7th WSEAS Int. Conf. on MATHEMATICAL METHODS and COMPUTATIONAL TECHNIQUES IN ELECTRICAL ENGINEERING, Sofia, 27-29/10/05 (pp129-133)

Multicast routing algorithm based on Extended Simulated Annealing Algorithm

Jin-Ku Jeong*, Sung-Ok Kim**, Chi-Hwa Song***
* **Dept. of Computer Eng, Hannam University
133 Ojeong-dong Daedeok-gu, Daejeon 306-791, Korea
***Dept. of Computer Science Chungnam National University
220 Gung-dong, Yuseong-gu, Daejeon, 305-764 Korea

Abstract: - In this paper, we propose a method that is able to find a good multicast routing tree in a wide area network. The problem domain that we should solve is modeled as a weighted, undirected graph. The graph that represents a wide area network has three node type, a server that sends the same data to the recipients, some nodes that do not request data to the server and the rest that don't need data. Our method uses the extended simulated annealing algorithm[ESA].

Keyword : Extended simulated annealing, multicast routing, optimization

1. Introduction

We propose an algorithm that searches route for sending data from one server to specific multiple clients to minimize the total distance in networks. This problem is different from the problem of finding a route for all nodes in networks. The complexity of finding route tree is increasing dramatically when more nodes are coming into action therefore, it is hard to find the best solution (route) with traditional algorithm.

Recently, the routing algorithms for multicasting are being studied widely, some of whitch are as follows. Simulated Annealing method[1], Applying SA on Dijkstra for multi-constraint routing tree[2], routing algorithm for real-time computing using Hopfield Neural Computing[3,7] and distributed algorithm[4]. The simulated annealing method for determining route uses end-to-end delay and error-rate with the consideration of network resources. The distributed algorithm determines by the grouping of the neighboring nodes that take their part in multicasting.

Our proposed algorithm uses ESA algorithm[5,6] to minimize the sum of the distance cost between the nodes. We have mapped network domain into the form of the graph and defined perturbation scheme for ESA algorithm with evaluation formula to find best multicast tree. The cost function of the proposed algorithm considers distance cost and node cost, but it does not considers the number of the buffer of router, average traffic quantity and transfer delay. However, these volumes can be changed into figures.

2. ESA Algorithm

SA algorithm extracts a possible solution set by the metropolis sampling scheme and the thermodynamic average is calculated with canonical average. Therefore, the possibility of energy E with the status of i can be described as equation 1 where, according to statistical dynamics figure such as fixed volume (V), fixed number of particles (N) and under the constant temperature (T) with in the closed system

$$P_i(N, V, T) = \frac{\exp[-E_i(N, V)/kT]}{\Sigma_i(\exp[-E_i(N, V)/kT])} \qquad (1)$$

This is the representation for the closed system. Traveling Salesman Problem(TSP) is representative problem that a solution could be found with the usage of the SA. TSP has a fixed number of the city to visit and the salesman must find the least cost route. We will consider the visiting city as the number of molecules(N) here then it will be canonical ensemble.

When, the molecules fluctuatoin each unit system is called grand canonical ensemble. Volume(V), temperature(T) and chemical potential (μ) is fixed while energy(E) and the number of molecules(N) is opened. The probability of state i with energy E and N molecules will be in equation 2.

$$P_i(N;\mu, V, T) = \frac{\exp[-(E_i(N, V) - \mu N)/kT]}{\Sigma_N \Sigma_i(\exp[-(E_i(N, V) - \mu N)/kT])} \qquad (2)$$

Therefore, the highest probability of state i appearance is in concern with energy $E(N, V)$ and number of molecules. The study of open system that not only allows energy but also the number of particles is based on grand canonical ensemble.

For example, one salesman has to visit every city (N) with the least cost and the least distance route which is the problem under canonical ensemble and the salesman has to visit with the limitation of the cost has to visit cities (n, $0 \leq n \leq N$) this condition it is grand canonical ensemble.

