QoS Aware Wireless Bandwidth Aggregation (QAWBA) by Integrating Cellular and Ad-hoc Networks

Danyu Zhu, Matt W. Mutka, and Zhiwei Cen
Department of Computer Science and Engineering
Michigan State University
East Lansing, MI 48824
Email: {zhudanyu,mutka,cenzhiwe}@cse.msu.edu

Abstract

Some mobile devices are beginning to support both cellular and IEEE 802.11 based network interfaces. Although rates are increasing, current cellular networks provide relatively low bandwidth that do not meet the QoS requirements of many high-demanding multimedia applications. In this paper, we propose an integrated network architecture that utilizes both wireless interfaces to provide better QoS support by QoS Aware Wireless Bandwidth Aggregation (QAWBA). Via QAWBA, mobile nodes form a mobile ad hoc network (MANET) using their IEEE 802.11 interfaces to share their cellular link capacity. Some mobile nodes act as proxies to contribute their idle cellular links to support a QoS request that may exceed the available bandwidth of any individual mobile node. A K-path proxy discovery algorithm is proposed for fast and efficient proxy discovery. Simulation results show that QAWBA can significantly improve network utilization and the admission rate of QoS requests.

1. Introduction

Wireless networks will be an integral part of the global communication architecture. Eventually, wireless users may demand the same Quality of Service (QoS) for applications that are currently available on today's wired networks. Two different kinds of wireless networks are widely available: the cellular network and the IEEE 802.11 based network.

The cellular network provides relatively low throughput and cannot meet the bandwidth requirements of many multimedia applications. The latest commercial deployment of 1xEV-DO offers only 38.6Kbps to 2.4Mbps depending on the signal strength, while the IEEE 802.11b standard can provide 1-11Mbps and the IEEE 802.11a/g standards can provide up to 54Mbps. However, the IEEE 802.11 based network can cover very limited areas, while the network access provided by the cellular network is virtually "anytime, anywhere."

With the popularity of both kinds of wireless networks, we envision the scenario in which mobile devices are equipped with both wireless interfaces and have the ability to access cellular based and IEEE 802.11 based networks at the same time. In this paper, we present a novel integrated network architecture for *QoS Aware Wireless Bandwidth Aggregation (QAWBA)* that utilizes the cellular interface and the IEEE 802.11 based interface to take the advantage of both networks, to meet the high availability and high bandwidth QoS requirements at the same time.

The basic idea of QAWBA is that cooperating mobile nodes form a mobile ad hoc network (MANET) using the IEEE 802.11 based interface operating in an ad hoc mode in order to share their cellular network connections. Several low bandwidth cellular network connections are aggregated to meet the high bandwidth QoS requirement of multimedia applications for which a single cellular connection is insufficient. For a mobile node (*client*) that requires bandwidth higher than its own available cellular capacity, several other nodes in the same MANET may function as "*proxies*" by contributing their idle cellular connections to the client. The traffic is forwarded by the proxies to the client in the MANET via IEEE 802.11 interfaces.

Since the cellular and the IEEE 802.11 networks utilize different frequencies, the traffic on the two networks will not interfere with each other. For a single node, as many proxies can be used as the IEEE 802.11 based network is able to support. Thus with QAWBA, it is possible to provide similar high bandwidth as the IEEE 802.11 based network while keeping the high availability of the cellular network.

