Maximizing Network Lifetime for Multicast Communication in Mobile Ad Hoc Networks

Faezeh Akbari [†], Seyed Majid Noorhosseini[††], Mahnaz Imanazari Bonab [†††]

[†, †††]Computer Engineering and Information Technology Department, Islamic Azad University, Qazvin, Iran
[††]Department of Computer Engineering and Information Technology, Amirkabir University of Technology, Tehran, Iran

Summary

Energy consumption is a crucial design concern in wireless ad hoc networks since wireless nodes are typically battery-limited. Power failure of a mobile node not only affects the node itself but also its ability to forward packets on behalf of others and hence affects the overall network lifetime. To this purpose, routing algorithms must be developed to discover routes between mobile nodes that energy consumption of nodes is considered as a primary goal. We present an energy and lifetime aware multicast algorithm by route selection strategy to equalize energy consumption of nodes. It can balance individual node's battery power utilization and hence prolongs the entire network's lifetime. The scheme has been simulated in various network scenarios to test operation effectiveness in terms of performance parameters such as packet delivery ratio, network lifetime and total consumed energy of network. Also, we compared proposed protocol with ODMRP. We observe from the simulation that proposed protocol performs better than ODMRP.

Key words: *Ad hoc networks, Multicast Algorithm, energy consumption, network lifetime*

1. Introduction

An ad hoc network is a multi-hop wireless network without any fixed network infrastructure. An ad-hoc network is usually formed by the mobile hosts that are participating at common activities and located over a certain geographically limited area.
Multicasting is the transmission of datagram to a group of hosts. The use of multicasting within a network has many benefits. Multicasting reduces the communication costs for applications that send the same data to the multiple recipients. Instead of sending via multiple unicasts, multicasting minimizes the link bandwidth consumption, sender and router processing, and delivery delay [1]. When a destination node is out of reach of the source node, the connectivity between the two nodes is maintained by the intermediate nodes. The energy constraint is a critical issue for such a network, and a lot of works have focused on how to optimize the energy consumption and keep the same level of network efficiency. Those routing protocols can generally be grouped into two categories [2]: tree-base and mesh-based. In a tree-based scheme, data delivery between a source/destination pair is provided over a single shortest path, whereas in a mesh-based scheme, data are delivered over multiple paths. The AMRoute (Adhoc Multicast Routing)[3] , the AMRIS (Ad hoc Multicast Routing protocol utilizing Increasing idnumberS)[4] and Firework[5] are the examples of the tree-based protocols. Examples of mesh-based protocols are the ODMRP (On-Demand Multicast Routing Protocol) [6], the CAMP (Core-Assisted Mesh Protocol) [7] and ABMRS (Agent-Based Multicast Routing Scheme) [8].
AMRoute [3] is a tree-based protocol. It creates a bidirectional shared multicast tree using unicast tunnels to provide connections between multicast group members. A node wanting to forms a multicast group will create one with a core of size 1. The core will then broadcast *JOIN-REQ* messages in the network. When a member node receives the *JOIN-REQ* it responds with *JOIN-ACK,* storing the information about the source of *JOIN-REQ.* This process creates a path from receiver to the source. After the mesh creation each source transmits *TREE-CREATE* messages. A node receiving a *TREE-CREATE* message will mark the path the message came from as a tree link. If a duplicate *TREE-CREATE* packet is received, a *TREECREATE-NAK* is sent back to the source.
AMRIS [4] is an on-demand, shared tree based multicast protocol. In this protocol, each node in a multicast session generates session-specific multicast session member id (msm-id), after receiving the *NEW-SESSION* message from its parent node. The *NEW-SESSION* message transmission is initiated by a special node called *Sid*, at which the shared tree is rooted. The msm-id increases from the root towards leaf nodes, which indicates the flow of multicast data. The protocol uses periodic, short broadcast beacon packets to determine whether a link has been broken. Upon link break, it executes a branch reconstruction process to maintain the multicast tree.
CAMP [7] expands the idea of core based tree, to form the mesh. But unlike the core based tree protocol, it contains more than one core. This protocol defines a shared multicast mesh for each multicast group to maintain the connectivity of multicast groups, even during the frequent

Manuscript received December 5, 2009
Manuscript revised December 20, 2009

movement of network routers. One or multiple cores are defined per multicast group to assist in join operations; therefore, CAMP eliminates the need for flooding. CAMP uses a receiver-initiated approach for receivers to join a multicast group. A node sends a *JREQ* towards a core if none of its neighbors is a member of the group; otherwise, it simply announces its membership using either reliable or persistent updates. If cores are not reachable from a node that needs to join a group, the node broadcasts its *JREQ* using an expanded ring search method, which eventually reaches some group member.

