Presenting a New Method to Solve QoS Multicast Routing Based on Harmony Search Algorithm

A. Peyman Rashidian, B. Mohammad Kheirandish, C. Mohammad Mosleh

Computer Engineering Group, Islamic Azad University, Dezful Branch, Dezful,Iran
p.rashidian@yahoo.com
Computer Engineering Group, Islamic Azad University, Dezful Branch, Dezful,Iran
kheirandish@iaud.ac.ir
Computer Engineering Group, Islamic Azad University, Dezful Branch, Dezful,Iran
mosleh@iaud.ac.ir

Abstract

The multicast routing problem by regarding the criterions quality of service is included as a nonlinear combinational optimization problem, and its goal is finding a least cost multicast routing tree, while reaching quality of service constraints, such as delay, bandwidth and etc. One of the most popular methods for solving this problem is finding the constrained Steiner tree in a network. Finding such tree in the network is one of NP-Complete problems. For this reason, many approximation methods using Meta-heuristic algorithms have been reported for solving it. In this paper we present a novel method based on harmony search algorithm which can create the optimized routing tree with a proper speed. The result of performed simulations show that the proposed algorithm don't be caught in local optimal and compared with genetic algorithm, particle swarm optimization algorithm and bees algorithm, has a proper performance, regarding to convergence time and the quality of generated solutions.

Keywords: Multicast Routing, Constrained Steiner Tree, Quality of Service, Harmony Search Algorithms

I. Introduction

Most of network applications in the past, had been unicast applications in which the quality of service (QoS) wasn't considered, significantly. For this reason, the routing algorithms were very simple and elementary. But today, by expanding the technology in real-time multimedia applications such as video conference, virtual universities, on web multimedia games and etc. this format won't be responsible; because in these applications, data’s are received by more than one user in which each of them has different QoS requirements, in various times. So, the resources must be managed in a manner that the QoS requirements can be reached for each user. Thus it can be said the key for solving this problem is having a proper scheme and method for routing while considering QoS criterions. Supporting a resource to communicate with several destinations requires applying and developing the multicast routing in the network. Regarding this fact that multicast algorithms have tree structures, it should be possible to find a tree with minimum cost in the network, such that it can connect a source node to all destination nodes, and

Simultaneously, reach the intended constraints of QoS. Finding such tree in the network has been known as constrained Steiner tree problem and is included as NP-Complete problems (Karp, 1972).

Hitherto, several innovative methods have been presented for solving this problem, among which the Kompella–Pasquale–Polyzos KPP heuristic (Kompella et al., 1993) and the bounded shortest multicast algorithm BSMA heuristic (Qing et al., 1995) can be mentioned. These two algorithms have solved this problem by regarding the delay, as a QoS parameter. But the simulation results performed by Salama (1997) showed that most innovative algorithms either perform very slowly or are not able to compute the minimal multicast tree with mentioned constraint. Also, it must be noted that the above algorithms have been designed for some special applications and have taken to consideration one constraint only, without can be generalized for real time applications, with more QoS constraints.

