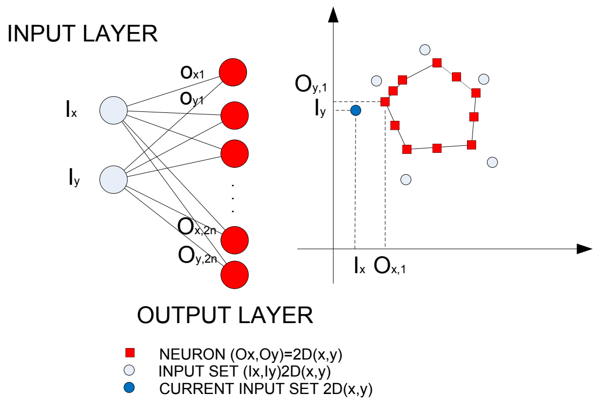

Fig. 2 Neural network structure

The inputs of the neutral network are two-dimensional and represent the coordinates of the location. There are exactly *2n* neurons. Each has two weights corresponding to the two inputs that also represent the output. In addition to the traditional neural networks, the neurons are linked together in a ring topology. Each neuron *i* is linked with two adjacent neurons with the rank *((i+1) mod n)* and *((i-1+n) mod n).*

In each epoch a complete list of locations is generated and randomly shuffled. For each location, the neuron closest to the location is declared the winner. A neuron can only be a winner once in epoch and consequently only for one location. The weights of the neuron are adjusted to shift it towards the current location. The shift is equal with the difference between the winner neurons coordinates and the location coordinates multiplied by the learning rate. The rest of the neurons are updated in a similar way by the formula:

$$O_{i,t+1} = O_{i,t} + \mu * f\big(d(i,w)\big) * (I_c - O_{i,t})$$

$O_{i,t}$	the output of neuron *i* in epoch *t*
I_c	the coordinates of current location
μ	the learning rate
w	the winner neuron index
$f(x)$	the neighborhood function
$d(i,w)$	absolute index distance function

$$d(i,w) = \min(n - |i - w|, |i - w|)$$

n	the total number of locations
i	the coordinates of current location
w	the winner neuron index

$$f(d,\theta) = \begin{cases} e^{-(d\theta)^2} & where\ d < n/5 \\ 0 & \end{cases}$$

n	the total number of locations
d	absolute index distance function
θ	energy loss coefficient

The algorithm for solving the optimum ring topology is:

1. Initialize *2n* neurons with the coordinate of a circle located in the geometrical center of all the locations $(\overline{x_V}, \overline{y_V})$. Initialize learning rate $\mu = 0.6$, energy loss coefficient $\theta = 0.1$, epoch counter $i = 0$ and $\alpha = 0.12$
2. Initialize a list that contains all the locations designated to be part of the core ring and random shuffle the list.
3. Initialize neuron winner list *lw*
4. For each location in the list
    - For each neuron select the neuron *w* closest to the current location that is not in the list *lw*
    - Adjust the neuron's weights and add it to the *lw* list
    - Adjust the neighbor neurons keeping in mind the distance from the current neuron in the hierarchy
5. Adjust the energy loss coefficient by multiplying it with the learning rate.
$$\theta_i = \mu * \theta_{i+1}$$
6. Increment epoch
$$i = i + 1$$
7. For each location *L* in the list
    - Initialize $S = 0$
    - For each neuron select the neuron closest to the current location as *P*
    - $S = S + d(L, P)$
8. If $S > 10^{-8}$ go to step 2.
9. For each neuron in index order *i* find the location L_i
    - Skip if is the first one
    - Add to *E* part of the problem solution *G(E,V)* the link (L_i, L_{i+1})

5 Model performance and results

In order to test the performance of the model, a standard problem formulation was used. This formulation is

ISBN: 978-960-474-281-3

defined by: 200 locations that need to be connected along with their geographical locations, 10 sub-networks and 10 levels maximal depth allowed in a sub-network. All the points are located within an area of 1.5 km x 1.5 km.

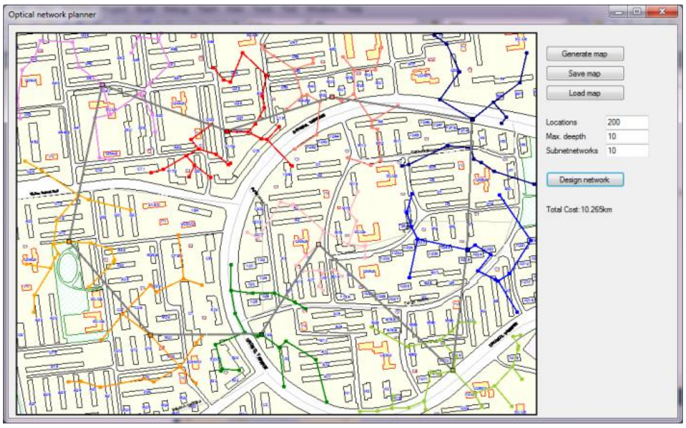

Fig. 3 Model proposed network structure

The performance of the solution is evaluated based on the cost of implementation as a function of the total amount of fiber optic cable used (less is better), along with the robustness of the network (more is better).

In order to evaluate the robustness, several network failure events are randomly generated and used in all test cases in order to preserve 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. The first node of the ring is considered to be the network exit router. If during a network failure 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.

