Review Of Research
Vol.1,Issue.VIII/May; 12pp.1-4 ISSN:-2249-894X

Network Optimization with Shortest-Path Inter-domain Routing Algorithm Using Self Organizing Map

Ambika Jaiswal
PG Department of
Computer Science,
SGB Amravati University,
Amravati, INDIA

Anupama Sakhare
Department of Electronics and
Computer Science ,
RTM Nagpur University,
Nagpur, INDIA

Abstract

A computational method for stable optimum route finding in the inter-domain network using the self organizing map.The present invention generally relates to the route finding mechanism using neuro-dynamic algorithm used in computer network. The present work relates more specifically to a route finding system which finds an optimum data path in an inter-domain network by using self organization network map.

1. Prior Art :

S.No.	Prior Art and its Drawbacks	Present Invention
1.	EP 1 062 616 [11] Sub-paths are pre-computed.	The sub-paths and the network is organized completely during runtime.
2.	EP 0 852 862 [12] The agents operate node by node thereby consuming more time.	At a time the entire map is considered for route optimization processing.
3.	EP 1 192 555 [10] Operates only with the central route server.	It does not require any central route-several but it operates on the entire map of inter domain network.

lOwing to the drawbacks stated above it was felt essential to establish a system for route finding having the mechanism to seek optimum path by using data routing method which can implement the self organization network map establish stability and optimizes the network in the inter-domain network.

Please cite this Article as : Anupama Sakhare and Ambika Jaiswal, Network Optimization with Shortest-Path Inter-domain Routing Algorithm Using Self Organizing Map : Review of Research (May ; 2012)

2. Introduction

A method and a system for implementing a neuro – autonomous system topology controller .This system uses the internet exchange points as its data input and determines the optimal network of a packet's tour by training a self organizing map using simulation.

Autonomous system topology map - its methodology and determination thereof is implemented by the following method : -

1. **A one-dimensional linear self–**organizing map is trained from an initial neural network that includes a plurality of neurons and the number of neurons is small in comparison to the number of autonomous systems (i.e. not greater than one-third the number of autonomous systems).

2.1 A neuron is then associated with a number of autonomous systems.
2.2 The neighborhood topology of autonomous system is formed.
2.3 Once the neighborhood topology is formed a winning neuron selected.
3. This winning neuron associates itself with the neighborhood attached to the winning neuron.
4.Steps 2-3 are repeated until the neurons in all the sub-topology are all connected together.
5.The final neural network output is a Self-organized Autonomous System Map.

3.COMPUTATIONAL POWER OF SOASM ALGORITHM

The computational power of a AS-PATH self organizing map is embodied in two main theorems:

Limit Theorems of Adaptation

Theorem 1

Let > 0, and s = XT W. Let (s) be an arbitrary scalar function of s such that E [(s)] exists. Let X(t) $\in R^n$ be a small random number. $\overline{x}$ being the mean of X(t) being independent of W. If equations of the form

$$\dot{W} = E\,[\alpha\, X - \gamma(s)\, W] \qquad (1)$$

Have non – zero bounded asymptotic solutions, then these solutions must have the same direction as that of $\overline{x}$.

Theorem 2

Let , s and (s) be the same in Theorem 1. Let R = E[xxT] be the correlation matrix of X. If equations of the form :

$$W = E[s\ X -(s)\ W] \qquad (2)$$

These are non-zero bounded asymptotic solutions, then these solutions must have the same direction

as η_{max} , is the maximal eigenvector of R with eigen value λ_{max}, provided $\eta^{T}_{max}\, W(0) \neq 0$.

4. FORMULATION OF THE PROBLEM

Assume that the input vector X_k, = $\left(x\frac{k}{1}.....,x\frac{k}{n}\right)^T \in R^n$ (R^n being the pattern space in question) is presented to a (m x m) field of neurons. Due to the planar nature index ij, ij = 1, …, m. The ijth neuron has an incoming weight vector $W_{ij}(k) = \left(w\frac{k}{1ij},.....,w\frac{k}{n\,ij}\right) \in R^n$, at time instant k.