So, using meta-heuristic algorithms inspired from the nature appeared to be more proper option for solving such these complex problems. As the advantages of using these algorithms, their wide application in solving diverse problems, high flexibility and easy implementation can be indexed. Although these algorithms don't guarantee the reaching to global optimal solutions, but in most cases, they can generate proper approximated solutions. For example, Haghighat et al. (2004) have tried to solving multicast routing problem while considering the constraints the delay and the bandwidth, by using genetic algorithm. They presented two novel methods for crossover operator, and by investigating different combinations of genetic algorithm operators and some kinds of coding usable in it, they presented a method that could generate appropriate multicast tree in most cases. But by increasing the size of network, sometime while computing global optimal solution, this method had been caught in local optima. Wang et al. (2004), presented a method named TSDLMRA that tries to solving Qos multicast routing problem with considering the delay as a constraint, by using Tabu search algorithm, In their proposed method, a set of supply paths were created by using the K-th shortest path algorithm and selecting the neighborhood for each destination node were done among these paths. The low time complexity, O (Kmn^3), was one of strength points of this algorithm. Wang et al. (2007) have tried to solving the mentioned problem with considering the delay and bandwidth, by using the particle swarm optimization algorithm. In their proposed algorithm, each particle was introduced as a solution for the problem and the particles, by knowing their current optimal position and the global optimal position, try to change their position and search the whole solution space. The advantage of their algorithm is an appropriate speed and relatively appropriate quality in term of generated solutions. TayebTaher and Rahmani (2012) have solved the problem by using Bee algorithm, while considering two constraints the delay and the bandwidth. They performed neighbor searching based on grading the space regions and designating the bees to these regions. Also, they presented a novel fitness function which could evaluate the solutions qualities, very well. There proposed method also had a proper performance in escaping from local minima. But disadvantage of their proposed method was slow converging. Kun et al. (2005), have tried to solving the distributed multicast routing problem with considering the delay and delay variation, as constraints, by using the simulated annealing algorithm. Their proposed method had two phases. In the first one, an initial solution was generated, only based on QoS parameters, without regarding the cost. Then, in the second phase, the algorithm tried to reduce the cost of the tree in an iterative manner, while don't violating the constraints. As advantages of this algorithm, we can refer to using the flexible strategies to make sure about the fast convergence and easiness of coding operations. Forsati et al. (2008), tried to solve the Qos multicast routing problem by using harmony search algorithm. The QoS constraints which were used by them for solving the problem included bandwidth and delay, and solving the problem has been done via creating the multicast Steiner tree. They used a modified version of prüfer number for coding the Steiner tree. Then they presented a coding structure called NPI which covers the weaknesses of previous method. Their proposed algorithm outperformed the genetic algorithm and BSMA, in term of solutions qualities, and had a relatively proper execution time, too. As another methods which used meta-heuristic algorithms for solving the Qos multicast routing problem such as, tabu search algorithms (Ghaboosi and Haghighat, 2007; Armaghan and Haghighat, 2009), genetic algorithms (Sun and Li, 2006; Koyama et al., 2008), Ant colony optimization algorithms (Gong et al., 2007; Wang et al., 2009), particle swarm optimization algorithms (Liu et al., 2006; Sun et al., 2011) and so on could be mentioned.

In this paper, we present a novel method to solving the QoS multicast routing problem by using harmony search algorithm. This method, by creating a proper combination of execution time and the quality of solutions, can find the optimal multicast tree in the network, during an acceptable time and without trapping in local optima.

II. Describing and formulating the problem

A communication network, usually, is depicted as a non-directional graph, $G=<V, E>$ in which, V is the set of all nodes (either routers or switches) and E denotes the set of all edges which represent the physical or logical links between nodes. Three weights $B(e): E \rightarrow R^{+}$, $D(e): E \rightarrow R^{+}$ and $C(e): E \rightarrow R^{+}$ are assigned to each edge $e \in E$ in G. All of these weights are positive real numbers and represent the cost, delay and available bandwidth parameters, respectively. A multicast tree $T(s,M)$ is a noncircular sub-graph of graph G which covers the source node $s \in V$ and a set of destination nodes, $M \subseteq V - \{s\}$. The number of destination nodes in the multicast tree is denoted by $m = |M|$. Also, $T(s,M)$ can include some other nodes which are members of the multicast tree but are out of multicast group. These nodes are known as Steiner points. $P_T(s,d)$ denotes an unique path in the tree, T, which connects the source node, s to a destination node, d, in the multicast tree. The intended parameters related to the cost and QoS are defined as follow:

- **Cost :** the cost of multicast tree $T(s,M)$ is the sum of the costs of all edges in the tree, and is computed as follows:

$$C(T(s,d)) = \sum_{e \in E_T} C(e) \tag{1}$$

- **Delay :** the total delay of path $P_T(s,d)$ is simply the sum of all edges delay in the path $P_T(s,d)$ and is computed as follows:

$$D(P_T(s,d)) = \sum_{e \in P_T(s,d)} D(e) \tag{2}$$

- **Bandwidth :** the bandwidth available on the path $P_T(s,d)$ is defined as minimum bandwidth available on each edge of the path $P_T(s,d)$ and is computed as follows:

$$B(P_T(s,d)) = \min_{e \in P_T(s,d)} B(e) \tag{3}$$

if we consider Δ_d as delay constraint and β_d as bandwidth constraint of destination node d, the problem is defined as finding the multicast tree *T(s,M)* such that it includes the source node s and all destination ones, $d \in M$, and reaches the following conditions:

