

---

**ANALYSIS OF ENERGY EFFICIENT DELAY LEAP MULTICAST ROUTING USING FEED BACK NEURAL NETWORKS**

---

**<sup>1</sup>Kaushal Kumar , <sup>2</sup>Dr. K.P. Yadav**<sup>1</sup>Research Scholar, <sup>2</sup>Research Supervisor<sup>1,2</sup>Department of Computer Science & Engineering<sup>1,2</sup>Sunrise University, Rajasthan, Rajasthan

---

**ABSTRACT:** Wireless Sensor Networks (WSNs) term can be mostly sensed as devices variety from mobile phones, laptops or PDAs to very small and simple sensing devices. Currently, the majority of available wireless sensor devices are significantly controlled in terms of computational power, memory, competence and communication abilities due to financial and technology reasons. This is the reason most of the study on WSNs has determined on the design of power and computationally proficient algorithms and protocols, and the functional domain has been restricted to simple data oriented observation and reporting uses. Wireless Sensor Network nodes are battery supplied which were implemented to carry out a specific assignment for a comprehensive period of time may be years. If WSNs nodes are more powerful or mains supplied devices in the surroundings, it is useful to utilize their computation and communication resources for complex algorithms. New network architectures with various devices and expected advances in knowledge are eliminating current limitations and expanding the range of possible applications for WSNs significantly.

**KEY WORDS:** Wireless Sensor Networks, power, memory, competence and communication abilities

---

**INTRODUCTION**

In today's world controlling the media traffic is one of the major goal facing contemporary scientists which is encouraging applications in Multimedia in such a way to ensure Quality service should be delivered to the end users. To ensure Quality service scientists are keeping in mind one end user to another end user delay leap, delay jitter leap, minimum bandwidth etc. The various minimum Steiner tree heuristics have been reported in literature [1-3]. The multicast tree, a NP complete problem, for communication network was first formulated by Kompella et al. [4]. Noronha and Tobagi [5] proposed an algorithm based on integer programming to construct the optimal source-specific leap-constrained minimum Steiner tree. The Leap shortest multicast algorithm [6, 7] was used to solve delay constrained tree optimization problem. The algorithm started by computing a least delay tree rooted at a source and spanning all group members and iteratively replaced the superedges in the tree by cheaper superedges not in the tree. The paper [8] summarized a tradeoff algorithm between the minimum Steiner tree and the least delay tree. Barathkumar and Jaffe [9] studied the algorithms to optimize the cost and delay of the routing tree. Rouskas and Baldine [10] studied the problem of constructing multicast trees subject to both delay and delay jitter constraints. Salama et al. [11] compared the performance of shortest path broadcast tree algorithm and a heuristic for tree cost.

On the other hand Hopfield Neural Network (HNN) model is a generally using for solving constrained optimization problem. The Hopfield Neural Network is a parallel, distributed information processing structure consisting of many processing elements connected via weighted connections [12]. The objective function was then expressed as quadratic energy function and the associated weights between neurons were computed using the gradient descent of energy function. The formulation of energy function associated with Hopfield Neural Network for shortest path computation was first proposed by Ali and Kamoun [13]. Hopfield Neural Network has been used for computation of shortest path for routing in computer networks and communication systems [13-15]. The Hopfield Neural Network was used to solve Quality service delivery Constrained for real time multicast routing [16-18]. The optimization of multicast tree using Hopfield Neural Network with delay and delay jitter has been reported in [18].

This paper reports a development of a Hopfield type neural network model to solve minimum

cost delay leap multicast routing problem. The delay leap optimized paths from source to various destinations are obtained recursively, which are then combined together using a union operator to ensure that all the links are participating in multicast route path only once.