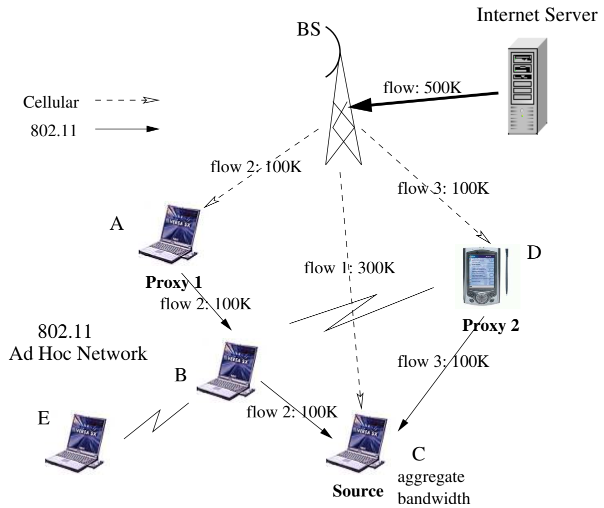


Figure 1. Example of QAWBA

Figure 1 shows an example of QAWBA, in which five mobile nodes form a MANET. The client node C executes an application requiring 500Kbps bandwidth, which can obtain only 300Kbps from its cellular link. A and D act as proxies to forward a portion of the total traffic to C. The 500Kbps traffic flow is split into three flows in the base station, and forwarded to C via different paths. Thus, with the help of nodes A and D, C is able to receive the required 500Kbps bandwidth by aggregating three flows, which would not be possible under one single cellular connection.

QAWBA requires a new QoS aware routing protocol for the MANET, which is much different from other existing QoS routing protocols. Typical QoS routing protocols in a MANET find a path or multi-path from the source to the destination that meets certain QoS requirements (bandwidth and delay) [1, 13]. The source and destination are known before the QoS path discovery procedure. The QoS request is initiated by the source node. However, in QAWBA, only the destination node (client) is known at the beginning of the route setup. The sources, which are proxies, should be discovered by the routing protocol. The QoS request is initiated by the destination node. To the best of our knowledge, none of the existing routing protocol can be used to solve this problem.

To support QAWBA, we present a K-path on-demand QoS aware proxy discovery protocol to find suitable proxies in the MANET based on the bandwidth requirement and maximum hop limitation. Proxies are discovered along K paths starting from the client. Only K messages are needed for each session. The cellular bandwidth is reserved progressively among the nodes in the path.

The rest of the paper is organized as follows. Section 2 presents related work regarding integrated network and QoS routing in the MANET. QoS aware on-demand routing protocol is presented in section 3. Performance evaluation and results are in section 4. Section 5 concludes the paper.

2. Related Work

Much research has been dedicated to solve Quality of Service issues for MANETs [12, 8]. INSIGNIA [4] is an effort to design a cross-layer framework to support QoS routing in ad hoc networks. INSIGNIA uses an in-band and soft-state based signaling protocol to support fast reservation, restoration and end-to-end adaptation of QoS parameters.

Several protocols have been proposed to address on the QoS aware routing in MANETs. The CEDAR algorithm [11] uses a set of ad hoc nodes called the *core* to establish a QoS aware route from the source to the destination. Information regarding the availability of bandwidth propagates among core nodes using a link state protocol. In AQDR [13], the source uses limited flooding for route establishment. The destination of the route is responsible for QoS violation detection and the destination-initiated recovery process begins when a QoS violation is detected.

Ticket based probing [1] is one of the flooding based QoS routing discovery algorithms. It assumes an imprecise state model and tries to reduce the amount of flooding routing messages by issuing logical tickets. When a probe arrives at a node, the tickets contained in the probe can be split to its neighbors. When one or more probes arrive at the destination node, the routing path is known and the networking information can be used to establish a quality aware path. The ticket based routing is again extended by Liao, et al. to find a multi-path QoS routing scheme between the source and the destination [5].

There has been some work in the area of integrating ad hoc and cellular data network [6, 2]. In [7], mobile users have both 3G cellular links and IEEE 802.11 based ad hoc links, forming an UCAN architecture. The base station forwards packets for clients with poor channel quality to proxy clients with better channel quality. The packets are further routed to the destination client via IEEE 802.11 based ad hoc link. UCAN is designed specifically for the 1xEV-DO (HDR) 3G cellular network, which limits its application. Unlike QAWBA, UCAN can only utilize one proxy for a client with poor link quality. The maximum bandwidth is limited by the cellular link capacity, which generally cannot meet the requirement of high-demanding multimedia applications. Also, UCAN is a best-effort approach. A client searches for a proxy with better cellular bandwidth. It does not provides any QoS guarantee.