3. Definition of a problem for ESA

3.1 representation of network

In a huge network one server must cast the same data to multiple clients so it has to decide the best rout for every node N or the some node ($n \leq N$). To find multicasting route we represent network in graph $G=(V,E)$. Every node in network including data server is represented as vertices $V=\{v_0,v_1,\cdots v_{n-1}\}$. The physical connection of each node is represented as edges $E=\{e_0,e_1,\cdots e_{m-1}\}$. Each edge is connected to two other nodes as $e_i = uv$.

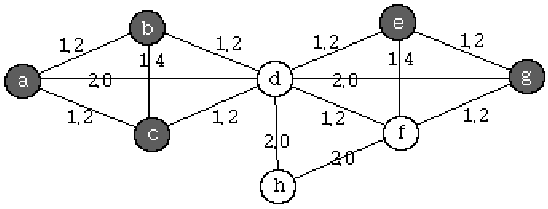


Figure 1. Graphic Representation of Network

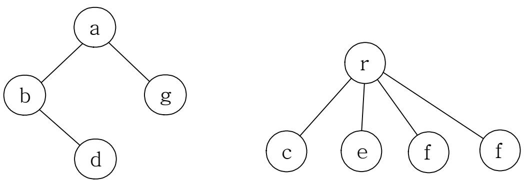
To represent figure 1 as formula will be as equation 3.

$$
\begin{aligned}
&G=(V,\ E)\\
&\quad V=\{\ a,b,c,d,e,f,g\ \}\\
&\quad E=\{\ ab,ac,ad,bc,bd,\ cd,de,df,dg,ef,eg,fg\ \}
\end{aligned}
\tag{3}
$$

To select the best route it is not necessary to divide into server and client but for the ease of the thinking let us consider first node a as a server. Node d,f,h is a node that did not requested for service but in the whole transfer route tree to connect to the neighboring node it could be inserted to transfer route. When node d is not included in transferring route it is not able to receive node e and node g, therefore, it must be in the route.

3.2 Representation of transferring route

To solve the problem like figure 1 in G select some of the node and edge and do the node a (server) as a root and routing tree. To set the transfer route for every node in network system it is general to use spanning tree algorithm. Set some of the node for transfer route.



T = a(b(d()),g()) R = r(c(),e(),f(),h())

Figure 2. Transfer route tree

The tree T in figure 2 represents transfer route and R represents the nodes that is not included in routing tree. The root node r in R is connected to the siblings that is excluded in T. The sum of node in T and R is always the same as the number of nodes in G.

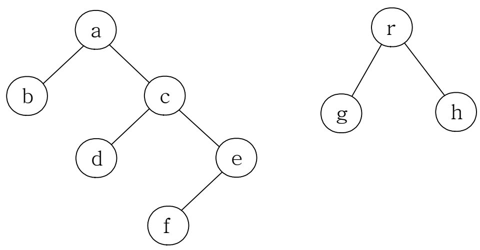
3.3 perturbation scheme

In graph G arbitrarily selects one transfer route tree T. We would use T as an initial solution for ESA algorithm and determine whether it is adequateness by evaluation function that we proposed. We use the evaluation function to verify T. Until we find better solution T' the system perturb. The series of perturbed solution have to form Markov chain so the methods can be described as following 3 types.

(1) Method of removal one node
(2) Method of no change in number of node
(3) Method of adding one node

In the method of (1) we could remove any node except root node in tree T. The removed node is reinserted in virtual tree R. In method of (2) there is two ways to change. First, move one node in tree T to the other location. Second, interchange one node in T and one node in R. In the method of (3) select one node arbitrary in R and insert node to T.

Let's consider the tree T before state transition as figure 3. At this point a,b,c,d,e,f is node of the T, and g,h is the node of external tree R.



