An efficient scheduling algorithm for radio resource reuse in IEEE 802.16j multi-hop relay networks

Wen-Hwa Liao, Kuei-Ping Shih, Chen Liu, Avinash Kumar Dubey, Sagar Arora, Sital Prasad Kedia

Department of Information Management, Tatung University, Taiwan
Department of Computer Science and Information Engineering, Tamkang University, Taiwan
Samsung India Software Operations Pvt. Ltd., Bangalore, India
Department of Computer Science and Engineering, Indian School of Mines University, Dhanbad, India

Abstract

The IEEE 802.16j standard for WiMAX introduced the concept of relay station in order to increase the service area and decrease the deployment cost of the network. In this paper, we have proposed an efficient scheduling scheme for IEEE 802.16j networks, which maximizes the number of connections that are scheduled in a particular time slot. The proposed scheme schedules the connections based on their priority, which is decided by the quality of service (QoS) requirement of the connections. A selected connection can be scheduled in the current timeslot only if it is not having interference with any of the already scheduled connections. Our algorithm considers scheduling of lower priority connections even if any higher priority connection is not scheduled due to interference. Thus, the delay for lower priority connections is reduced without increasing the delay for higher priority connections. In addition, our algorithm achieves higher frame utilization and higher system throughput by reducing the length of the schedule.

1. Introduction

A wireless network is considered as cheaper and timesaving alternative to the traditional wired counterpart. In addition, some developing countries and uncivilized regions often lack the infrastructure required for wired networks. To satisfy the huge demand of wireless services, worldwide interoperability for microwave access (WiMAX) has been advocated as IEEE 802.16 wireless technology with high throughput over long distances (up to 30 miles). WiMAX provides data rate of up to 75 Mb/s with single channel and 350 Mb/s with multiple channels [1,2].

Basic WiMAX network includes a base station (BS) and several subscriber stations (SSs). Usually, one BS serves several SSs in its transmission range. Due to the limited transmission range of the BS, the number of SSs that can be served by a BS is constrained. Furthermore, signals between the BS and the SSs can be blocked by buildings in the real world. One of the ways to solve this problem is to deploy new BSs in the service area. However, the cost of deploying BS is pretty high. IEEE 802.16j introduces the concept of relay station (RS), which relays traffic between BS and SSs and the cost of deployment of RS is significantly lower than that of BS [3–5]. The architecture of IEEE 802.16j is shown in Fig. 1.

WiMAX uses orthogonal frequency division multiple access (OFDMA), which has the ability to split the data into multiple logical data streams to achieve higher data rate [2]. As shown in Fig. 2, an OFDMA frame structure consists of a downlink (DL)
and an uplink (UL) map. Transmit/receive transition gap (TTG) and receive/transmit transition gap (RTG) give wireless devices time to switch from the transmission mode to the reception mode, or vice versa. The preamble at the beginning of DL sub-frame synchronizes the stations in the network for further transmission. The frame control header (FCH) follows the preamble, which is usually transmitted using quadrature phase shift keying (QPSK) rate of 1/2 with 4 repetitions to ensure that all stations can receive the FCH. Downlink map (DL-MAP) and uplink map (UL-MAP) correspond to the resources allocated to downlink and uplink requests. The SSs receive/transmit data in the order specified in the DL/UL MAP. A DL/UL-MAP consists of many MAP information elements (IEs) which provide information such as when to start transmission for a connection between a BS and an SS. Each SS is assigned a unique connection identifier (CID) in MAP IEs to distinguish it from other SSs.

However, the MAP scheme of traditional 802.16 standards is inefficient when applied to the 802.16j standards, since it allows only one station to transmit in a particular time slot. When a station (BS or RS) is transmitting, other stations should be idle during the same time slot. To enhance the radio resource utilization of IEEE 802.16j networks, some scheduling schemes were proposed which allow more than one station to transmit simultaneously. However, these schemes are still unable to maximize the utilization of radio resources of the IEEE 802.16j networks. Hence, we propose an efficient scheduling scheme, which exploit the full potential of multi-hop relay systems in IEEE 802.16j networks by maximizing the radio resources utilization.

The rest of the paper is organized as follows. Section 2 introduces some related work. Our resource reuse scheduling scheme has been described in Section 3. Simulation results are presented in Section 4 and Section 5 concludes this paper.
2. Related work

Various researchers have devoted their efforts towards improving the performance of IEEE 802.16j networks. Some literatures have proposed basic frame structures and scheduling algorithms for IEEE 802.16j multi-hop relay networks, while others have focused on deployment of relay nodes in the service region to extend the coverage and enhance the capacity of network.