The iCAR [10] architecture addresses two problems for current cellular networks: (1) the network capacity is limited by the cell boundary; (2) bursty traffic is unevenly distributed among cells. iCAR places ad hoc relay stations at strategic locations to relay signals between mobile hosts and base stations. Bursty traffic could be diverted from one congested cell to another one in order to circumvent congestion. Although iCAR can efficiently balance traffic between different cells, it has some limitations. First, special kinds of relay stations other than the base station and the mobile hosts must be placed by a network operator for packet relaying. However most of current mobile hosts have the ability for relaying packets via their IEEE 802.11 based network interfaces. Second, iCAR is useful for diverting bursty traffic to nearby idle cells. It does not provide a way to improve utilization of the idle cellular link under light traffic load. Again, the maximum bandwidth is limited by the bandwidth of one single cellular link.

CHUM has been presented as an approach to integrate 3G networks and ad hoc networks in a manner that significantly reduces 3G network costs to provide support for *instant messaging (IM)* [14] and to download multimedia data, such as on-air TV programs [3].

3. QoS Aware On-demand Routing

To provide QoS, QAWBA should integrate on-demand proxy discovery, bandwidth reservation and maintenance, and hop-by-hop routing. On-demand proxy discovery is provided by a K-path QoS aware proxy discovery algorithm. For each QoS session, K QoS requests are transmitted along different paths, to search for suitable proxies. When a mobile node receives a QoS request, it determines the amount of cellular bandwidth to reserve for this session and then sends back a QoS reply for the reservation along the reverse path. The request is further forwarded to a neighbor, if needed. The total amount of bandwidth requested within K requests is the amount of bandwidth required for the session minus the bandwidth provided by the client itself. The reservation is successful if enough bandwidth is reserved within the time interval T_{setup}. Otherwise, the reservation request fails.

The overhead of connection maintenance and teardown is eliminated by the soft-state reservation mechanism in which continuous packets from the base station destined for the client serve the purpose to provide the reservation update signals. An extended AODV [9] routing protocol is used for hop-by-hop routing in the MANET, using the QoS request (QRREQ) as route request (RREQ) and the QoS reply (QRREP) as route reply (RREP). The routing table is set up along the path from the client to the proxies in the process of proxy discovery.

3.1. Neighborhood Maintenance

Periodic “HELLO” messages are used by the mobile node to obtain the neighborhood information to conduct proxy discovery and traffic admission control. A node I includes the bandwidth available in its cellular network, in the MANET, and the consumed bandwidth in the MANET in the HELLO message. Every node maintains a neighborhood table composed of the information obtained from the HELLO messages. A failure to receive a packet from a neighbor for a T_{nb} period means the link to that neighbor is broken.

3.2. Bandwidth Reservation Tables

The following three reservation tables are kept within each mobile node to store traffic and bandwidth allocations for the cellular network and MANET. This information is used to compute the available bandwidth for the cellular network and the MANET.

- *QoS session table*: For each QoS session, it records the session id (SID), total bandwidth required, maximum hop limitation in the MANET, total bandwidth currently reserved, status and a list of proxies contributing cellular bandwidth for this session (the client is considered to be a special proxy when providing cellular bandwidth for the session). The status of a QoS session could be REQUEST or RESERVED, which represents waiting for a QoS reply or successfully reserved.

- *Cellular flow reservation table*: For each cellular traffic flow, it records the session id, the client node id, the number of hops to the client node and the reserved bandwidth. A cellular reservation entry is inserted into the table when a cellular link reservation is made for a QoS session.