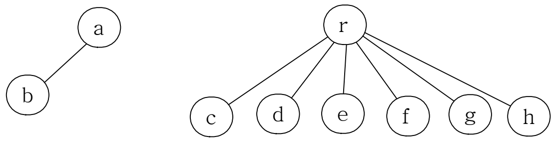
T=a(b()c(d()e(f()))) R=r(g()h())

Figure 3. Initial tree T

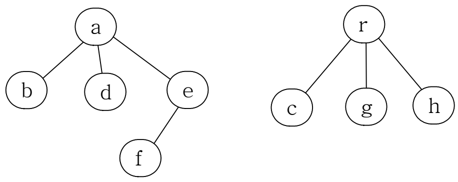
To remove one node in T, we consider two types one is to delete leaf node and the other is to delete a node of the middle one. In T to remove middle node like c,e we must consider where to send child node's location. For example, in T when node c is deleted the siblings location will be determined by the deletion of itself when the deletion of the parent is taking place (figure 4-a) or the connect siblings to the parent of the parent like in figure 4-b.

When the deletion of every node in sub-tree the energy change is so rapid depending on the location of the node and according to the specification of its nature it is hard to secure optimal solution. And in the condition of figure 4-b the root node a is connected to the node d,e

via c. In this situation without consideration of deletion of node c the root a must send its data directly to d,e and node f is receiving its data from node e without consideration of deletion of node c.



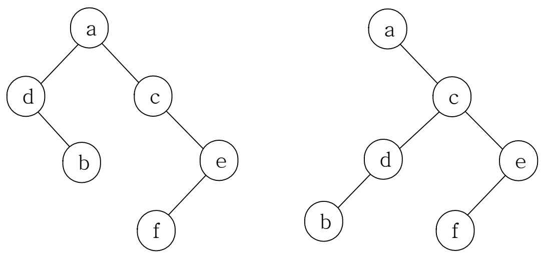
(a) deletion of total subtree



(b) delete a node after moving

Figure 4. Deletion of the node

To produce T' without change of the number of the node it extracts one arbitrary node and change its location and gains new transfer route in tree T. Figure 5, figure 6 and figure 7 show the move of the node. There are two cases of moving node; figure 5 represents the move of the leaf node, and figure 6 and 7 represent the move of the middle node. The condition of change of the middle node is categorized by where to move its child node. This condition is represented in figure 6 and figure 7.



(a) (b)

(a) move leaf node to nonleaf node

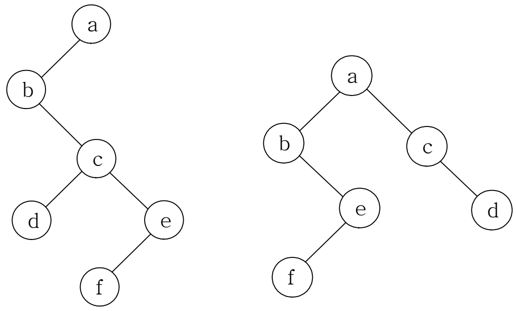
(b) move leaf node to leaf node

Figure 5. Move of the leaf node

It is meaning less that the sibling nodes left or right. Figure 6-b represents move of the leaf node to leaf node here and node b is connected directly to the root, but in T' the depth of node b is increased therefore, the data transfer route is a→c→d→b whereas the data must travel two more node to get access to the root. If there node b does not have connection edge in $G=\{a,b\}\not\subset E$ and in new route there exists edge where $G=\{ac,cb,bd\}\subset E$ the transfer cost of T' is smaller the T''s cost.

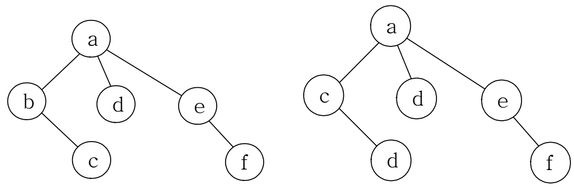
In figure 7-a it illustrates move of the non leaf node to leaf node. And figure 7-b illustrates non leaf node to non leaf node.