The first step is to find the best matching weight vector Wij(k) for the present input, and to thereby identify a neighborhood Nij around the winning neuron. One can find the best matching weight vector by comparing the inner products $X_k^T\, Wij(k)$

of the impining input Xk with each weight vector Wij(k). The winning neuron is the one that has the largest inner product. Equivalently, with normalized weight vectors we have seen that the maximum inner product criterion reduces to a minimum Euclidean distance criterion. The winning neuron is the one that minimizes the distance ||Xk = Wij(k). || Kohonen suggests using the later since it is more general and applies to natural signals in metric vector spaces :

$$\|X_k - W_{ij}(k)\| = \min_{i,j} \{\|X_k - W_{ij(k)}\|\} \quad (3)$$

The SOFM algorithm assumes that the planar field of neurons has some kind of Mexican hat interaction that allows the identification of a neuron cluster around the winning neuron. For simulation purposes it is expedient to define the topological neighbourhood Ɲij of indices of neurons in a region surrounding the winning neuron with index IJ. The shape of this neighbourhood might be either square or hexagonal and the width of the region around the winning neuron IJ is specified by a radius r measured discretely by terms of the number of neurons.

The width of the neighbourhood is a function of time - as epochs of training elapse, the neighbourhood shrinks. In the continuous version this shrinkage of neighbourhood width can be implemented by gradually decreasing the positive lateral feedback and increasing the negative lateral feedback in the Mexican hat function. The use of Ɲij simulates the quick formation of an activity cluster. Once a winning cluster has been identified, weights of neurons within the cluster are updated.

Adaptation in SOFM's takes place according to the second generalized law of adaptation.

$$\omega_{i,ij} = \eta x i s_{ij} - \gamma(s_{ij}) w_{1,ij} \quad (4)$$

which incorporates learning according to the Hebbian hypothesis but with a forgetting term scaled by some function $\gamma\,(s_{ij})$ of the neuronal signal sij.

$$\gamma\,(s_{ij}) = \beta\, s_{ij}$$

which yield :

$$w_1.ij = \eta x_1 s_{ij} - \beta s_{ij}\, w_{1ij} \quad (6)$$

5. Operational summary of the SOFM algorithm

Step 1. Initialization: Choose random values for the initial weight vectors Wj(0). The only restriction here is that the wj(0) be different fo j = 1,2 ,….l, where l is the number of neurons in the lattice. It may be desirable to keep the magnitude of the weights small.

Another way of initializing the algorithm is to select the weight vectors [wj (0)]ij= 1 from the available set of input vectors [xi]Ni=1 in a random manner.

Step 2. Sampling: Draw a sample x from the input space with a certain probability; the vector x represents the activation pattern that is applied to the lattice. The dimension of vector x is equal to m.

Step 3. Similarity Matching: Find the best – matching (winning) neuron i (x) at time step n by using the minimum – distance Euclidean criterion:

i (x) = arg min ||x(n) – wj||, j = 1, 2,…,l

Step 4. Updating : Adjust the synaptic weight vectors of all neurons by using the update formula

$$wj(n+1) = wj(n) + \eta\,(n)\, hj,_{i(x)}\,(n)\,(x)\,(n) - w_j(n))$$

where η is the learning – rate parameter, and hj,I(x) (n) is the neighborhood function centered around the winning neuron I(x); both η (n) and hj,I(x) (n) are varied dynamically during learning for best results.

Step 5. Continuation: Continue with step 2 until no noticeable changes in the feature map are observed.

6.COMPUTER ORIENTED EXPERIMENT FOR NUMERICAL ANALYSIS :

In this experiment we assume an 8 x 8 planar array of neurons. We want to derive a topological map that would correspond to a uniform distribution of 500. Autonomous Systems, in the 2D bipolar square [-1, 1] x [-1, 1] . These data are plotted in fig. 1.