ABMRS [8] employs a set of static and mobile agents in order to find the multicast routes. The steps of the ABMRS are the following: reliable node identification, reliable node interconnection, reliable backbone construction, multicast group creation, and network and multicast group management. The *RMA* (Route Manager Agent) at each node computes the RF (Reliability Factor, which depends on various parameters such as power ratio, bandwidth ratio, memory ratio, and mobility ratio) and advertises to each of its neighbors. The *NIA* (Network Initiation Agent) at each node receives the advertised packet and determines who has the highest RF. The node with the highest RF will announce itself as a reliable node and inform its *RMA*. The *RMA* in each of the reliable nodes will broadcast information about their adjacent reliable nodes throughout the network. Using this information, *RMA* applies Dijkstra's algorithm to compute the routes between the reliable nodes and generate the forwarding table. The intermediate nodes generate the forwarding table based on the information given by *NIA*s. Finally, the multicast group is created by the *MIA* (Multicast Initiation Agent). *MIA* travels to each reliable node and invites the multicast group to join. After performing the initial membership survey and collecting the necessary group membership information, the *MIA* forms an initial multicast tree comprising reliable nodes, intermediate nodes, and group members. The *NMA* (Network Management Agent) is responsible for managing the multicast group. Whenever an intermediate node or reliable node is disconnected, the *NMA* will ask the *RMA* to initiate the *NIA* to find the new paths between the reliable nodes. A child node has the responsibility of finding a new reliable node whenever there is a disconnection between a reliable node and its child node because of mobility.

The primary goal of the conventional multicast routing protocols has been to reduce the delay since most multicast applications tend to be delay sensitive audio/video broadcasting. Hence, the cost of the multicast routing protocols is designed to construct a multicast tree that minimizes the communication latency. Since the number of hops is a good heuristic metric for capturing this latency. In wireless ad hoc networks, there are two other criteria that make routing design an even more complicated task, i.e., mobility and energy efficiency. The issue of mobility has extensively been addressed [2-7]. In fact, the performance of multicast routing protocols has been evaluated in regard to their robustness to link failure due to the mobility. However, there has been little work on developing a wireless multicast routing protocol in which energy is a key objective or constraint. Most of the existing works require a global view of the network and cannot be applied in a distributed way whereby the nodes have only local knowledge.

This paper addresses the problem of designing a lifetime aware multicast routing protocol and algorithm that can be applied to an ad hoc network where nodes only have limited knowledge of network topology and the power states of other nodes in the network. The paper is structured as follows. Section 2 reviews ODMRP. Section 3 describes the proposed lifetime aware multicast routing algorithm. Section 4 shows the simulation results and section 5 is conclusion.

2. ODMRP Description

We first explain ODMRP because we use ODMRP as a point of comparison for proposed protocol since it has been demonstrated to perform well [6]. In addition, our multicast forwarding tree flooding operates similarly to the forwarding group flooding in ODMRP, allowing us to better compare the different aspects and overall behavior of the two protocols. ODMRP is an on-demand; mesh-based multicast protocol that attempts to establish a forwarding group only when a source of the group has data to send. Source of the multicast group periodically generates JOIN QUERY, which contains information about itself, and floods the packets throughout the network. If the node receives the JOIN QUERY, it updates the routing table to record the routing information for the source of the received JOIN QUERY, and rebroadcasts it to its neighbors. The nodes in the forwarding group form a mesh that connects the group members together. When a member of the multicast group receives a JOIN QUERY, it constructs and broadcasts a JOIN REPLY packet containing the source ID and the upstream node ID to all of its neighbors. Upon receiving a JOIN REPLY, a node which ID matches the upstream ID in the packet realizes that it is on the path between the source and a member, so it becomes a forwarding node for the group by setting its FG_FLAG (Forwarding Group Flag). It then constructs and broadcasts its own JOIN REPLY using its corresponding upstream node ID. In ODMRP, sender exploits periodic flooding of control packets to refresh group connectivity and update the routes. Redundancy paths in ODMRP forwarding yield high overhead and energy and also additional load as the network size and the number of member of the multicast group increases.