The final state transition method is to add one more node to tree T. In virtual tree R selection of arbitrary node but for the root node and insert into tree for node it is illustrated in figure 8.



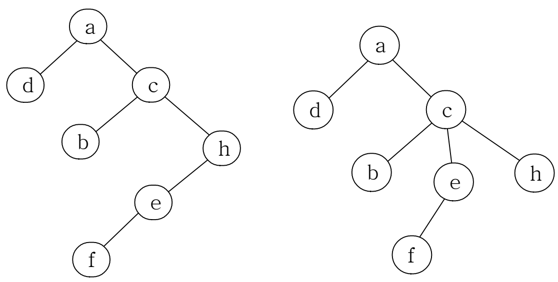
(a) move c to b (b) move e to b

Figure 6. Move of the sub-tree



(a) leaf node moving (b) non-leaf node moving

Figure 7. Move of the node



(a) insert as a leaf (b) insert as a non leaf

Figure 8. Add of the node

3.4 Cost function

In figure 1 each edge has its own weight. Weight represents distance between the nodes. The transfer cost didn't consider network delay or processing time of the router or bandwidth. We derived tree T from figure 1 as follows.

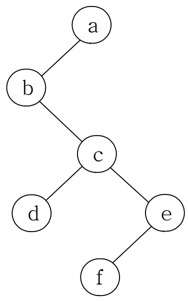


Figure 9. Transfer tree

Transfer cost of this state will be the sum of the edge's weight of $E_t = \{ab, bc, cd, de, ef\}$. Therefore, it will be $1.2+1.4+1.2+ce+1.4$. In the original graph G there is no edge ce where $E(\{ce\} \not\subset E)$. Therefore, to calculate total energy cost we must decide the cost of edge like ce and it must be greater than any other values of edge represented in G. So we use the value of edge that is not exist as

$$e_i = \alpha * MAX(e_j), \quad e_j \not\in E, \quad e_i \not\in E \quad \alpha \geq 1.0$$

In the condition of figure 1 the maximum edge value is 2.0. When variable $\alpha = 1.5$ the edge value that is not represented in G is 3.0 therefore, the transfer tree in figure 10 will be 8.2

According to figure 1 the node v, c, e, g must receive the data from the server. But transfer tree in figure 10 or figure 11 node g is neglected from transfer tree. Therefore, the minimum cost with the same node like figure 11 it will be a solution that cannot be accepted.

In the process to calculate optimized transfer tree the nodes can be categorized into 4 categories.

Case 1: It must be provided with service and that relies with in the transfer route.

Case 2: It must be provided with service but it is in virtual tree R.

Case 3: It does not need to be serviced but within the route of transfer.

Case 4: It does not need to be serviced and it is not in the route of transfer.

In the proposed algorithm we used boolean type index to the node to know whether a node is need to connect the server. The cost function is represented in the formula as follows

$$Cost = distance + \alpha * nodecost \tag{4}$$

$$distance = \sum_{i=1}^{N} \sum_{j=1}^{N} (a * e_{ij})$$

$$a = \begin{cases} 1 & \text{if node i and j are in the path tree} \\ 0 & \text{orthers} \end{cases}$$

$$N = \text{number of node in the graph}$$

$$nodecost = \sum_{i=1}^{N} n_i$$

$$n_i = \begin{cases} -b/2 * Max(e_{ij}) & \text{case 1 type nodes} \\ b * Max(e_{ij}) & \text{case 2 type nodes} \\ 1 & \text{case 3 type nodes} \\ 0 & \text{case 4 type nodes} \end{cases}$$

3.5 Algorithm

To do the convergence to the optimal value we use three state transition functions. We select a value randomly among 0,1,2 when 0 is selected the function $perturb(|T|, |T|-1)$ will be used and. In the condition of 2 the function $perturb(|T|, |T|)$ is used and In the condition of 3 function $perturb(|T|, |T|+1)$ is used. These functions are state transition functions and the first function is used at reduction state transition, and the following is without the change of the transfer node but change its location and the latter is for the change with inserting external node in. We derive new transfer tree T' from existing transfer tree T. The transfer cost contain the gain of node(The gain value of each node in Grand canonical ensemble represents the chemical potential μ). We determine with the change of the cost whether the new derived transfer route tree is accepted or not. These processes are conducted until the temperature is stabilized.