The process of adaptation is initiated by assuming the initial weight vectors to be the

Autonomous systems (shown as random points) distributed in the input pattern space.
In this experiment the weight vectors are initialized to random points in [-1,1] x [-1,1]
The learning rate was initialized to 0.9 and it is assumed to decrease linearly for the first 999 epochs in accordance with the function :

$$\eta = 0.9 \; 1 - \frac{\text{epoch}}{1000}$$

The first 999 epochs comprise the ordering phase. At the end of 999 epochs the learning rate was maintained at 0.005 through the convergence phase.
The initial value of the neighborhood radius was 6

The neighborhood is a square initially with width of 12 centered around the winning neuron (IJ).
The neighborhood width contracts by 1 every 200 epochs.
After 1000 epochs (i.e. end of ordering phase) the nbd is maintained radius is maintained at 1.

The winning neuron and its 4 adjacent neurons update their weights on all subsequent iterations. Only the winning neuron updates its weight

7. CONCLUSION

INITIALIZATION PHASE

The initial value of nbd = 6
The initial neighborhood was a square of width 12 centered around the winning neuron (IJ).
CONVERGENCE PHASE
The neighborhood width contracts by 1 after every 200 epochs.

ORDERING PHASE

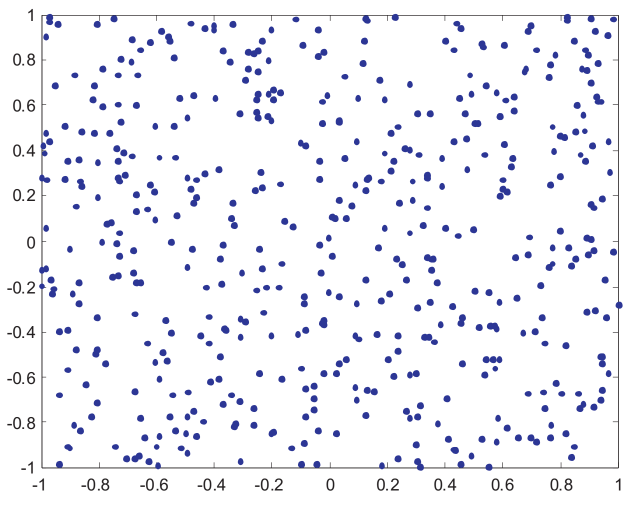
After 1000 epochs, i.e. end of ordering phase the nbd radius was maintained at 1.

UPDATION PHASE

The winning neuron and its and adjacent neurons were designated to update their weights on all subsequent iterations.

8. Snapshots of the simulation of the SOAS Map

Fig.1 Shows the initial map which represents a completely randomized topology of the input space.
Fig.2 Reconstructs the Kohonen maps at different epochs of the training process.
During the ordering phase the map unfolds to approximate the shape of a square.
The convergence phase refines the topological features of the map.
The successful simulation of SOASM depends on a careful choice of parameters.
At the end of 999 epochs, the learning rate was maintained at 0.005 through the convergence phase.



Review Of Research
Vol.1,Issue.VIII/May 2012;

Fig.2.1The data employed in the experiment comprised 500 Autonomous System points distributed uniformly over the Bipolar square [-1,1] X [-1,1]. The points thus describe a geometrically square topology