In Lin et al. [6], a heuristic algorithm has been proposed for the relay node placement under the assumption of dual-relay mode in an IEEE 802.16j network with uniformly distributed traffic. In Lu and Liao [7], they have given an integer liner programming (ILP) formulation for joint BS and RS deployment under budget and coverage constraint for IEEE 802.16j multi-hop relay networks. They have proposed a two-stage network deployment scheme, which considers both issues of capacity enhancement and user fairness. The work in Li and Jin [8] proposes a RS selection algorithm for multi-hop relay network based on QoS, which improves the system performance in term of signal-to-noise-rate and latency.

A frame structure for IEEE 802.16j was designed in Tao et al. [9]. They propose a simple framework based upon the current IEEE 802.16e OFDMA frame structure design, which enables multi-hop operation while still maintaining the backward compatibility with the legacy mobile stations. This structure consists of an access zone and a relay zone. The access zone is used for transmissions between the SS and the RS. The relay zone is used for transmissions between the RS and the BS. By scheduling different types of transmissions in different zones, frequency reuse can be achieved.

In Tao et al. [10], aggregation and concatenation in IEEE 802.16j were introduced. All the packets that are being relayed through the same RS are gathered and transmitted simultaneously. After receiving the packets, the RSs identify their destinations and relay them to the correct destination. A new element was added into the frame structure to identify whether the connection was aggregated or not.

Many scheduling schemes [11–17] have been proposed for IEEE 802.16j networks. In Lin et al. [15], resource scheduling is focused on networks where both the BS and the RSs are equipped with directional antennas. System throughput can be dramatically increased by the use of directional antennas, but the cost of implementation also increases with this approach. In Nie et al. [16], they have proposed a bandwidth allocation method for polling service in IEEE 802.16j networks. This method is adaptive to the traffic pattern to provide bandwidth efficiency over access and relay links. Further, the bandwidth requests of the SSs are aggregated at the RS to save bandwidth. However, the method fails to consider the bandwidth allocation for non-polling service in IEEE 802.16j networks. The work in Bayan and Wan [11] proposes a scalable QoS scheduling architecture for IEEE 802.16j networks to guarantee the QoS demands of different applications. In Ismael et al. [13], they propose a scheduling scheme for IEEE 802.16j multi-hop relay networks to reduce the bandwidth request delay. In this scheme, the BS estimates the bandwidth demand of its subordinate RSs using grey prediction algorithm and allocates the bandwidth accordingly. Thus, it eliminates the overhead caused by the standard bandwidth request procedure. In Wang and Jia [17], they have proposed a scheduling scheme for both uplink as well as downlink traffic using the control theory method, which is implemented in the RSs to achieve the desired end to end flow control and congestion control in IEEE 802.16j networks. A scheduling algorithm for IEEE 802.16j networks, which adaptively selects the relay and access zone boundaries based on traffic demand and link condition has been proposed in Ghosh et al. [12].

In Izumikawa et al. [14], a MAP-multiplexing approach for scheduling in IEEE 802.16j multi-hop relay networks has been proposed. As shown in Fig. 3, the above approach affixes an end time (ET) element to each MAP IE in order to split those connections that need more than one timeslot for transmission. For example, if a connection needed two continuous timeslots for transmission, then by using the ET element, the connection can be split and scheduled in two separate timeslots. In addition, this approach reduces the number of connections to transmit simultaneously. However, it does not consider scheduling of lower
priority connections when higher priority connections are interfered by other scheduled links. As a result, connection with lower QoS may suffer a huge delay if higher QoS connections occupy many time slots.

3. Resource reuse scheduling

In this section, we will illustrate how our algorithm works to improve the efficiency of 802.16j networks. As mentioned in Izumikawa et al. [14], we assume that a BS arranges all the transmissions for the SSs including those SSs that are under RSs. In the beginning, a SS-management table is constructed by the BS to catalog the possible interfering stations for all connections. Each SS in the network transmits a list of interference stations to the BS by ranging request (RNG-REQ) packet. The BS records this information in the SS-management table. SS-management table also includes the SS-identification and point of attachment of the SS. While scheduling the transmissions for the SSs, the BS refers to this table in order to avoid interference between connections transmitting in the same time slot.