When it comes into a stabile situation in given temperature we reduce temperature a little bit and redo the sampling. When the temperature is high the system accept the most state that is perturbed but when the temperature is close to 0.0 it starts to accept the lower energy state only with none zero possibility. The system will evoke the result of the present transfer route tree until it comes to low temperature ($temperature \cong 0.0$) that is not 0.0 possibilities.

Initial temperature and α are variable. T and T' represents transfer route tree and $Cost$ and $Cost'$ represents the cost of each T and T'. The system uses T and $Cost$ to get a new state that is temporal transfer tree T' and transfer cost $Cost'$. If the new solutions have less energy it will be the new solution. When $Cost'$ is bigger than that of $Cost$ with the possibility of $\exp(-(\Delta Cost')/temperature)$ T' and $Cost'$ will be replaced to T and $Cost$.

Decide initial parameter values :
- T : get a random path tree as a initial solution,
- $temperature$ = initial temperature,
- $Cost = calculate\ Cost\ according\ to\ T$

Step 1 :
- select a value within 0,1,2 randomly
  - case 1 : $T' = perturb(|T|, |T|-1)$
  - case 2 : $T' = perturb(|T|, |T|)$
  - case 3 : $T' = perturb(|T|, |T|+1)$

Step 2 :
- $Cost' = calculate\ Cost\ according\ to\ T'$
- $\Delta Cost = Cost - Cost'$

Step 3 :
- $P(Cost') = \exp(-(\Delta Cost')/temperature)$
- if $\Delta Cost < 0$ OR $P(Cost')$ then
  - $T = T'$, $Cost = Cost'$
- if (state is in equilibrium) then
  - decrease temperature

Repeat step1~step3 Untile temperature $\cong 0.0$

Figure 10. ESA Algorithm

4. Experiment and results

In figure 11 it represents the data that was used in the experiment. The circle represents network node and the numbers inside of the circle stands for its indexes of the node. The index does not effects order or the size. The connection between the nodes represents physically connected node and the weight of edge represents distance and for the experimental it is used for data transfer cost. Number 0 node is server and the others are clients.