$$\mathrm{Cost}(T) = \min\Big(\sum_{(a,b)\in E_T} C(a,b)\Big) \tag{4}$$

$$D(P_T(s,d)) \leq \Delta_d , \qquad \forall d \in M \tag{5}$$

$$B(P_T(s,d)) \geq \beta_d , \qquad \forall d \in M \tag{6}$$

A. Evaluating the solutions

One of the most important factors in evaluating the solutions in meta-heuristic algorithms is selecting a proper objective function such that can evaluate and give value the quality, in the best manner. Regarding to proposed algorithm, the proper evaluation function for creating the least cost multicast tree, while considering the bandwidth and delay constraints, can be shown as following:

$$F(T(s,M)) = f_c + A \times f_d + B \times f_b \tag{7}$$

Where:

$$f_c = C(T(s,M)) = \sum_{(a,b)\in E_T} C(a,b)) \tag{8}$$

$$f_d = \sum_{d\in M} \varphi(\mathrm{Delay}(P(s,d) - \Delta_d) \tag{9}$$

$$\varphi(z) = \begin{cases} 0 & z \leq 0 \\ z & z > 0 \end{cases} \tag{10}$$

$$f_b = \sum_{d\in M} \omega(\mathrm{Bandwidth}(P(s,d) - \beta_d) \tag{11}$$

$$\omega(z) = \begin{cases} 0 & z \geq 0 \\ |z| & z < 0 \end{cases} \tag{12}$$

The coefficients, A and B are regarded as the penalizing coefficients and $\varphi(z)$ and $\omega(z)$ are penalizing functions.

III. Harmony search algorithm

The harmony search algorithm (HS) has been presented by Geem et al. (2001) and inspired from a routine that is traversed by a musician to reach its intended harmony. This algorithm includes five main steps and operates in a manner in which the first step, the primary parameters of

algorithm are initialized; these parameters include the size of harmony memory (*HMS*), the probability of selection from the harmony memory (*HMCR*[1]), the probability of creating partial changes (*PAR*[2]) and the maximum number of generations. Then in second step, the initial population is created for beginning the algorithm and is known as initial harmony memory. This memory is shown as a matrix in which each row is equivalent with a harmony or solution of the problem and each its column denotes a note or a variable of the intended solution.

$$\mathrm{HM} = \begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_{N-1}^1 & x_N^1 \\ x_1^2 & x_2^2 & \cdots & x_{N-1}^2 & x_N^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_1^{HMS-1} & x_2^{HMS-1} & \cdots & x_2^{HMS-1} & x_2^{HMS-1} \\ x_1^{HMS} & x_2^{HMS} & \cdots & x_2^{HMS} & x_2^{HMS} \end{bmatrix}$$

Figure .1. The harmony memory matrix

In the third step which is the most important one among the steps of harmony search algorithm, the new harmonies are generated based on operators of harmony search algorithm. In this step, all variables of new harmonies are taken from harmony memory with probability *HMCR*, or are selected randomly from an acceptable range, with the probability 1-*HMCR*. Then they are embedded in the new harmony.

$$x_i' \leftarrow \begin{cases} x_i' \in \{x_i^1, x_i^2, .., x_i^{HMS}\} & \text{with probability } \mathrm{HMCR} \\ x_i' \in X_i & \text{with probability } 1 - \mathrm{HMCR} \end{cases} \quad (13)$$

The variables which are selected from the harmony memory are changed with the probability *PAR* or are left intact with the probability *1-PAR*.

$$x_i' \leftarrow \begin{cases} x_i' \pm \mathrm{rand}(0,1).\mathrm{bw} & \text{with probability } \mathrm{PAR} \\ x_i' & \text{with probability } 1 - \mathrm{PAR} \end{cases} \quad (14)$$

In the fourth step, the worst harmonies in harmony memory, are replaced with the best harmonies generated in the new generation; in the other word, the harmony memory is updated. Finally, in the last step, the condition for terminating the algorithm is investigated. This

[1] Harmony memory considering rate
[2] Pitch adjusting rate

condition can be reaching to maximum generation number, having no change in harmony memory during several generations and etc.