Fig. 4(a) gives an example of IEEE 802.16j network topology which is composed of one BS, four RSs, and four SSs and Fig. 4(b) shows the SS-management table for the network. From the SS-management table, the BS can determine the interfering transmissions and avoid scheduling them in the same timeslot. For example, when a transmission from BS to SS3 has been scheduled in a particular timeslot, a transmission from RS1 cannot be scheduled in the same timeslot because of possible interference at SS3. This is because of the fact that SS3 is in the transmission range of both BS and RS1 and so simultaneous transmission from BS and SS3 will cause interference at the receiver SS3. The BS examines if the target connection is having interference with the already scheduled connections by checking the SS-management table. If there is no interference, the target connection can be scheduled in the same time, otherwise not. The maximum number of entries in the SS management table will be equal to the total number of sender–receiver pairs in the network.

![Fig. 4. (a) An example of IEEE 802.16j network topology. (b) A SS-management table for the network.](image-url)
Our resource reuse scheduling (RRS) algorithm is given in Fig. 5. The \textit{active\_connection\_set} is the set of connections for which all the packets have not reached to their corresponding SS. \(C_1\) is the set of active connections for which BS acts as a sender while \(C_2\) is the set of active connections whose sender is any RS. \textit{check\_interference} function uses SS-management table to check for possible interference between an already scheduled connection and the target connection. In each timeslot, our algorithm first schedules the highest priority connection from the set \(C_1\). The priority of a connection is decided by its QoS requirements. After scheduling the highest priority connection from \(C_1\), we select the highest priority connection from \(C_2\) as the target connection and using SS-management table we check for the possible interference between the target connection and already scheduled connections. If there is no interference between the target connection and already scheduled connections, the target connection can be scheduled in the same timeslot. On the other hand, if interference exists between the target connection and scheduled connections then the target connection cannot be scheduled in the current timeslot. In both the cases, the target connection is removed from the set \(C_2\) and the next high priority connection in \(C_2\) is chosen as the target connection. This procedure is repeated until there is no connection left in \(C_2\). This procedure is repeated until all the packets have been reached to their destination. IEEE 802.16 MAC specifies four different scheduling services in order to meet the QoS requirements of multimedia applications: unsolicited grant service (UGS), real-time polling service (rtPS), non-real-time polling service (nrtPS), and best effort (BE). The QoS priority of a connection is based on the different scheduling services.

We will now illustrate our resource reuse scheduling algorithm using the example shown in Fig. 4(a). There are four connections and the order of their QoS priority is BS–RS1–SS1 > BS–RS2–SS2 > BS–RS3–RS4–SS4 > BS–SS3. As already discussed, for each timeslot, our algorithm will first schedule the highest priority transmission involving the BS. For timeslot t1, the connection set \(C_2\) will be empty and \(C_1\) will have four active connections (BS–RS1, BS–RS2, BS–RS3, BS–SS3), so only one connection BS–RS1 having highest priority is scheduled. In timeslot t2, the connection RS1–SS1 will be added to \(C_2\) and \(C_1\) will have three active connections (BS–RS2, BS–RS3, and BS–SS3). Our algorithm will again select BS–RS1, the connection set \(C_2\) will be empty and \(C_1\) will have four active connections (BS–RS1, BS–RS2, BS–RS3, BS–SS3), so only one connection BS–RS1 having highest priority is scheduled. In timeslot t3, the connection RS1–SS1 will be added to \(C_2\) and \(C_1\) will have three active connections (BS–RS2, BS–RS3, and BS–SS3). Our algorithm will again select BS–RS1, the connection set \(C_2\) will be empty and \(C_1\) will have four active connections (BS–RS1, BS–RS2, BS–RS3, BS–SS3), so only one connection BS–RS1 having highest priority is scheduled. In timeslot t4, the connection RS1–SS1 and RS2–SS2 will be added to \(C_2\) and \(C_1\) will have three active connections (BS–RS2, BS–RS3, and BS–SS3). Our algorithm will select BS–RS1, the connection set \(C_2\) will be empty and \(C_1\) will have four active connections (BS–RS1, BS–RS2, BS–RS3, BS–SS3), so only one connection BS–RS1 having highest priority is scheduled. In timeslot t5, the second RS1–SS1 transmission is selected as the target connection and because of possible interference with BS–SS3, it