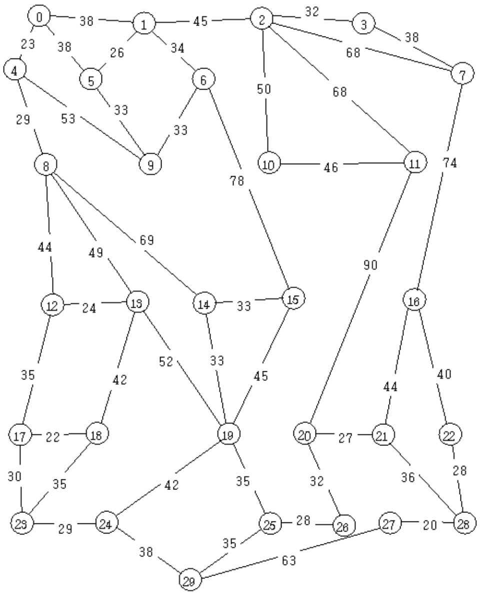


Figure 11. Experimental Data

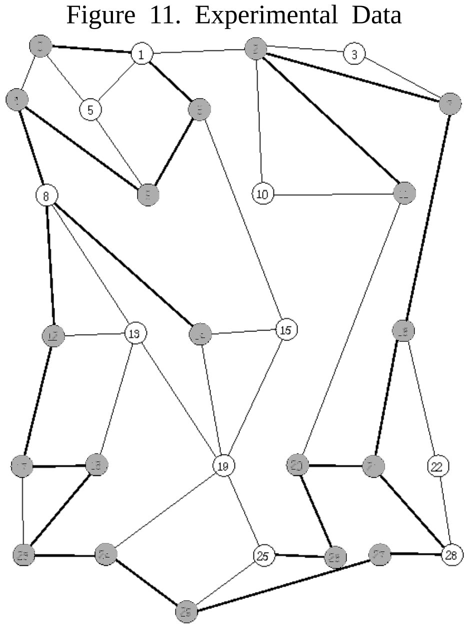


Figure 12. Final conclusion of network route

Figure 12 is the final output. In figure 12 the nodes that wants to be serviced has a color of gray and the other that has white circle nodes will be the nodes that is not being serviced. Node 1, 8, 28 are presented in transfer routing tree so that relays the data to descendant nodes.

Figure 13 illustrates the change of transfer cost during processing time. The temperature started from 1000000.0 and cooled down to 0.01 and the initial transfer cost is 2750.0 and the final transfer cost is -8502.0.

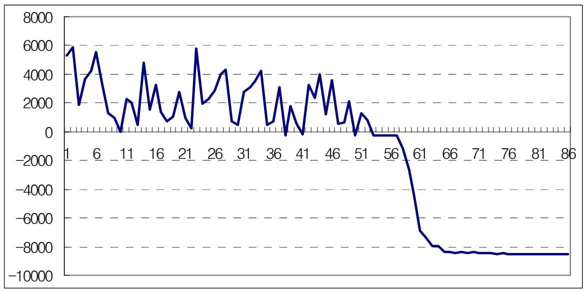


Figure 13. The cost of each temperature level

5. Conclusion

As in equation 4 it configures node cost with α and it prevented the type case 3 nodes from the final route. When the value of α is too big or too small we could not get a good solution because it disturb transition to a positive state because it breaks the balance of node cost and distance cost. These are general problem of Simulated Annealing Algorithm.

In this experiment we used the grand canonical ensemble for ESA algorithm. And we showed transfer route problem for every node and selection of arbitrary node and could be broodedto the finding of the optimized transfer route.

References :

[1] Xingwei Wang, Hui Cheng, Jiannong Cao, Lianwei Zheng, Ming Huang, “A Simulated-annealing-based QoS Multicasting Algorithm”, Proceedings of ICCT2003

[2] Yong Cui, Ke Xu, Jianping Wu, Zhongchao Yu, Youjian Zhao, “Multi-constrained Routing Based on Simulated Annealing”, iEEE 2003

[3] Chotipat Pornavalai, Goutam Chakraborty, Norio Shiratori, “A Neural Network Approach to Multicast Routing in Real-Time Communication Networks”, 1995,IEEE

[4] Fred Bauer, Anujan Varma, “Distributed Algorithms for Multicast Path Setup in Data Networks”, IEEE/ACM Transaction on Networking, Vol. 4, No. 2, April 1996.

[5] C.H.Song, K.H.Lee, W.D.Lee, “An Algorithm for Augmented Multiple Vehicle Routing Problem”, WSEAS TRANSACTIONS ON SYSTEMS Issue 4, Vol. 2, pp 896-899, Oct. 2003

[6] Jong Suk Choi, Won Don Lee, Sukhoon Lee, Nae Hyun Park, Sang Il Hwang, Yeo Duk Youn, "Extended Simulated Annealing based on Grand Canonical Ensemble", Journal of the Korea Information Science Society, Vol 17, No. 4, July 1990

[7] Pallapa Venkataram, Sudip Ghosal, B.P. Vijay Kumar, "Neural network based optimal routing algorithm for communication networks", Neural Networks 15, 2002, pp1289-1298