IV. Presenting the proposed method

In this section introduces the proposed method for solving the QoS multicast routing problem. In the proposed method, before starting the algorithm, a preprocessing phase is performed during

Which all edges of the graph that violated the bandwidth constraints that mean their bandwidth is less than minimum bandwidth determined by the problem $(B(e) < \beta_d)$ must be removed from the graph. As the result of this, the objective function of Equation (7) changes as follows:

$$F\big(T(s,M)\big) = f_c + A \times f_d \tag{15}$$

If in the refined graph, the source node and all destination nodes are not in a connected sub-graph, it can be said this topology does not meet the bandwidth constraint. In this case, the source should negotiate with the related application to relax the bandwidth bound.

A. Representation

The first step and one of the most important parts in designing a meta-heuristic algorithm is representation of solution space. As already explained, the multicast tree is structured by a set of paths which connect the source node to each of destination nodes. In order to coding the multicast tree, each of paths must be coded based on a unique method. In this paper, we have used the integer coding structure for coding each of solutions. In this method, each path in multicast tree is denoted by a sequence of positive integer numbers, each of which depicts the intended node number in the path. And the position of each node in the sequence expresses the position of that node in the path. As an example, the first node in this sequence always is known as the source node and the last node is known as destination node.

B. Generating the initial population

The procedure for generating the initial members of harmony memory, includes selecting all paths between source and destination nodes for each harmony, by using the path random traversing method. In this method traversing the graph starts from the source node and each node randomly selects one of its neighbors as the next node. In order to prevent the circulation, the nodes passed through are placed in a list so that they can't be selected again. This procedure continues until all multicast destinations are met and can be said that the multicast tree has been generated. The Figure (2) shows the pseudo code for generating the initial harmony memory.

1: i=0;
2: While i<harmony memory size
//Generate harmony i
3: j=0;
4: $V_T = E_T = \emptyset$
5: While j<|M| do
6: Search a random Path $P_T(s, d_j)$ *which can guarantee* $T \cup P_T$ *be an acycle graph;*
7: Add all the nodes and links in P_T *into* V_T *and* E_T *, respectively*
8: j++
9: End while
10: i++
11: End while

Figure .2. Pseudo code for generating harmony memory

C. Generating the new population

After creating the initial harmony memory, each harmony (solution) is evaluated by the objective function presented in equation (15). By terminating this step, the algorithm enters to its main loop; this loop continues until the stopping condition is reached, and in each generation, the new harmonies are generated based on harmony search algorithm and as following. For each harmony, until all its variables are determined (the multicast tree is created), this procedure continues:

- A harmony is randomly selected from the harmony memory, with the probability *HMCR* and its i^{th} variable is considered as i^{th} variable of the new harmony, otherwise with the probability 1-HMCR a random path $P_T(s, di)$ is created in the acceptable range and is considered as i^{th} variable of the new harmony.
- If the i^{th} variable has been selected from the harmony memory, it must be partially changed, with the probability *PAR*. In this paper, the random changing method has been used for this operator. For this purpose, one point such *p*, in the intended path $P_T\left(s \overset{H_j}{\rightarrow} d_i\right)$ is selected as adjust point and the sub-branch path from *p* to the intended destination, changes in a random manner.
- Finally, all generated harmonies are evaluated by objective function defined in equation (15) and the worst harmonies in the harmony memory are replaced with the best generated harmonies in the new generation, and the harmony memory is updated. It must be noted that the size of harmony memory is constant in all generations. Figure (3) shows the flowchart for the proposed algorithm.