\begin{algorithm}
\caption{Resource reuse scheduling (RRS)}
1: set timeslot \(\tau = 1\)
2: \textbf{while} some packets have not reached their destination \textbf{do}
3: \hspace{1em} set the connection sets \(C_1 = \emptyset\) and \(C_2 = \emptyset\)
4: \hspace{1em} \textbf{for} all connection \(i \in \text{active\_connection\_set} \textbf{do}
5: \hspace{2em} \textbf{if} sender(\(i\)) = BS \textbf{then}
6: \hspace{3em} \(C_1 = C_1 \cup \{i\}\)
7: \hspace{2em} \textbf{end if}
8: \hspace{1em} \textbf{end for}
9: \hspace{1em} \(C_2 = \text{active\_connection\_set} - C_1\)
10: \hspace{1em} \textbf{if} \(C_1 \neq \emptyset\) \textbf{then}
11: \hspace{2em} set target connection \(t = \text{highest\_priority\_connection}(C_1)\) and schedule \(t\) in timeslot \(\tau\)
12: \hspace{1em} \textbf{end if}
13: \hspace{1em} \textbf{while} \(C_2 \neq \emptyset\) \textbf{do}
14: \hspace{2em} set target connection \(t = \text{highest\_priority\_connection}(C_2)\)
15: \hspace{2em} \textbf{if} \text{check\_interference}(t) \textbf{then}
16: \hspace{3em} schedule \(t\) in timeslot \(\tau\)
17: \hspace{2em} \textbf{end if}
18: \hspace{2em} \(C_2 = C_2 - \{t\}\)
19: \hspace{1em} \textbf{end while}
20: \hspace{1em} do the transmission for all the scheduled connection in timeslot \(\tau\)
21: \hspace{1em} \(\tau = \tau + 1\)
22: \textbf{end while}
\end{algorithm}

Fig. 5. Resource reuse scheduling algorithm.
cannot be scheduled in t5. Next, we choose the lower priority connections RS2–SS2 and RS2–SS4, respectively as target connection and check for possible interference. In Izumikawa et al. [14], they do not consider scheduling of lower priority connections, when a higher priority connection cannot be scheduled in a particular timeslot. If it is used, we have to wait until all BS–SS3 transmission is over and after that we can schedule the RS1–SS1 transmission in t8. However, in our algorithm, we check the lower priority connections RS2–SS2 and RS3–SS4 for scheduling, even if the higher priority connection RS1–SS1 was not scheduled.

As shown in Fig. 6, the scheduling algorithms given in [4] and [14] require 12 and 10 timeslots, respectively for scheduling of all the connections. Whereas, our resource reuse scheduling requires only 8 time slots for scheduling of all the connections. As a result, the length of the schedule is reduced and our algorithm can achieve higher radio resource reuse and higher system throughput. Furthermore, connections with lower priority (ex: RS2–SS2, RS3–RS4, RS4–SS4) can be scheduled even if

![Scheduling Example](image_url)

**Fig. 6.** (a) Scheduling example without MAP-multiplexing [4]; (b) scheduling example with MAP-multiplexing [14]; (c) scheduling example for resource reuse scheduling algorithm.
higher priority connections (ex: RS1–SS1) weren’t scheduled successfully. Therefore, the delay of lower priority connections is reduced significantly without increasing the delay of higher priority connections.

Fig. 7. The star-shape scenario.

Fig. 8. The delay of star-shape scenario by the connection.

Fig. 9. The delay of star-shape scenario by the priority.
4. Simulation results

We performed simulation to measure the performance of our resource reuse scheduling scheme (RRS) by comparing it with MAP-multiplexing [14]. All the simulation results were obtained by running these algorithms for 50 iterations and then taking the average. We have taken four scenarios, namely, star-shape scenario, tree-shape scenario, mesh-shape scenario, and a random case scenario into consideration for simulation. In the star-shape and tree-shape scenarios, the priority of transmission of every SS is in a random order for each iteration. Each SS varies its transmission slots from 1 to 8. Similarly in mesh-shape and random scenarios, every SS has a random priority of transmission for each iteration, and transmission slots vary from 1 to 8, 1 to 16, 1 to 24 or 1 to 32, as four different cases.

Fig. 10. The delay of star-shape scenario by number of transmission slots.

Fig. 11. The frame utilization of star-shape scenario by number of transmission slots.

Fig. 12. The tree-shape scenario.
The star-shape scenario is shown in Fig. 7. BS is placed at center of the service area, RSs and SSs have been deployed symmetrically around BS such that each RS is in the transmission range of BS and each SS is in transmission range of only one RS. In this scenario, five RSs and five SSs were allocated. Every SS randomizes its transmission slots and priority of transmission. SS2 and SS5 are interfered with RS1 since they are located in the transmission range of RS1. SS1 and SS3 are also interfered.