3. Proposed Protocol

The objective of our routing is to extend the useful service life of a MANET and use the battery fairly. This is highly desirable in the network since death of the certain nodes leads to network partitions, rendering other live nodes unreachable. It is better to find a good path with low energy consumption from the intermediate nodes. The path between the multicast member and forwarding set to the multicast source is not always the shortest. We use cost function used in [9]. This function solves the problem of finding a route *p* at route discovery time *t* such that the following cost function is minimized:

Min $C(\pi, t) = \sum_{i \in \pi} C_{ij}(t)$ (1)

Where $C_{ij}(t) = p_i \cdot \left(\frac{F_i}{R_i(t)}\right)^{\alpha}$ (2)

P_i: Transmit power of node i
F_i: Full-change battery capacity of node i
$R_i(t)$: Remaining battery capacity of node I at time t
α_i (t): A positive weighting factor that increases with $\frac{F_i}{R_i(t)}$

In ODMRP, because the route selection is done based on a shortest path finding algorithm (i.e., those with the minimum number of hops), only mobility of the nodes may cause a selected path to become invalid. In contrast, in proposed protocol, both the node mobility and the node energy depletion may cause a path to become invalid. Since the route discovery in the proposed protocol are more complicated compared to the ODMRP this step will be described in detail.

In proposed protocol, all nodes except the destination calculate their link cost, *Cij* (cf. equation 2) and add it to the path cost in the $cost_s$ field of header of the RREQ packet (cf. equation 1). When the node receives a duplicate RREQ packet, it doesn't drop packet but it examines the cost in the header of RREQ packet. If that cost is less than the cost of a previous copy of that RREQ packet that has already passed through the node, then it will pass on the new copy as well as updating the routing table to record the routing information for the source of the received RREQ, and rebroadcasts it to its neighbors; otherwise, the new RREQ packet is dropped. Destination waits for a specific time after the first RREQ packet arrives. Then destination examines the cost of the route of every arrived RREQ packet. The destination node selects the route with the minimum cost and replies. Subsequently, it will drop any received RREQs. The reply also contains the cost of the selected path and its upstream node ID that send it to all of its neighbors. Each node receives a RREP, matches the upstream ID in the packet with its own ID. Node realizes that it is on the path and so is a member node, so it becomes a forwarding node for the group by setting its FG_FLAG and adds this route along to its route table. This process continues by the member nodes until RREP receives to source. The result of flooding and reply-back procedure is that a multicast tree rooted at the source is constructed.

This scheme results in a significant power saving as it will be shown later and also decreases networks load than ODMRP. Source of the multicast group periodically generates JOIN packet, and send it to tree member nodes. The primary goal of our proposed algorithm is to reduce the energy whereas packet delivery ratio is also up. There for, we use common nodes along different paths to group members. In this algorithm group connectivity can be made more efficient by having some members share common paths to the multicast source with other members in order to further reduce the total cost of forwarding data packets. For the multicast group, proposed protocol determines the forwarding nodes that connect all the group members together and are shared among all the paths to multicast group. The forwarding nodes are initially formed by nodes that are on the paths that yields lowest cost (Eq. 2) between the source and the group members. This protocol creates a forwarding set consisting of all the intermediate nodes on the paths that they have often optimal cost, as illustrated in figure 1. However, group connectivity can be made more energy efficient by having *D1* chooses another path that is shared by *D2* to reduce the size of the forwarding set, as shown in figure 2, which lowers the total cost of forwarding data packets.

We establish and update routes by a number of steps as described below:

Step1: The source of multicast floods the network to find the least cost path to all multicast receivers. This process has already been explained. The result is that a multicast tree rooted at the source is constructed. Just as a reminder that the multicast receivers add the cost of the path that they select to the header of the reply-back packet before sending this packet to the source.

Step2: The source sorts the receivers in decreasing order of their respective path costs. It means that worst cost is list first.

Step3: Source periodically sends a JREQ command to the receiver whose path has the most cost in the list. Next, it removes the same receiver from the sorted list.

Step4: The receiver that receives a JREQ command tries to connect itself to other path with less cost, so it starts the network searching to find optimal path and connect itself to multicast source in the tree. Number of hops is at most 3.