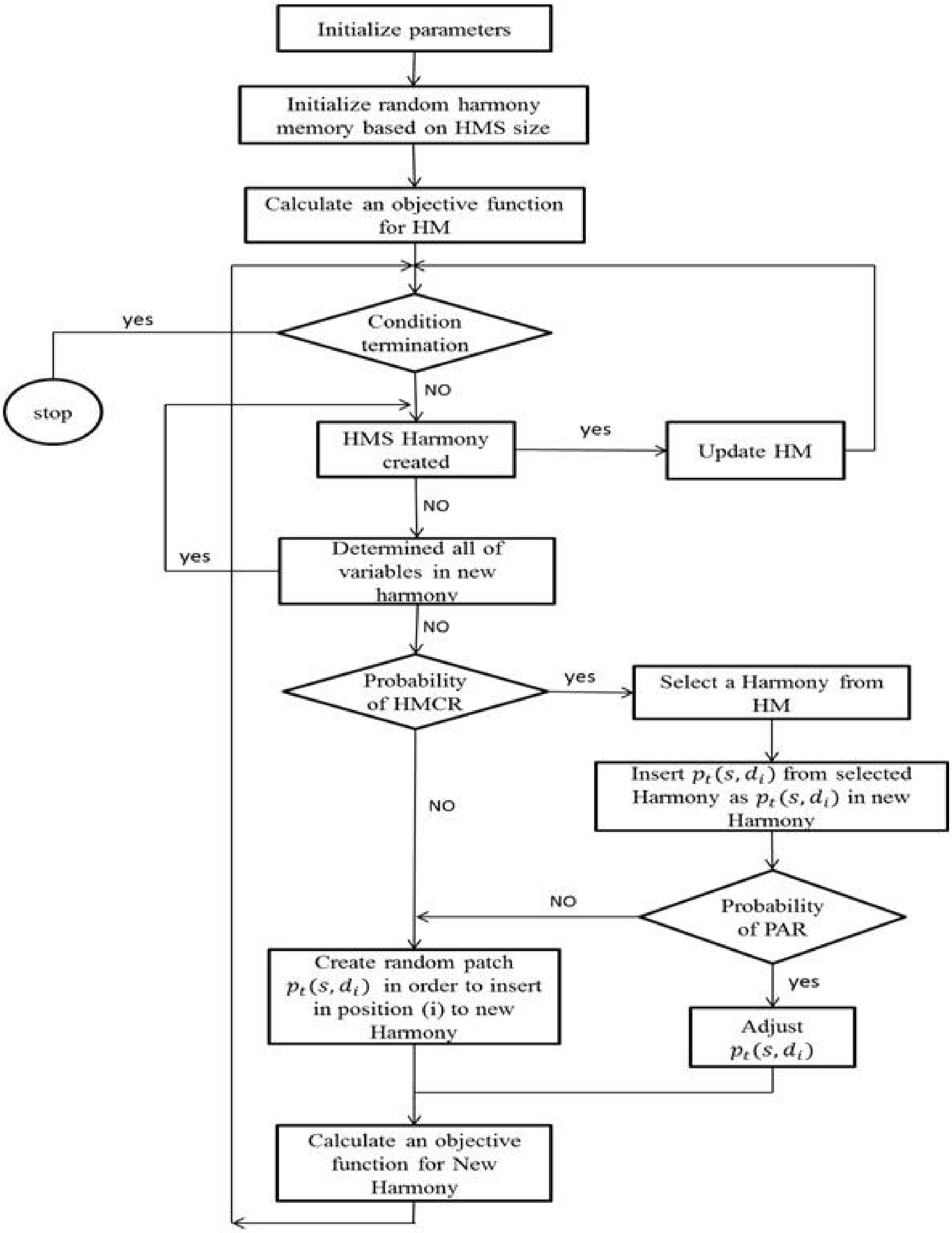

Figure .3. The flowchart for proposed algorithm

V. Simulation Results

In order to simulate the proposed method, the software Matlab R2009a has been used. Also for creating the network topology, some graphs with different sizes have been generated by using a random graph generator based on Waxman (1988). For each edge in these graphs, three weights values in the range [1,10] were assigned, randomly, which denote the cost, bandwidth and delay parameters. For each graph, the source node and all destination nodes were selected randomly. The function computing the relative error percentage, $\frac{cost-cost_{best}}{cost}$, has been used to compute the relative difference percentage between the cost obtained via the experiments and the global optimal cost in which the *cost* denotes the cost obtained from experiments and $cost_{best}$ is the global optimal cost obtained from 10 times repeating the algorithms (Tayeb Taher and Rahmani, 2012; Haghighat et al., 2004) and during 2000 steps in each iteration. Finally, in order to evaluating the performance of presented method, the proposed algorithm has been compared with three algorithms, Genetic algorithm proposed by

Haghighat et al., (2004), PSO algorithm (Wang et al., 2007) and the honey bee of Tayeb Taher and Rahmani, (2012).