**Fig. 13.** The delay of tree-shape scenario by the connection.

**Fig. 14.** The delay of tree-shape scenario by the priority.

**Fig. 15.** The delay of tree-shape scenario by number of transmission slots.
with RS2 for the same reason and so on. Fig. 8 illustrates the delay of this scenario for each SS and the total delay. Result shows that there is no obvious difference between connections, which means our algorithm can keep the fairness of the network. Fig. 9 illustrates the delay of star-shape scenario by priority. Result shows that our algorithm improves the delay for those connections having lower priority. The reason of this improvement is that our algorithm will consider the transmission of lower priority connections earlier without influencing the transmission of higher priority connections. Fig. 10 illustrates the total delay when the number of time slots for each SS randomize from 1 to 8, 1 to 16, 1 to 24 or 1 to 32 as four different cases. As the number of random time slots for each SS increases, the number of time slots for higher QoS connections also increases. If higher QoS connections cannot be scheduled due to interference then the delay for lower QoS connections will increase in MAP algorithm. But, our algorithm tries to schedule those lower QoS connections which are ready to send the data when higher QoS connections are waiting to complete their transmission due to interference. So in MAP-multiplexing algorithm average delay for lower QoS connections increases while in our algorithm delay for lower QoS connections do not increase because our algorithm tries to schedule the lower QoS connections when the higher QoS connections cannot be scheduled. Fig. 11 illustrates the frame utilization i.e. number of data units sent in one time slot. It shows that our algorithm
has better frame utilization than the MAP algorithm because our algorithm tries to accommodate the lower QoS connections without waiting for higher QoS connections to complete their transmission. Hence our algorithm can schedule more frames in one time slot as compared to the MAP algorithm. The frame utilization graph shows that, as the number of slots increases

Fig. 18. The delay of mesh scenario by number of transmission slots.

Fig. 19. The frame utilization of mesh scenario by number of transmission slots.

Fig. 20. The random scenario of two layers.
the difference between the frame utilization of two algorithms is increasing. The above reason of frame utilization graph can be explained in the same way as in the case of average delay.

The tree-shape scenario is shown in Fig. 12. Two RSs and four SSs were allocated in this case. BS is kept at the center of service area, RSs and SSs have been placed such that SS2 is interfered with RS2 as it is located in the transmission range of RS2. SS3 is interfered with RS1 for the same reason. Delay for this case is illustrated in Figs. 13–15. Fig. 16 shows the frame utilization and in the case of tree shape scenario also, our algorithm improves the delay and frame utilization as compared to the MAP-multiplexing algorithm.

The mesh scenario is shown in Fig. 17. BS is deployed at the center of service area, 4 RSs and 20 SSs have been deployed symmetrically around the BS such that each SS is either in transmission range of BS or RS. In this scenario we have 4 RSs and 20 SSs. SS1, SS2, SS3, SS4 are directly connected to the BS, while other stations SSij (i = 1, 2, 3, 4 and j = 1, 2, 3, 4) are connected to the RSi (i = 1, 2, 3, 4). Every SS randomizes their transmission slots and priority. Each SS randomizes their transmission slots from 1 to 8, 1 to 16, 1 to 24 or 1 to 32 as four different cases. RS1 and RS4 stations act as interfering stations during transmission from BS to SS1. Similar cases can be seen with BS to SS2, SS3 and SS4. While during the transmission from RS1 to SS14, RS4 acts as an interfering station, and with the same generality RS1, RS2, RS3 acts as an interfering station for RS2 to SS24, RS3 to SS34 and RS4 to SS44, respectively.

![Fig. 21. The random scenario of three layers.](image1)

![Fig. 22. The delay of two layer random scenario by number of transmission slots.](image2)
In Fig. 18, we can see that as the number of random time slots for each SS increases our algorithm improves the average delay of SSs. Fig. 19 shows that our algorithm has better frame utilization as compared to MAP algorithm. In mesh network as shown in Fig. 17 and 20 SSs are handled by 4 RSs so the cost for implementing the RSs can be reduced, and our algorithm reduces the average delay for the mesh network as compared to the MAP algorithm. Even under compact scenario like mesh (4 RS and 1BS covering 20 SSs) our algorithm is reducing the delay to some extent.