- *MANET flow reservation table*: It stores the session id, the source (proxy), the destination (client) node id, and the reserved bandwidth for each MANET flow. A reservation entry is inserted into this table when a QoS reply is received in the mobile node.

3.3. K-path proxy discovery algorithm

Generally, MANET routing protocols use flooding based discovery algorithms to find a path from the source to the destination, which are not suitable for the proxy discovery in QAWBA. The client only has local topology and traffic information, which is obtained from the periodic "HELLO" messages. Therefore, the number of possible proxies and their available cellular and MANET bandwidth are unknown for the client in the beginning of the discovery process. It is difficult for the node to determine the amount of reservation for a QoS session when receiving a QoS request.

In QAWBA, proxy discovery is done on-demand by a K-path discovery algorithm. An entry will be inserted into the QoS session table for a new QoS session S with the bandwidth requirement, $B_{req}(S)$, and the maximum hop limitation, $MHop(S)$. The status of the session is set to "REQUEST", indicating that it is in the process of proxy discovery. The client X then decides the bandwidth reserved in its own cellular link for this session. If the available cellular bandwidth $B_{avail}(X, c)$ is greater than $B_{req}(S)$, the reserved bandwidth in the cellular link will be the requested bandwidth: $B_{res}(X, S, c) = B_{req}(S)$. Otherwise, X reserves all the available bandwidth for S: $B_{res}(X, S, c) = B_{avail}(X, c)$. A new entry will be inserted into the cellular flow reservation table indicating the new cellular flow reserved for the session S in the client X with bandwidth $B_{res}(X, S, c)$.

If the bandwidth requirement is fulfilled by the client's cellular link, the reservation process is finished and the status of the session is changed to "RESERVED". The application will be notified of the successful reservation. Otherwise, one or more proxies should be discovered to meet the remaining bandwidth requirement of this QoS session. K QoS request (QRREQ) messages are generated by the client. Each QoS request carries with part of the remaining bandwidth requirement, $\frac{B_{req}(S)-B_{res}(X,S,c)}{K}$, and is sent to one of the neighbors with the highest available cellular bandwidth.

When mobile node I receives a QRREQ message for a QoS session S, with a request for bandwidth $B_{qrreq}(S)$, it rejects this QoS request if the current available MANET bandwidth is smaller than the consumed MANET bandwidth for the session S. A QoS failure (QFAIL) message is then sent back to the client. Otherwise, I determines the amount of cellular bandwidth it may reserve for S based on $B_{qrreq}(S)$ and the current available cellular bandwidth, $B_{avail}(I, c)$. Similar to the reservation procedure in the client, the reserved bandwidth is the minimum of the available cellular bandwidth and the requested bandwidth. $B_{res}(I, S, c) = min\{B_{qrreq}(S), B_{avail}(I, c)\}$

QoSAwareProxyDiscovery(sid, bw, max_hop)

```
K=dynamicK(bw)  // get K value based on request bw
if (k<0) then
  return notify_app(sid, FAIL) // fail
endif
if (k==0) then
  reserved_bw=bw  // enough cellular bw
else
  reserved_bw=cellular_availbw  // not enough cellular bw
endif
request_bw=bw - reserved_bw
if (request_bw>manet_availbw) then
  return notify_app(sid, FAIL)  // not enough MANET bw
endif
// insert a new QoS Session entry
QoSSession=insert_QSession(sid, bw,
     max_hop, reserved_bw, REQUEST)
if (reserved_bw>0) then
  cflow=insert_cflow(sid, reserved_bw, node_id, 0)
  QoSSession.insert_proxy(sid, node_id, reserved_bw)
endif
if (request_bw==0) then
  QoSSession.status=RESERVED
  return notify_app(sid, RESERVED)  // successful
endif
// send K QoS requests to neighbors
sendQoSRequest(node_id, K, sid, request_bw/K, max_hop)
return notify_app(sid, REQUEST)  // wait for reply
```