Step5: receiver of command issues Join Query to find optimal path. This message append $cost_s$ of the sender node and calculates cost function same as RREQ and add it to field of the Cost Route in header Join Query.

Step6: If a node that is a tree member receives this packet (only tree member can response), it examines the connection cost of sender node of Join Query to multicast source through current route with new discovered route to the tree. It yields by comparing the field of $cost_s$ in header of Join Query with sum Cost Route and $cost_s$ of receiver. If new discovered path have best cost, tree member that receive Join Query reply back Reply Query by set the field of Update Route in the packet. Update Route is a binary field. A 0 indicates the discovered route have higher cost than current route. A 1 indicates the discovered route is good.

Step7: After receiving Reply Query packet by the originator of Join Query, the Update Route field of Reply Query is examined. If this field is unset, node sends Join Query to the previous node. This node repeats steps 4 through 7 until packet arrives to the source. Otherwise if Update Route field is set, the node send MACT to previous node on path to arrives to the multicast source.

Step8: Nodes which receive a MACT message addresses to themselves update their route table and remove all the corresponding entries to this multicast group from route table. The multicast source repeats steps 3 through 8 until the list becomes empty and the source sets its timer.

The proposed protocol optimizes the transmitted power level of each node and saves the network nodes energy by using common nodes in different routes. In fact we decrease number of nodes that participate in routing.

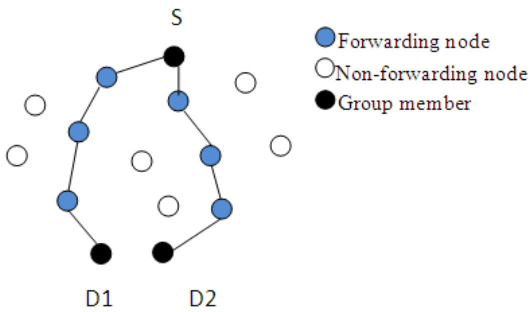

Figure1. The routes formed using the cost function from the source S to the other two members D1, D2.

To the best of our knowledge, we consider figure 1 as an example. The paths are computed and established based on cost function. After the data delivery tree is built, the source node can now broadcast messages to the forwarding nodes. This is accomplished with searching of best path between forwarding nodes. The forwarding nodes using Join Query message start to find best route. If a Join Query message arrives at a node by limited hop count who is currently serving as a forwarding node for the group (node D2 via D1 in this case), it calculates the cost of connecting the node it is currently at join the forwarding set via the forwarding node it previously found. Then it sends a Reply request to its originator via the reverse path by regulating of the Update Route field. The path by optimal cost is updated on the routing table of intermediate nodes, if field of Update Route is set (it means field value is one) and also each of nodes be as a forwarding node. D2 learns new path that have best cost (as seen in figure 2). So, it sends MACT message via reverse paths. Nodes that lie on that path will remove themselves from the forwarding set. D2 utilizes shared path by D1.

We use common nodes for different routes in the multicast group as much as possible. Thus minimizing number of tree nodes helps energy saving. This method could significantly saves energy compared to those multicast algorithms for MANETs, it also maximizes the network lifetime, of course with insignificant overhead.

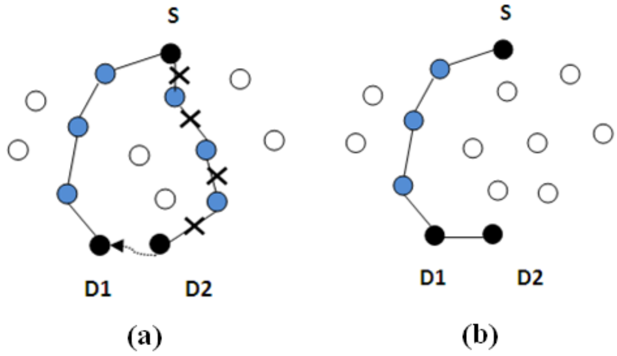

Figure2. D1 shares the same path to the source S with D2, which results in more efficient data packet forwarding. Number of forwarding nodes is lower than former.

4. Simulation Results

We have simulated our proposed protocol by C++ program and compared it to ODMRP (but without mobility prediction which requires GPS). The simulation environment consists of different nodes in a rectangular region of 800m*800m.