**DESCRIBING DELAY LEAP MULTICAST ROUTING**

The multicast network can simply be represented as a weighted connected network like  $A = (V, W)$ , where  $V$  denotes the set of vertices (nodes) and  $W$  denotes the set of arcs (links). Let  $K$  is a subset of  $V$  i.e. forms the multicast group with each node of  $K$  is a group member. The node  $s$  is a multicast source for multicast group  $K$ . A multicast tree  $T_r$  is a sub-graph of  $A$  that spans all nodes in  $K$ , while it may include non-group member nodes along a path in the tree. The link between nodes  $i$  and  $j$ , has its properties, cost  $C_{ij}$  and delay  $D_{ij}$  as real positive values. The link cost  $C_{ij}$  may be the monetary cost incurred by the use of the network link or may be some measure of network utilization. The cost coefficients  $C_{ij}$  for the nonexistent arcs are defined as infinity. The delay  $D_{ij}$  represents the time needed to transmit information through link that includes transmission, queuing and propagation delays.

Delay leap multicast routing problem can be defined as to find a tree rooted at the source  $s$  and spanning to all the member of  $K$  such that the total cost of the links of the tree is minimum and the delay from source to each destination is not greater than the required delay constraint. Therefore, the end-to-end delay Leap multicast routing problem is defined as –

Minimize cost of multicast tree

$$Cost(T_r(p, K)) = \sum_{i,j \in T_r(p,K)} C_{ij} \tag{1.1}$$

Subjected to End-to-end delay constraint

$$D_k(P_T(p,k)) = \sum_{i,j \in P_T(p,k)} D_{ij} \leq \Delta \quad \text{for } k=1,K \tag{1.2}$$

The  $PT(p,k)$  is a path between source node  $s$  and the destination node  $k$ . The path  $PT(p,k)$  is an ordered sequence of nodes connecting from source node  $k$  to destination node  $m$ , indicating the traverse of data from ‘ $p$ ’ to ‘ $k$ ’ as-

( $p \rightarrow i \rightarrow j \rightarrow k \dots \rightarrow r \rightarrow d$ ).

**DELAY LEAP MULTICAST ROUTING WITH HOPFIELD NEURAL NETWORKS**

The delay leap multicast tree using Hopfield neural networks is obtained by recursively obtaining the delay leap shortest paths from source to each destination in the multicast group and combining them by the union operator. The union operator ensures that a link is appearing only once in the multicast tree.

**EXPLORING SHORTEST PATH PROBLEM WITH HOPFIELD NEURAL NETWORKS**

The total cost associated with path  $PT(p,k)$  is therefore expressed as –

$$C_{pk} = C_{pi} + C_{ij} + C_{jk} + \dots + C_{rm} \tag{1.3}$$

The shortest path problem is aimed to find the path  $PT(p,k)$  that has the minimum total cost  $C_{pk}$ . It can be explored using Hopfield neural network through the following procedure -

A  $(N \times N)$  matrix  $B=[B_{ij}]$  is used where  $N$  is the number of nodes. The diagonal elements of matrix  $B$  are taken as zero. The each element in the matrix is representative of a neuron described by double subscripts  $i$  and  $j$  representing the node numbers. Therefore, only  $N(N-1)$  neurons are to be computed and the neuron at the location  $(x, i)$  is characterized by its output  $B_{xi}$  defined as follows –

$$B_{xi}^k = \begin{cases} 1 & \text{if arc from node } i \text{ to node } j \text{ is in path otherwise} \\ 0 & \end{cases} \tag{1.4}$$

Let define as-

$$p_{xi} = \begin{cases} 1 & \text{if arc from node } x \text{ to node } i \text{ does not exist otherwise} \\ 0 & \end{cases} \tag{1.5}$$