Figure 2. QoS Aware Proxy Discovery Algorithm

If $B_{res}(I, S, c) > 0$, a new entry is inserted into the cellular flow table in the mobile node I and I is considered as one of the proxies for session S. A QoS reply (QRREP) message is sent back to the client for the new reservation in I's cellular link. If I cannot provide enough cellular bandwidth for session S, the QRREQ request will be forwarded to one of the neighbors with an updated bandwidth request: $B'_{qrreq}(S) = B_{qrreq}(S) - B_{res}(I, S, c)$. I sends back a QFAIL message to the client if the maximum hop limitation is reached or there is no available neighbor for the QRREQ request. The QRREQ request stops propagating if the bandwidth request is satisfied ($B'_{qrreq}(S) = 0$).

When the client X receives a QRREP reply message, it adds the amount of reserved bandwidth in the QRREP message to the reserved bandwidth field for the corresponding session entry in the session table. If the reserved bandwidth is equal to the requested bandwidth for session S, the reservation for S is successful. When the client X receives a QFAIL message with the amount of unreserved bandwidth $B_{unres}(S)$ for session S, it may choose a new neighbor for the retry of QRREQ requesting for $B_{unres}(S)$. If not enough bandwidth is reserved for the session S within the T_{setup} interval, the reservation has failed.

OnRecvQoSRequest(client, hop_count, sid, bw, max_hop)

```
if (bw>manet_availbw) then
  sendQoSFail(client,node_id,sid)
endif
if (cellular_availbw>bw) then
  reserved_bw=bw
else
  reserved_bw=cellular_availbw  // reserve all available
endif
if (reserved_bw>0) then
  cflow=insert_cflow(sid,reserved_bw,client,hop_count)
  mflow=insert_mflow(sid,client,node_id,reserved_bw)
  sendQoSReply(client,node_id,sid,reserved_bw)
endif
request_bw=bw - reserved_bw
if (request_bw>0) then
  if (max_hop - 1<=0) then
     sendQoSFail(client,node_id,sid)
  else
    // send 1 request to neighbor
     sendQoSRequest(client,1,sid,request_bw,max_hop-1)
  endif
endif
```

Figure 3. Processing of QoS Request Message

Figure 2 shows details of the K-path discovery algorithm. The processing of QoS request messages and QoS reply messages are presented in figures 3 and 4.

OnRecvQoSReply(client, proxy, sid, reserved_bw, hop_count)

```
mflow=insert_mflow(sid, client, proxy, reserved_bw)
if (client_id==node_id)
   // the reply is for me
  QoSSession=session_lookup(sid)
  QoSSession.reserved=QoSSession.reserved+reserved_bw
  QoSSession.insert_proxy(sid, proxy_id, hop_count)
  if (QoSSession.reserved==QoSSession.required)
     // the bandwidth requirement fulfilled
    QoSSession.statud=RESERVED
    return notify_app(sid, RESERVE_SUCCESS)
  endif
else
  forward_reply()
endif
```

Figure 4. Processing of QoS Reply Message

The value of K could be predetermined by the client or be dynamically computed based on the bandwidth requirement in the QoS session, current available bandwidth and the maximum hop limitation. A smaller K value results in a longer path and higher risk of exceeding the maximum hop limitation. On the other hand, the larger K increases the contention on the client node. Therefore, the optimal K value should be the smallest one that meets the maximum hop limitation.

We compute the value of K based on the assumption that the network load is evenly distributed among all mobile nodes. Although this assumption is not always true, it is adequate for estimating the value of K. K is then computed as the minimum value of the maximum possible value of K, $maxK$, and $\frac{B_{req}(S)}{B_{avail}(X,c)*MHop(S)} + 1$. $MaxK$ is determined by the minimum value of the number of available neighbors N_{nb} and a predetermined parameter $MAXK$: $maxK = min\{N_{nb}, MAXK\}$. $K = min\{maxK, \frac{B_{req}(S)}{B_{avail}(X,c)*MHop(S)} + 1\}$

3.4. Computation of Available Bandwidth

To determine the admission of a QoS session and the amount of cellular bandwidth reservation, we need to know the available bandwidth in the cellular link and the MANET. Here, we assume that the mobile node knows the current link capacity in its cellular interface and MANET interface.