Each of nodes is randomly distributed in the region and has transmission range of 150 meters. Each node was randomly assigned an initial energy, which varied between 1400 and 1500 joule. There is one source in our multicast group and the group size varies from 2 to 8 nodes. The performance evaluation metrics used in our simulation are as follows: packet delivery ratio, network lifetime and energy consumption.

4.1 Packet delivery ratio

Packet delivery ratio is defined as number of data received by destinations over the number of data sent by source. The packet delivery ratios as a function of mobility speed are shown in Figure 3. We can observe that as speed increases because of links break and as time passes the nodes die because of battery energy exhaustion, so the packet delivery ratios decrease in both protocols.

Many of packets might not have reached to their intended destination due to the lack of existence of a route between the source and target destination and this may occur when some nodes die out. The behavior of both protocols in different speeds is same, but it is possible ODMRP operates better in high speeds. The proposed protocol has similar delivery ratio in same speed but with more delay.

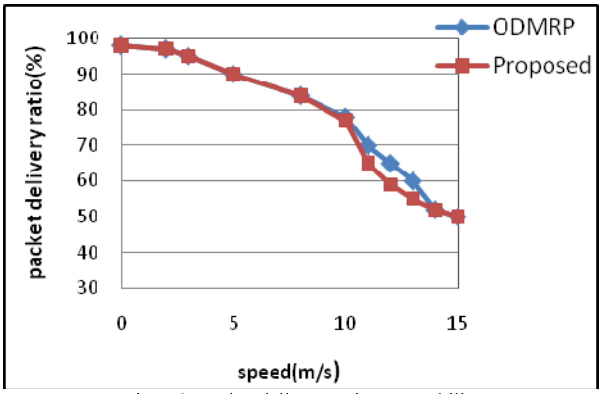

Figure3. Packet delivery ratio Vs. Mobility.

4.2 Network lifetime

Network lifetime can be defined as the time taken for the first node or a fixed percentage of the nodes to energy resource exhaustion.

We define the network lifetime as the total elapsed time from the state of network connectedness to a state in which the network connectivity ratio drops to 30%. We chose 30% because we observed that near this percentage, the network graph becomes disconnected that it cannot be considered as functional anymore. The disconnection in the network graph results in disconnectedness of most of the remaining multicast connections. As the group size increases (as seen in figure 4), the network lifetime decreases since more nodes participate in the multicast tree connection and the probability that some receivers in each connection become unreachable increases. We observe that the proposed protocol achieves longer lifetime than the ODMRP, since it selects energy efficient routes and uses lower number of nodes.
It balances the power consumption for the all nodes in the network.

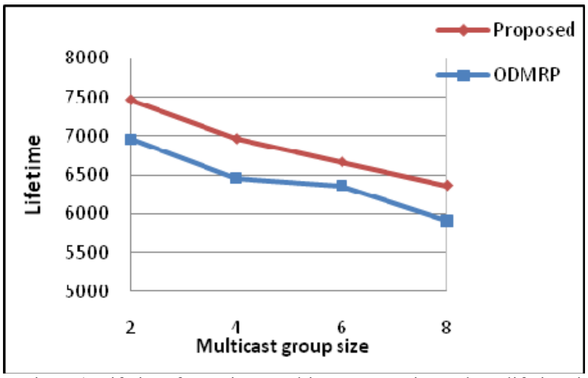

Figure4. Lifetime for various multicast group sizes where lifetime is defined as from time 0 until network connectivity ratio drops to 30%.

4.3 Energy Consumption

The node energy consumption measures the energy dissipated by the all network nodes during simulation time. We use the common nodes as much as possible. In this scenario we decide to study the amount consumed energy by all nodes of network during the simulation time. The number of multicast group varies between 2 and 8. As demonstrated in figure 5, by balancing consuming of energy on the nodes and decreasing their number of the forwarding nodes, the network will be partitioned later and the network can be used for longer period of time. As the multicast group size increases the number of the necessary forwarding nodes also increases. Hence, energy consumed of entire network rises in the both protocols.

The maximum energy consumption of the proposed protocol is smaller than ODMRP, and the energy consumption increases slowly when group size increases. This means our protocol is scalable and so results in a longer network lifetime. Our protocol can efficiently balance the energy consumption between nodes, which increases the network lifetime.
ODMRP uses alternate paths thus it increases energy consumption whereas the proposed protocol utilizes lower number of the forwarding nodes.