Furthermore, we have designed a random case shown in Fig. 20 and Fig. 21 by referencing the deploy scheme in [18] and [19]. Fig. 20 is a two layers transmission and Fig. 21 is a three layers transmission scenario. The scenario shown in Fig. 20 is random distribution of the SSs and the RSs where the BS is kept fixed. The position of SSs and RSs are generated randomly. The number of SSs directly in the range of the BS is kept to 30. The range of BS and RSs is 30 m. 50 SSs are generated in the next layer in between the circles whose radius is 30–50 m from BS so that most of SSs can be connected through the BS via RSs. But, still in every case there are some SSs that do not come in the range of the RSs, so the BS cannot send data to these SSs. The 50 RSs are generated on the circumference of the ranging circle of the BS. The RSs which have no SS in its range of transmission do not function. Similarly, RSs which have same list of connected SS as in some other RS has also made to stop function. This should optimize the number of RSs in some way. After this, the interference tables are generated and scheduling is done.

In Fig. 21, we have introduced a new layer in which the 70 RSs are generated just inside the border of first layer of RSs (circle of radius 58 m form BS) which connect the new SSs in the outer layer to the BS. The new 120 SSs are generated in between the circle of radius 60 to 70 m so that some of them can be connected to outer RSs, but still most of the SSs do not come in the range of RSs. Here too the same optimization is applied and even those outer RSs are made to stop function which does not come in range of any inner RS. After this, similar process of finding interference table is done and result is found. Fig. 22 illustrates the delay for layer 2 and Fig. 24 shows the delay for the layer 3 while Figs. 23 and 25 show the frame utilization for layer 2 and 3, respectively. Results show that even in random scenario our algorithm improves the delay and frame utilization as compared to the MAP algorithm.

![Fig. 23. The frame utilization of two layers random scenario by number of transmission slots.](image)

![Fig. 24. The delay of three layers random scenario by number of transmission slots.](image)
5. Conclusion

In this paper, we have proposed a resource reuse scheduling algorithm for multi-hop relay system in IEEE 802.16j networks. Our algorithm is based on MAP-multiplexing, which can schedule multiple connections in a single timeslot to increase the frame utilization and exploit the benefits of multi-hop relay networks. We consider scheduling of lower priority connections even if higher priority connections are interfered by already scheduled connections. As a result, our algorithm decreases the average delay and increases the frame utilization, and thus enhances the performance of multi-hop relay networks. In simulation part, we have compared our algorithm with the MAP-multiplexing for regular deployment scenarios like star-shape, tree-shape and mesh-shape as well as for random deployment scenario. Performance evaluation shows that our algorithm performs better than the MAP-multiplexing scheme in terms of average delay and frame utilization in all of the above deployment scenarios. Our scheduling algorithm reduces the average delay by 5–10% as compared to the MAP-multiplexing algorithm.

Acknowledgements

This work was supported in part by the National Science Council, Taiwan, under Grant NSC 99-2221-E-036-036-MY2 and Tatung University, under Grant B99-N03-065.

References


Wen-Hwa Liao received the Ph.D. degree in computer science and information engineering from National Central University, Taiwan, in 2002. Currently, he is an Associate Professor in the Department of Information Management, Tatung University, Taiwan. His current research interests include wireless sensor networks and mobile communications. He is an associate editor of the International Journal of Distributed Sensor Networks.

Kuei-Ping Shih received the Ph.D. degree in computer science and information engineering from the National Central University, Taiwan, in 1998. Currently, he is a Professor in the Department of Computer Science and Information Engineering, Tamkang University, Taiwan. Dr. Shih’s current research interests include network protocol designs for wireless LANs, MANs, wireless sensor networks, and mobile ad hoc networks.

Chen Liu received the M.S. degree in information management from Tatung University, Taiwan, in 2009. Her current research interests include wireless sensor networks and mobile ad hoc networks.

Avinash Kumar Dubey received his B.S. degree in Computer Science and Engineering from Indian School of Mines University, Dhanbad, India, in 2009. Currently he is working in wireless protocol group of Samsung India Software Operations. His research interests include wireless networks and mobile communication. His current work focuses on implementation of different protocol layers form mobile communication.

Sagar Arora received his B.S. Degree in Computer Science and Engineering from Indian School of Mines, Dhanbad, India, in 2009. Currently he is working in RF group of national instruments (R&D) Bangalore. His research interests include wireless networks and his current work focuses on RF domain of various wireless technologies.

Sital Prasad Kedia received his B.S. degree in Computer Science and Engineering from Indian School of Mines University, Dhanbad, India, in 2009. Currently he is working in wireless protocol group of Samsung India Software Operations. His research interests include wireless sensor networks, mobile ad hoc networks, distributed and embedded systems.