Table (1) and the Figure (4), respectively show the relative error rate and the execution time of each algorithm, concerning with increasing the number of terminals (the number of multicast group members). For performing this experiment, a random graph with 50 nodes was considered and the number of multicast group members was selected as 10% to 90% of all graph nodes.

TABLE I

COMPARING THE RELATIVE ERROR RATE OF ALGORITHMS PROPORTIONAL TO INCREASING THE NUMBER OF TERMINALS

Bees	GA	Proposed	PSO	Terminal Number
0	0	0	0	5
0	0	0	0	10
0	0	0	0	15
0	0	0	0.009	20
0.006	0.0031	0.009	0.0031	25
0.0079	0.0093	0.0042	0.074	30
0.011	0.0199	0.0095	0.0196	35
0.0183	0.0308	0.0165	0.0297	40
0.0312	0.0394	0.0179	0.0391	45

As can be seen, by increasing the number of terminal nodes, the relative error rate of proposed algorithm grows with a lower rate compared to other algorithms.

As can be seen in Figure (4), the execution time of proposed algorithm has a better performance compared to genetic and Bee algorithms, concerning with increasing the number of multicast Group members, and in comparison with the particle swarm optimization, has a more speed grow.

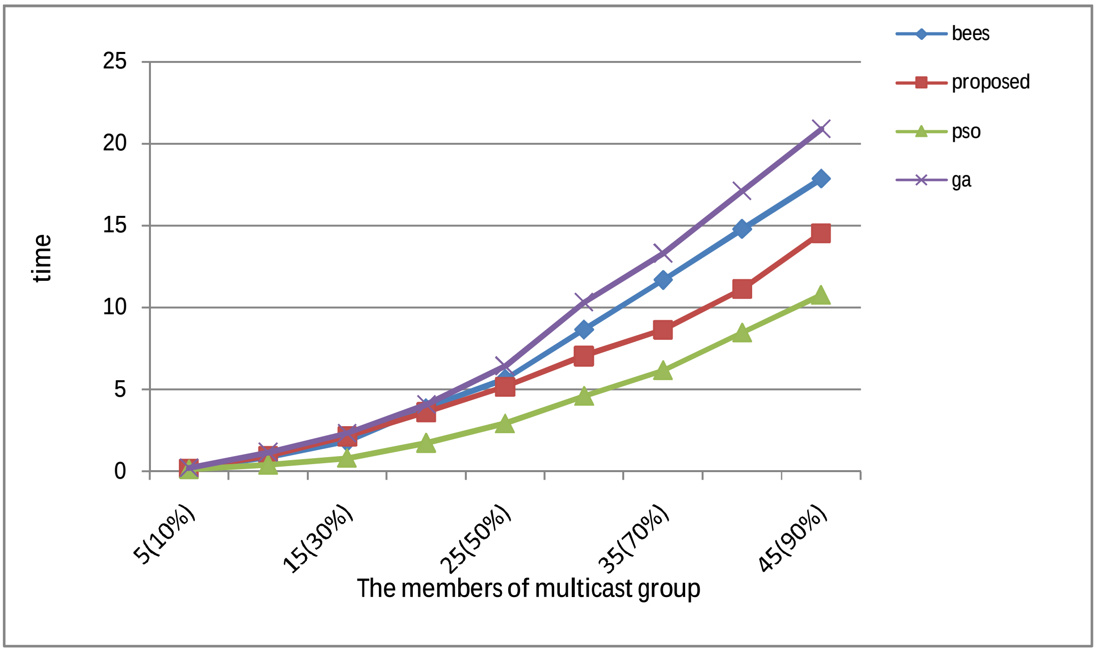

Figure .4. The effect of increasing the number of terminals on execution time of each algorithm

The next experiment evaluates the behavior of each algorithm until reaching to global optimal solution. In order to do this experiment, a graph with 100 nodes has been considered and the number of multicast group members has been considered as 20% of all node count. The results of this experiment have been shown in Figure (5).