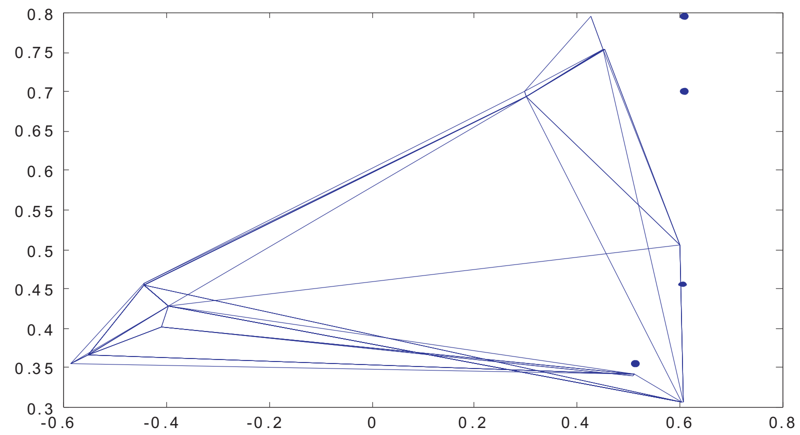


Fig. 2.2 Initial randomized state

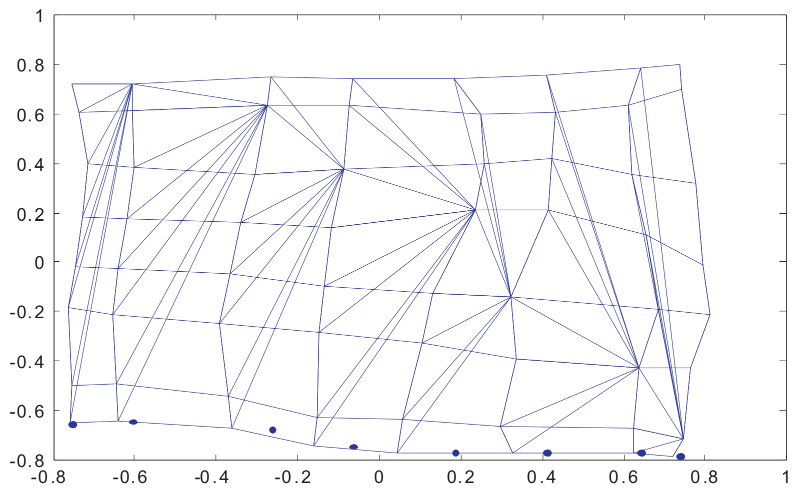


Fig.2.3 Iteration 900

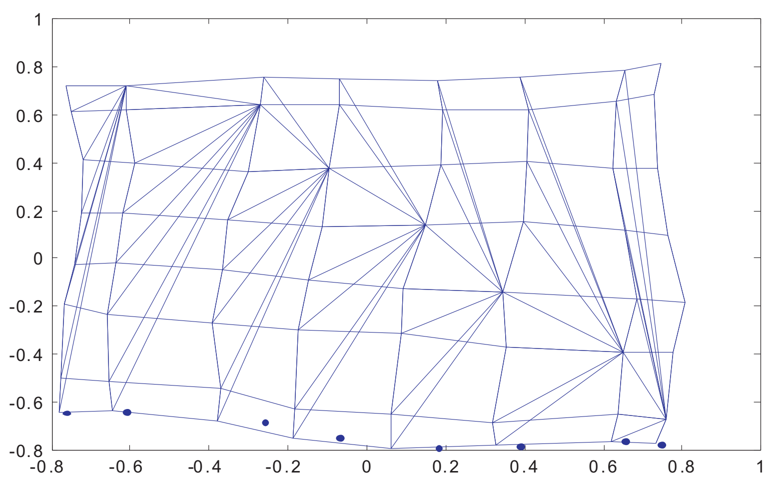


Fig.2.4 Iteration 1000
Snapshots of the simulation of the Kohonen network
Forming the Self-organizing map