The optimized path (minimum cost path) can only be obtained by minimizing constrained parameters. On minimizing the constrained parameters through an annealing schedule, the probability of visiting lower energy states increases. The energy function must favor states that correspond to valid

paths between the specified source-destination pairs. Among these, it must favor the one which has the optimized path (minimum cost). An energy function satisfying such requirement is given by [13] –

$$E^k = E_1^k + E_2^k + E_3^k + E_4^k + E_5^k \tag{1.6}$$

Such that,

$$E_1^k = \frac{W_1}{2} \sum_{x=1}^n \sum_{i=1}^n C_{xi} V_{xi}^k \tag{1.6.1}$$

$$E_2^k = \frac{W_2}{2} \sum_{x=1}^n \sum_{\substack{i=1 \\ i \neq x}}^n P_{xi} V_{xi}^k \tag{1.6.2}$$

$$E_3^m = \frac{W_3}{2} \sum_{x=1}^n \left\{ \sum_{\substack{i=1 \\ i \neq x}}^n V_{xi}^k - \sum_{\substack{i=1 \\ i \neq x}}^n V_{ix}^k \right\}^2 \tag{1.6.3}$$

$$E_4^m = \frac{W_4}{2} \sum_{i=1}^n \sum_{\substack{x=1 \\ x \neq i}}^n V_{xi}^k (1 - V_{xi}^k) \tag{1.6.4}$$

$$E_5^k = \frac{W_5}{2} (1 - V_{kp}^k) \tag{1.6.5}$$

Where,  $W_1$  weight to force the minimum cost of the path by accounting cost of existing links,  $W_2$  weight to prevent the nonexistent links being included in the chosen path,  $W_3$  weight to ensure that if a node has been entered in, it will also be exited,  $W_4$  weight to force the state of neural network to converge to one of the corner of the hypercube defined by  $V_{xi}^k$ ,  $W_5$  weight to enforce the construction of path originating at source p and terminate at destination k.

### UNION OF SHORTEST PATHS

The  $V_k$  obtained above represents the output matrix of unicast route or shortest path from source p to destination k. Final output of multicast tree is obtained by the union of unicast routes from source to various destinations. The element  $V_{xi}$  in multicast tree is obtained as –

$$V_{xi} = V_{xi}^1 \cup V_{xi}^2 \cup V_{xi}^3 \cup \dots \cup V_{xi}^k \tag{1.7}$$

#### 1.4.3 Evaluating Delay Leap in Shortest Path with Hopfield Neural Networks

We are here introducing a new energy term, where one end to another end delay constraint is referred as penalty, can be added for delay leap optimized path. This energy function term is therefore expressed as –

$$E_6^k = \frac{W_6}{2} \sum_{x=1}^n \sum_{\substack{i=1 \\ i \neq x \\ (x,i) \neq (k,p)}}^n DL_{xi}^k V_{xi}^k \leq \Delta \tag{1.8}$$

Where,  $W_6$  weight to enforce that the delay of the constructed path is less than or equal to specified delay constant.

The total energy function for kth destination including delay leap is therefore defined as-

$$E^m = E_1^m + E_2^m + E_3^m + E_4^m + E_5^m + E_6^m \tag{1.9}$$

Using eqn (1.6) and (1.9),

$$E^m = \frac{W_1}{2} \sum_{x=1}^n \sum_{\substack{i=1 \\ i \neq x}}^n C_{xi} V_{xi}^k + \frac{W_2}{2} \sum_{x=1}^n \sum_{\substack{i=1 \\ i \neq x}}^n P_{xi} V_{xi}^k + \frac{W_3}{2} \sum_{x=1}^n \left\{ \sum_{\substack{i=1 \\ i \neq x}}^n V_{xi}^k - \sum_{\substack{i=1 \\ i \neq x}}^n V_{ix}^k \right\}^2 + \frac{W_4}{2} \sum_{i=1}^n \sum_{\substack{x=1 \\ x \neq i}}^n V_{xi}^k (1 - V_{xi}^k)$$

$$+ \frac{W_5}{2}(1 - V_{kp}^k) + \frac{W_6}{2} \sum_{x=1}^n \sum_{\substack{i=1 \\ i \neq x \\ (x,i) \neq (k,p)}}^n DL_{xi}^k V_{xi}^k \tag{1.10}$$

**STOCHASTIC APPROACH FOR OPTIMIZATION IN DELAY LEAP IN SHORTEST PATH**

To explore the optimized Energy function depicted in eq. (1.10), the Hopfield net with the probabilistic update rule can be used. The probabilistic distribution of states will be stationary or independent of time for a network to be in stochastic equilibrium. On decreasing the constraint parameter according to the probabilistic annealing schedule the network produces a new energy landscape, which contains the minimum energy states with respect to the previous one. This process continues until the network reaches to the global minimum energy state. This state refers to optimized delay leap in shortest path.