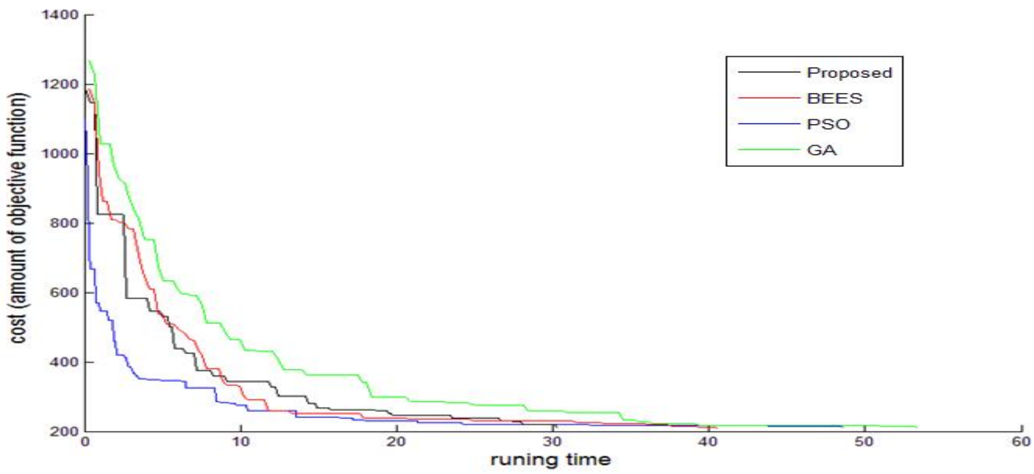

Figure .5. Comparing algorithms behaviors during the execution

As can be seen in the figure, the reduction rate of the cost of proposed algorithm has a proper speed and the proposed algorithm converges to global optimal solution with a more speed than other algorithms and without trapping in local optima. The next experiment shows the effect of growing the network size on convergence to global optima for the algorithms. For performing this comparison, 10 random graphs with node counts between 10 and 100 were generated, and the number of multicast group members was set to 20% of all nodes in each graph. Then the average execution time until reaching the global optimal solution was computed for each algorithm.

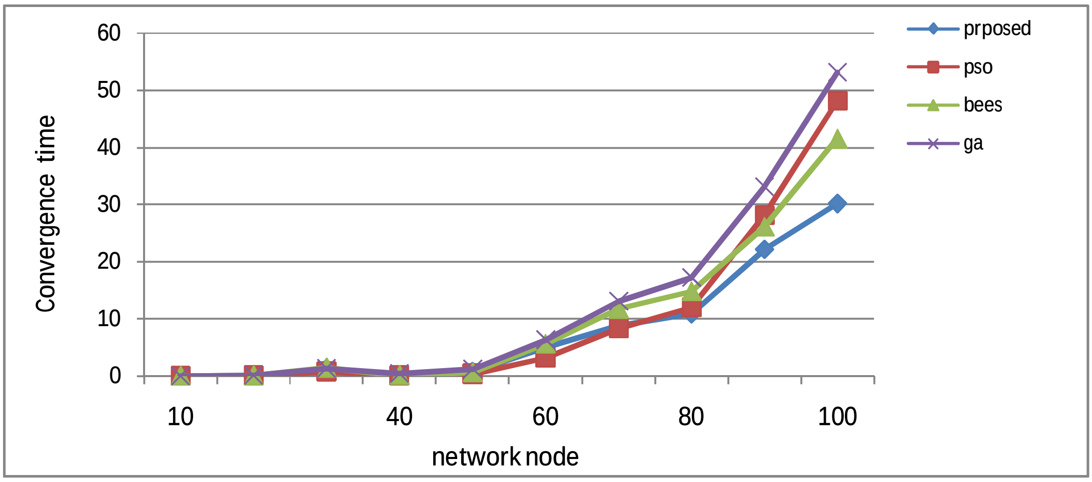

Figure .6. The effect of increasing network node number on convergence time to global optima

As can be seen, by growing up the network size, the time for converge the proposed algorithm to global optimal solution has increased by a lower slope in comparison with other algorithms.

VI. Conclusions

In this paper, first, we describe the multicast routing problem while considering QoS constraints and then the proposed algorithm, based on harmony search were presented for solving this problem. The simulation results showed that the proposed algorithm has not be caught in local optima, and although the growing ratio for its execution time proportional to increasing in multicast group members is greater than the particle swarm optimization algorithm, but in general, regarding the relative error rate, the quality of solutions and also regarding the time to reach the global optima solution has a more proper performance in comparison with particle swarm optimization algorithm and other compared algorithms.