REFERENCES

[1] Simon Haykin, "Neural Networks – A comprehensive foundation", Pearson Edu. Inc., Chapter 9 ; pp. 443-483, 1999.
[2] Jang, J.S.R. et al., "Neuro – Fuzzy and Soft computing – A computational approach to learning and machine intelligence", Pearson Edu. Inc., Chapter 11, pp. 327-358, 1997.
[3] Bose N.K. and Liang P., "Neural Network fundamentals with Graphs, Algorithms and Applications", Tata McGraw-Hill Inc., Chapter 9, pp. 343-406, 1998.
[4] Phil Picton, "Neural Networks", Grassroot series, Palgrave Pub. Ltd., Chapter 7, pp. 115-135, 2000.
[5] Limin Fu, "Neural Networks in computer Intelligence", Tata McGraw-Hill Inc., Chapter 2, pp. 41-59, 2003.
[6] Kumar S., "Neural Networks", Tata McGraw Hill Pub. Com. Ltd., Chapters 9,10,11,12, pp. 347-570, 2004.
[7] Dass S.K., "Neural Network and Fuzzy Logic", Shree Publishers and Distributors, Chapter 15, pp. 181-210, 2006.
[8] Rao V. and Rao H., "C++ Neural Networks and Fuzzy Logic", BPB Pub., Chapter 11, pp. 271-304, 1996.
[9] Sivanandam S.N. et al., "Implementation of Spatio-Temporal feature Map using Modified Self Organizing Map", proceedings of Int. Conf. on cognitive systems, vol. I, Dec. 13th – 15th Allied Publ. Ltd. New Delhi, pp. 22-27, 1997.
[10] Ashok Rao et al., "An efficient internet service implementation for mesh satellite networks", EP 1 192 555 Al, 2000
[11] Amakawa Koji et al., "A system for path finding", EP 1 0 62 616 Al, 1999.
[12] Lewis Kevin et.al., "Traffic Routing in a tele communications network", EP 0 852 862 Al, 1996.
[13] Hannes Schabauer et al., "Solving very large traveling salesman problems by SOM parallelization on cluster architectures", proceedings of the sixth international conference on parallel and distributed computing applications and technologies, pp 954-958, 2005, ISBN : 0-695-2405-2.
[14] Vieira F.C. et al., "An efficient approach of the SOM Algorithm to the traveling salesman problem" proceedings of the VII Brazilian symposium on Neural Networks, 2002, Published by IEEE computer society, Washington DC, USA . ISBN: 0-7695-1709-9/02.
[15] Ye, et al., " Coarse – to fine self organizing map for automatic electrofacies ordering", US patent no. 6477 469, Nov. 5, 2002.
[16] Werbas P.J. "Neural networks for intelligent control", US patent no. – 6882 992, April 2005.
[17] Rajpurohit, G.M., "BGP Instability", lecture Notes on BGP instability, http://www.cse.iitk.ac.in/users/braman/courses/cs625-fall 2003/lect-notes/lecnotes 07-1. html last updated : 2003 Date of download : 11/13/2007.
[18] "The BGP Instability Report" http://bgpupdates.potaroo.net/instability /bgpupd.html Date of access : 10/29/2007.
[19] "BGP problems and mitigation", http: en.wikipedia.org/wiki/Border _Gateway_Protocol Date of updation: Date of download: 3/28/2008.
[20] "Route flap damping" , RFC 2439 http://www.ietf.org/rfc/rfc
[21] http://www.ietf.org/internet-drafts/draft-Li-bgp-stability/01.txt
[22] Dan Pei et al., "An analysis of convergence delay in path vector routing protocols", computer networks 50, 398 –421, Elsevier B.V., 2006.
[23] Sakhare, A.D., and Khot, P.G., "Comparative Analysis of BGP Ipv4 and BGP Ipv6 Instability, presented and published in the International Conference on Statistics and applications in OR, Management / Life Sciences, In conjunction with the XXVII Annual Convention of ISPS, Jan. 10-12, 2008 held at PGTD Deptt. of Statistics, RTMNU., Nagpur.
[24]Sakhare A.D. and Khot P.G., "Neural Network based optional routing algorithm for interdomain routing", published in the research journal Shoudha Yatra vol. IV, pp. 249-258, ISBN: 81-89730-02-9.

Please cite this Article as : Anupama Sakhare and Ambika Jaiswal, Network Optimization with Shortest-Path Inter-domain Routing Algorithm Using Self Organizing Map : Review of Research (May ; 2012)