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

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

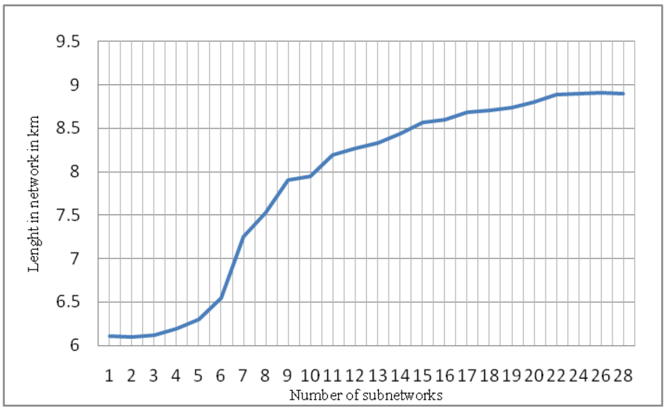

Fig. 4 Total cable length of the proposed infrastructure

The chart in Fig. 4 reveals the relationship between the total amount of fiber optic cable needed to build the proposed the infrastructure and the number of sub-networks.

The results indicate that the total amount of cable increases with the number of sub network. Also, it shows that there are two asymptotic limits both of the minimum amount and at the maximum amount of cable needed. These limits correspond to less than three sub-networks and respectively more than twenty sub-networks.

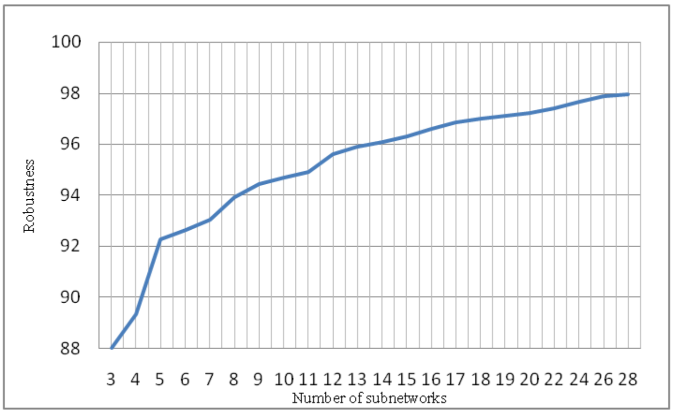

Fig. 5 Network robustness as a function of number of sub-networks

The result clearly shows the increase in robustness as the number of sub-networks increases. However, the increase is not linear, but more logarithmic in nature, as the robustness increase amount is getting smaller as more sub-networks are added. It is interesting to note, that even for three sub-networks linked in a triangle shape, the robustness is rather high (88%).

6 Conclusions

The purpose of our work was to create a network design model that allows an efficient planning of a metropolitan area network. The model has been calibrated and integrated in a network management system [8] that works with real geographical data.

ISBN: 978-960-474-281-3

The proposed optical network design is a balance between cost and redundancy. Because of the hierarchical structure of the network solution and the methods involve in solving the problem it is impossible to obtain an exact optimum solution. However, the self organizing neural network algorithms used offer a very close approximation of the optimum solution. In [9] solutions for solving the optimum ring topology show results within 1.3% of optimality in comparison to other traditional approximation methods that are within 3% of optimality.

The model can also be used by management as a method of assessing the cabling cost involved in building the infrastructure.

Although in real applications is not always possible to implement the exact proposed network and adjustment need to be made, the proposed network layout still offers very good results.

Further extension of the model will take in account elements such as: including in the cost of the network the price of the media convertors, router interfaces and operational costs related to equipments.

For a more realistic modeling of the sub network, information from the GIS system could be used, such as: road, underground ducks and building infrastructure data.

7 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]Dumitrescu S., Smeureanu A., Diosteanu A., Cotfas L. A., *Adaptable Network Management System using GIS and network ontology*, 9th RoEduNet International Conference, 2010, pp.310–315.

[2]Cisco online white paper, Cisco EtherChannel Technology http://www.cisco.com/en/US/tech/tk389/tk213/technologies_white_paper09186a0080092944.shtml#wp34608, retrieved, Jan 2011

[3]Gutin G., Punnen A. P., *The Traveling Salesman Problem and Its Variations*, Springer, 2006.

[4]MacGregor J. N., Ormerod T., *"Human performance on the traveling salesman problem"*, Perception and Psychophysics Vol.4, No.58, pp. 527–539

[5]Garey MR, Johnson DS, *Computers and Interactability:A Guideto the Theory of NP-completeness*, Freeman: San Francisco, 1979

[6] Hopfield JJ,Tank DW, *Neural computation of decisions in optimization problems.* Biological Cybernetics No.52,1985, pp141-152.

[7]Burke LI., Ignizio JP. , *Neural networks and operations Research*, Comp Ops Res, Vol.19, No.4, 1992, pp. 179-189.

[8]Cook, William; Espinoza, Daniel; Goycoolea, Marcos, *Computing with domino-parity inequalities for the TSP*, INFORMS Journal on Computing, Vol.19, No.3,2007, pp 356–365.

[9] Smeureanu A., Dumitrescu D.S., *Implementing a GIS application for Network Management*, Journal of applied quantitative methods, Bucharest, Vol.5, No.1, 2010

[10]Somhom, A. Modares, T. Enkawa, *A Self-Organising Model for the Travelling Salesman Problem*, The Journal of the Operational Research Society, Vol. 48, No. 9 (Sep., 1997), pp. 919-928

[11]Sagy Bara, Mira Gonena, Avishai Wool, *A geographic directed preferential internet topology model*, Computer Networks, Vol.51, No.14, 2007, pp.4174-4188