References

i. Armaghan M. and Haghighat A (2009). QoS Multicast Routing Algorithms Based on Tabu Search with Hybrid Candidate List. *Future Generation Information Technology, Springer Berlin Heidelberg*. Vol(5899), pp 285-294.

ii. Forsati R, Haghighat AT and Mahdavi M (2008). Harmony search based algorithms for bandwidth-delay-constrained least-cost multicast routing. *Computer Communications*, Vol(31) pp 2505-2519.

iii. Ghaboosi, N and Haghighat A (2007). Tabu search based algorithms for bandwidth delay-Constrained least-cost multicast routing. *Telecommunication Systems*, Vol(34),pp 147–166.

iv. Gong B, Li L and Wang X (2007). Multicast Routing Based on Ant Algorithm with Multiple Constraints. *Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on*, pp 1945-1948.

v. Geem ZW, Kim JH and Loganathan GV (2001). A New Heuristic Optimization Algorithm: Harmony Search. *Simulation*, Vol(76), No.2, pp 60–68.

vi. Haghighat AT, Faez K, Dehghan M and Mowlaei A (2004). GA-based heuristic algorithms for bandwidth-delay-constrained least-cost multicast routing. *Computer Communications*, Vol(27), No.1, pp 111-127.

vii. Karp RM (1972). Reducibility among combinatorial problems. *Complexity of Computer Computations. Plenum Press*, pp 85-103.

viii. Kompella VP, Pasquale JC and Polyzos GC (1993). Multicast routing for multimedia communication. *Networking, IEEE/ACM Transactions on*, Vol(1), No.3, pp 286-292.

ix. Kun Z, Heng W and Feng-Yu L (2005). Distributed multicast routing for delay and delay variation-bounded Steiner tree using simulated annealing. *Computer Communications*, Vol(28), No.11, pp 1356-1370.

x. Koyama A, Nishie T, Arai J and Barolli L (2008). A GA-based QoS multicast routing algorithm for large-scale networks. *International Journal of High Performance Computing and Networking*, Vol(5), No.5, pp 381-387.

xi. Liu J, Sun J and Xu W(2006). QoS Multicast Routing Based on Particle Swarm Optimization. *Intelligent Data Engineering and Automated Learning – IDEAL 2006*, Vol(4224), pp 936-943.

xii. Qing Z, Parsa M and Garcia-Luna-Aceves JJ (1995). A source-based algorithm for delay-constrained minimum-cost multicasting. *INFOCOM '95. Fourteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Bringing Information to People. Proceedings. IEEE*, Vol(1), pp 377-385.

xiii. Salama HF, Reeves DS, Viniotis Y (1997). Evaluation of multicast routing algorithms for real-time communication on high-speed networks. *Selected Areas in Communications, IEEE Journal on*, Vol(15), No.3, pp 332-345.

xiv. Sun B and Li L (2006). A QoS multicast routing optimization algorithm based on genetic algorithm. *Communications and Networks, Journal of*, Vol(8), No.1, pp 116-122.

xv. Sun J, Fang W, Wu X, Xie Z and Xu W (2011). QoS multicast routing using a quantum-behaved particle swarm optimization algorithm. *Engineering Applications of Artificial, Intelligence*, Vol(24), No.1, pp 123-131.

xvi. TayebTaher M and Rahmani AM (2012). Multicast Routing in Computer Networks considering Quality of Service (QoS) based on Honey Bee Algorithm. *International Journal of Computer Applications*,Vol(58), No.2, pp 20-24.

xvii. Wang H, Fang J, Wang H and sun Y (2004). TSDLMRA: an efficient multicast routing algorithm based on Tabu search. *Journal of Network and Computer Applications*, Vol(27), No.2, pp 77-90.

xviii. Wang Z, Sun X and Zhang D (2007). A PSO-Based Multicast Routing Algorithm. *Natural Computation, 2007. ICNC 2007. Third International Conference on*, Vol(4), pp 664-667.

xix. Wang H, Shi Z and Li S (2009). Multicast routing for delay variation bound using a modified ant colony algorithm. *Journal of Network and Computer Applications*, Vol(32), No.1, pp 258-272.

xx. Waxman BM (1988). Routing of multipoint connections. *Selected Areas in Communications, IEEE Journal on*, Vol(6), No.9, pp 1617-1622.