The cellular link could be viewed as a dedicated point-to-point link from node I to the base station. The available bandwidth in the cellular link of I, $B_{avail}(I,c)$, could be computed as the cellular link capacity, $B_{cap}(I,c)$, minus the total cellular bandwidth that has been reserved in the cellular flow table: $B_{avail}(I,c) = B_{cap}(I,c) - \sum_S B_{res}(I,S,c)$.

In the MANET, the radio channel of each node is shared by all neighbors. A node can successfully use the channel only when all its neighbors do not transmit or receive packets at the same time. We use the algorithm in [13] to estimate the upper bound limit of available bandwidth and consumed bandwidth in the MANET. The available bandwidth on the MANET could be computed as the MANET capacity minus the total consumed traffic in I's neighbors $N(I)$. $B_{avail}(I,m) = B_{cap}(I,m) - \sum_{J \in N(I)} B_{consumed}(J,m)$. The consumed bandwidth depends on the location of the node I in the MANET flow. For a MANET flow with bandwidth B, if I is the source or destination, the consumed bandwidth is the flow bandwidth B. Otherwise, the consumed bandwidth is twice the flow bandwidth $(2 \times B)$ since I should receive and send packets in this flow, which cannot be done simultaneously.

When receiving a QoS request with $B_{req}(S)$ requirement for session S, two MANET flows are needed if I could not fulfill this QoS request alone. I is the source of one MANET flow with bandwidth $B_{res}(I,S,c)$ and the intermediate node for the other MANET flow with bandwidth $B_{req}(S) - B_{res}(I,S,c)$. Therefore, the total consumed bandwidth for S,$B_{consumed}(I,S,m)$, in the MANET is: $B_{res}(I,S,m) + 2 \times (B_{req}(S) - B_{res}(I,S,m))$.

3.5. Failure Recovery and Automatic Resource Release

A MANET is characterized by frequent topology change and unreliable physical media. QAWBA needs to provide a mechanism to detect QoS violations and communication failures. A common approach is to use the HELLO message in the routing protocol, in which failure to receive a HELLO message from one of its neighbors within a timeout period indicates its failure. The node then sends a route error message back to the client node. Due to the bandwidth consumption of sending HELLO messages, the frequency of neighbor detection must not be too high. This prevents the system to detect a broken route quickly.

In QAWBA, we use the QoS timeout interval (T_{int}) to detect route breakage at both the client and proxy side. At the client side, if a proxy is not heard by the client for T_{int} (no packets are received from the proxy), the proxy is believed to be out or the route is broken. The client then eliminates the QoS entry of the broken proxy and updates the reserved bandwidth of the current session. The base station is informed about the broken proxy and new arrived packets will not be sent to the client via that proxy. A new round of proxy searching starts if the reserved bandwidth is less than the required bandwidth. At the proxy side, if within a timeout interval it fails to receive any packets from the base station for the client node, the proxy would release the cellular bandwidth reservation and tear down the ad hoc network routes. The intermediate nodes that are along the paths from the proxy to the client node also maintain a timeout timer. If no traffic exchange occurs between the proxy and the client, the reserved ad hoc network routes are also released.

Only when the client node detects a route breakage, the system attempts to recover. Within the recovery process, the client can either unicast one QoS search request or start a new K-path QoS searching session if a single QoS fails. The newly found proxies join the existing session and provide service to the application. If the client fails to find new proxies, the application has to choose either to continue running under a relaxed QoS service or to try again later.