The shortest path with delay leap has been given by the eq. (1.10). The probability distribution of this states can be given as

$$P(s) = \frac{1}{z} e^{-\frac{E^k(p)}{\xi_v}} \tag{1.11}$$

Where, z represents the partition function, P(p) represents the probability of visiting state p and Ek(p) represents the energy function of state p.

According to this probability distribution the probability of visiting the lower energy states decreases as the network approaches to higher energy states and it also shows that for the larger value of constraint parameter, the probability of visiting the lower energy states decreases. Now, as the constraint parameter is reduced as per the annealing schedule of mean field approximation, the probability of visiting the lower energy states increases. Finally at the allowable lower limit of constraint parameter i.e., the probability of visiting the lower energy states approaches to 1 (i.e. highest probability), so that the network settles in the minimum energy state, which describes the optimized delay leap shortest path amongst the multicast routing. Since, the implementation of simulated annealing requires computation of stationary probabilities at equilibrium state for each annealing schedule. To speed up this process, we may use mean field approximation [16] in which the stochastic update of bipolar units is replaced with deterministic states. The basic idea of mean field approximation is to replace the fluctuating activation values of each unit by their average values [17]. Activation of a unit can be given as

$$\langle x_i \rangle = \left\langle \sum_i W_{ij} S_i \right\rangle, \tag{1.12}$$

or,  $\langle x_i \rangle = \sum_i W_{ij} \langle S_i \rangle$

where,  $\langle x_i \rangle$  is the average of the states of node and Wij represents the weight with ith and jth node.

$$\text{and } \langle S_i \rangle = \tanh\left(\frac{x_i}{\xi_v}\right) \tag{1.13}$$

In mean field approximation the activation of unit is replaced by  $\langle S_i \rangle$ , so that using equation (1.14), we have

$$\langle S_i \rangle = \tanh\left(\frac{1}{\xi_v} \sum_j W_{ij} \langle S_j \rangle\right) \tag{1.14}$$

The set of these equations is a result of minimization of an effective energy defined as a function of constraint parameter [18]. Thus, eq. (1.14) may be expressed as

$$\langle S_i \rangle = \tanh\left[-\frac{1}{\xi_v} \frac{\partial E^m(\langle S \rangle)}{\partial \langle S_i \rangle}\right] \tag{1.15}$$

Where the change in energy function for the average states of unit is given by

$$\frac{\partial E^m(\langle S \rangle)}{\partial(\langle S_i \rangle)} = -\sum_{i \neq j} W_{ij} \langle S_j \rangle \quad (1.16)$$

The non-linear deterministic equations given by eq. (1.15) are solved iteratively. As the constraint parameter is lowered to the minimum value, the steady equilibrium values of will be obtained. At the allowable lower limit of constraint parameter the probability of visiting the minimum energy states has

the maximum value i.e.  $P(p) = \frac{1}{z} e^{-\frac{E^k(p)}{z}} \approx 1$ , so that the network will achieve one of the minima of energy landscape. At this global minimum of the energy landscape the average value of the state of the network will represent the optimized energy function for delay leap optimized path in multicast routing.

$$\text{i.e. } \langle S_i \rangle = kD_{op}, \quad (1.17)$$

Where m is the proportionality constant and Dop is the delay leap optimized path.

### 1.6 Conclusion

Hopfield neural network model with stochastic annealing is hereby successfully applied to obtain minimum cost delay leap constrained parameter for multicast tree. The multicast tree is obtained by recursively obtaining the delay leap optimized path from source to various destinations and combining them by union operator. The union operator ensures that a link is appearing only once in the multicast tree. The minimum energy function is obtained with minimization of constrained parameter as per a defined annealing schedule, which increases the probability of visiting lower energy states. Finally, the goal of minimization of objective function (minimum cost delay leap route) is achieved by using mean field approximation with stochastic annealing process of reducing constrained parameter. It is suggested here, that in future, the effectiveness of the developed scheme should be tested to obtain optimum multicast trees for different sets of source and destinations on any weighted connected network.