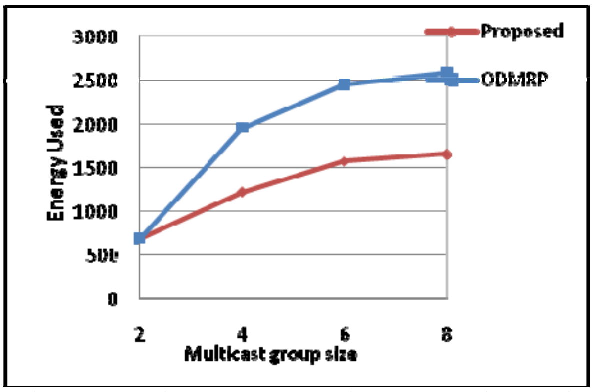

Figure5. Comparison Energy consumption ODMRP and Proposed protocol.

5. Conclusion

In this paper, we have addressed the problem of the proper route discovery between mobile nodes. We present an energy and lifetime aware multicast algorithm by route selection strategy to equalize energy consumption of nodes. It tries to reduce the number of nodes used to establish connectivity. For this purpose, the nodes tend to choose paths that are partially shared by others to reduce the size of the forwarding set; hence it utilizes lower forwarding nodes. It can balance individual node's battery power utilization and hence prolongs the entire network's lifetime. Extensive simulation results were provided to evaluate the performance of proposed protocol with respect to a number of different metrics (i.e., the network lifetime, energy consumption and the packet delivery ratio) in comparison to ODMRP. These results clearly demonstrate the proposed protocol performs better than ODMRP.

References

[1] S. Paul, "Multicasting on the Internet and its Applications", Kluwer Academic Publishers, ISBN 0792382005, June 1998.

[2] Z.Wang, Y. Liang, and L. Wang, "Multicast in Mobile Ad Hoc Networks," IFIP International Federation for Information Processing, vol. 258/2008, pp. 151-164, 2008.

[3] J. Xie, R. R. Talpade, A. McAuley, and M. Liu, "AMRoute: Adhoc Multicast Routing Protocol," Mobile Networks and Applications, vol. 7, pp. 429-439, 2002.

[4] C.W.Wu, Y.C.Tay, and C.-K. Toh,"Adhoc Multicast Routing protocol utilizing Increasing id-numberS (AMRIS)," draft-ietf-manet-amris-spec-00.txt, 2000.

[5] L. K. Law, S. V. Krishnamurthy, and M. Faloutsos, "A novel adaptive protocol for lightweight efficient multicasting in ad hoc networks," Computer Networks, vol.51, pp. 823-834, 2007.

[6] S. J. Lee, W. Su, and M. Gerla, "On-demand Multicast Routing Protocol in Multihop Wireless Mobile Networks," ACM/Baltzer Mobile Networks and Applications, special issue on Multipoint Communications in Wireless Mobile Networks, vol. 7, pp. 441-453, 2002.

[7] J. J. Garcia-Luna-Aceves and E. L. Madruga, "The Core Assisted Mesh Protocol," IEEE Journal on Selected Areas in Communications, Special Issue on Ad-Hoc Networks, vol. 17, pp. 1380-1394, 1999.

[8] S.S.Manvi and M. S. Kakkasageri, "Multicast Routing in Mobile Ad hoc Networks by using a Multiagent System," Information Sciences, vol. 178, pp. 1611-1628, 2008.

[9] M.Maleki, and M.Pedram,"Lifetime-Aware Multicast Routing in Wireless Ad Hoc Networks ," In Processings of IEEE WCNC ,vol.3, pages 1317-1323, 2004.

Faezeh Akbari received B.E. degree from Kashan Azad University in 2004. She received M.E. degree from Qazvin Azad University in 2009. Her research interests include network management as well as sensor and ad hoc networks.

Seyed Majid Noorhosseini received the B.Sc. and M.Sc degrees from Amirkabir University of technology in 1986 and 1989, respectively. He received his PhD degree from McGill University in Montreal, Canada in 1994. He was a senior scientist at Nortel Networks in Canada and U.S during 1996-2005, working in different areas of network management. He has a US patent 6707795 in alarm correlation method and system. He is now with the Department of Computer Engineering and Information Technology at Amirkabir University of Technology.

Mahnaz Imanazari Bonab received her B.E. degree from Payamnoor University in 2006. She received her M.E. degree from Qazvin Azad University in 2009. Her research interests include network management, databases and image processing.