4. Performance Evaluation

4.1. Simulation Model

We implement the QAWBA protocol in the $ns2$ network simulator. Twenty mobile nodes are randomly placed in the area of $300m \times 300m$. Each node has a cellular interface and an IEEE 802.11b interface. The cellular interface is used to communicate with the base station, which in turn connects to the Internet. It uses an one-hop routing scheme, from the base station to the mobile node. Different cellular interfaces use different radio channels and do not interfere with each other. The link capacity of the cellular link is uniformly distributed between 28.8Kbps to 2Mbps.

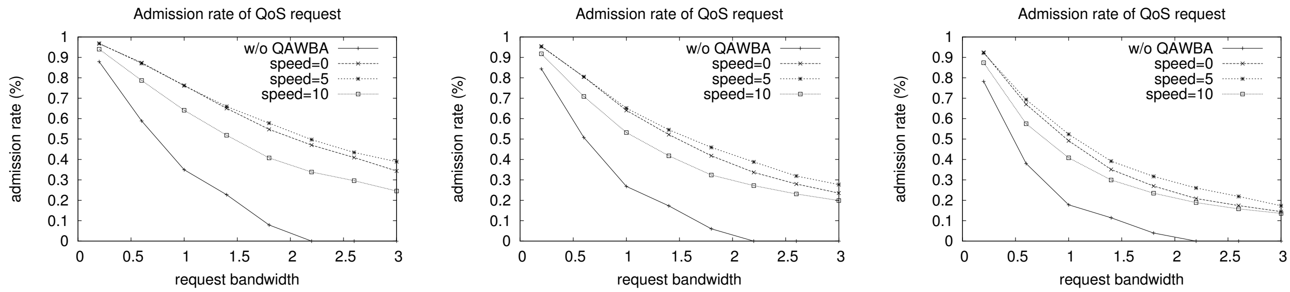
We use the IEEE 802.11b implementation from ns-2 version 2.1b9, where 11Mbps data rate is supported at the 100 meter range. The radio propagation model for IEEE 802.11b uses the two-ray ground reflection model. Node mobility is set according to the random waypoint model. Each node moves toward a random destination within the field at a specified speed. After reaching the destination, the node pauses for a certain amount of time and then starts moving again. A QoS request is generated in the mobile node with a predetermined bandwidth requirement. The length of each QoS request is fixed to be 20 seconds. The time interval between two QoS request is exponentially distributed with mean δ.

Two metrics are evaluated by the simulation: the QoS admission rate (r) and cellular link utilization (u). The QoS admission rate r_I for node I is defined as the ratio of the number of successful QoS requests, $N_{suc}(I)$, and the total number of QoS requests, $N_{total}(I)$. The value r is defined as the average of admission rate of all n mobile node: $r = \frac{1}{n} \times \sum_I \frac{N_{suc}(I)}{N_{total}(I)}$. The link utilization for node I is defined as the total link capacity times the simulation running time divided by total bandwidth in use in this time period for I. Suppose in time period T_j, the bandwidth used by the cellular link of node I is $B_{res}(I, c)$, the total bandwidth in use in the time period T_j is defined as $T_j \times B_{res}(I, c)$. Therefore, u could be defined as: $u = \frac{1}{n} \times \sum_I \frac{\sum_j (T_j \times B_{res}(I,c))}{T \times B_{cap}(I,c)}$

QAWBA is compared with a simple scheme, in which a QoS request is successful if the mobile node currently has enough available cellular bandwidth, otherwise, the it fails.