### REFERENCES

1. L. Kou, G. Markowsky, and L. Berman. A fast algorithm for Steiner trees. *Acta Informatica*, vol. 15, no. 2, pp. 141-145, 1981.
2. H. Takahashi and A. Matsuyama. An approximate solution for the Steiner problem in graphs. *Mathematica Japonica*, vol. 24, no. 6, pp. 573-577, 1980.
3. V. R. Smith. The computation of nearly minimal Steiner trees in graphs. *International Journal of Mathematical Education in Science and Technology*, vol. 14, no. 1, pp. 15-23, 1984.
4. V. Kompella, J. Pasquale, and G. Polyzos. Multicast routing for multimedia communication. *IEEE/ACM Transactions on Networking*, vol. 1, no. 3, pp. 286-292, 1994.
5. C. Noronha and F. Tobagi. Optimum routing of multicast streams. *Proceedings of IEEE INFOCOM '94*, pp. 865-873, 1991.
6. Q. Zhu, M. Parsa, and J. G. L. Aceves. A source-based algorithm for delay-constrained minimum-cost multicasting. *Proceedings of IEEE INFOCOM '95*, pp. 377-385, 1996.
7. Pandey A., Bansal K.K. "Performance Evaluation of TORA Protocol Using Random Waypoint Mobility Model" *International Journal of Education and Science Research Review* Vol.1 (2) pp 193-199
8. Yadav R.K., Bansal K.K. "Analysis of Sliding Window Protocol for Connected Node" *International Journal of Soft Computing and Engineering (IJSCE)* Vol.2 (5) PP. 292-294
9. M. Parsa and Q. Zhu. An iterative algorithm for delay-constrained minimum-cost multicasting. *IEEE/ACM Transactions on Networking*, vol. 6, no. 4, pp. 461 – 474, 1998.
10. L. Wei and D. Estrin. The trade-offs of multicast trees and algorithms. *Proceedings of the Third International Conference on Computer Communications and Networking (IC 3N '94)*, pp. 17-24, 1991.
11. K. B. Kumar and J. Jaffe. Routing to multiple destinations in computer networks. *IEEE Transactions on Communications*, vol. 31, no. 3, pp. 343-351, 1984.
12. G. N. Rouskas and I. Baldine. Multicast routing with end-to-end delay and delay variation constraints. *Proceedings of IEEE INFOCOM '96*, pp. 353-360, 1996.
13. H. F. Salama, D. S. Reeves and Y. Viniotis. Evaluation of multicast routing algorithms for real-time communication on high-speed networks. *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 3, pp. 332 – 345, 1997.
14. J. J. Hopfield and D. W. Tank. Neural computation of decisions in optimization problems. *Biological Cybernetics*, vol. 52, pp 141-152, 1986.
15. M. K. Ali and F. Kamoun. Neural networks for shortest path computation and routing in computer networks. *IEEE*

- Transactions on Neural Networks, vol. 4, no. 6, pp 941-953, 1994.
16. H. E. Rauch and T. Winarske. Neural network implementation of the shortest path algorithm for routing communication traffic. IEEE Computer System Magazine, pp. 26-30, 1988.
  17. L. Zhang and S. C. A. Thomopoulos. Neural network implementation of the shortest path algorithm for traffic routing in communication network. Proceedings of International Conference on Neural Networks, 1999.
  18. S. Kirkpatrick, C. D. gelatt and M. P. Vecchi. Science. Vol. 220-671, 1984.
  19. G. L. Bilbro, W. E. Snyder, S. J. Garnier and J. W. Gault. IEEE Trans. Neural Networks. 3 no.1- 131, 1993.
  20. S. Haykin. Neural Networks: A Comprehensive Foundation. New York, Macmillan College. 338, 1991.