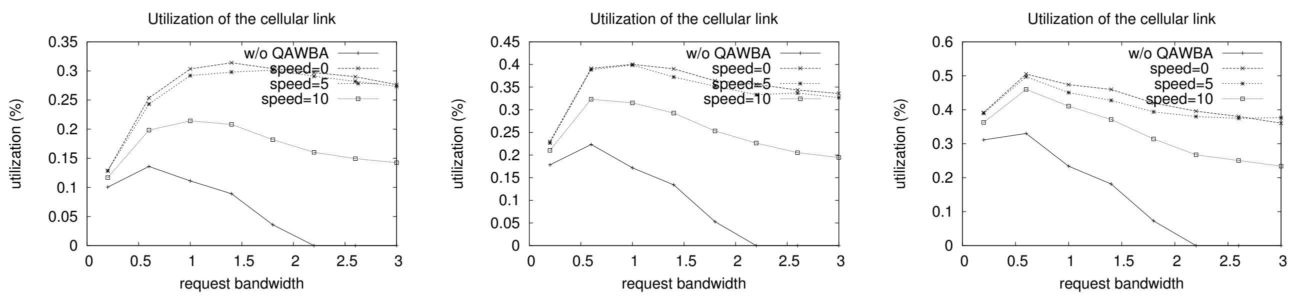
4.2. Simulation Results

Figure 5 presents the average admission rate r of QoS request with different requested bandwidth (B_{req}), different traffic load ($\delta = 40$, $\delta = 20$, and $\delta = 10$) and different node moving speed ($speed = 0$, $speed = 5$, $speed = 10$). Without the help of QAWBA, the admission rate drops dramatically with the increase of the size of the QoS request. The admission rate is less than 22% when $B_{req} > 1.4Mbps$ and $\delta = 40$. It drops to zero when the B_{req} is greater than 2Mbps, the maximum of the cellular link capacity. In QAWBA, the admission rate is much higher than the one without QAWBA. The mobile nodes are able to accept more QoS requests than are possible under normal condition. For example, the QAWBA approach enables the admission rate to reach 50% when $\delta = 40$ and $B_{req} = 2Mbps$. Without QAWBA, none



(a) $\delta = 40$ (b) $\delta = 20$ (c) $\delta = 10$

Figure 5. QoS request admission rate with different request bandwidths and δ



(a) $\delta = 40$ (b) $\delta = 20$ (c) $\delta = 10$

Figure 6. Cellular link utilization with different request bandwidths and δ

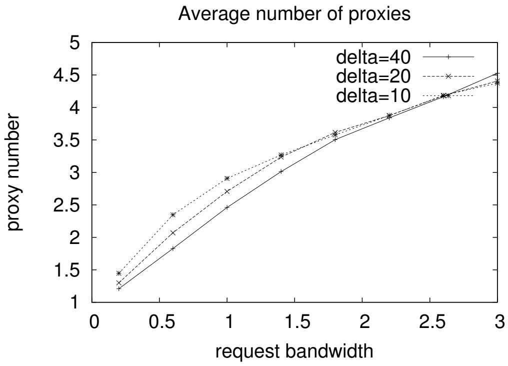
of the QoS requests can be accepted. With heavier system traffic load (smaller δ value), the increase of the admission rate in QAWBA becomes more significant. When $\delta = 10$ and $B_{req} = 1.4Mbps$, the admission rate in QAWBA is 35% while it is only 11% without QAWBA. QAWBA helps to increase the admission rate by 318% in this case. Mobility has some effect on the admission rate in QAWBA. The admission rate drops a little as the mobile nodes move faster. It may be explained due to an increased number of broken routes during the proxy discovery when the mobile nodes move faster.

Figure 6 shows the utilization of the cellular link u with different requested bandwidth (B_{req}), different traffic load ($\delta = 40$, $\delta = 20$, and $\delta = 10$) and different node moving speed ($speed = 0$, $speed = 5$, $speed = 10$). Without QAWBA, the utilization rate is very small, less than 10% in the most cases. The QAWBA scheme significantly increases the utilization of the cellular link under different system loads. For example, the utilization rate is only 8.9% with $\delta = 40$ and $B_{req} = 1.4Mbps$ without cooperation. In QAWBA, the utilization rate increases 3.5 times and reaches 31.3%. The increase is more significant with larger requests. Mobility also decreases the utilization of the cellular link due to its effect on the admission rate.

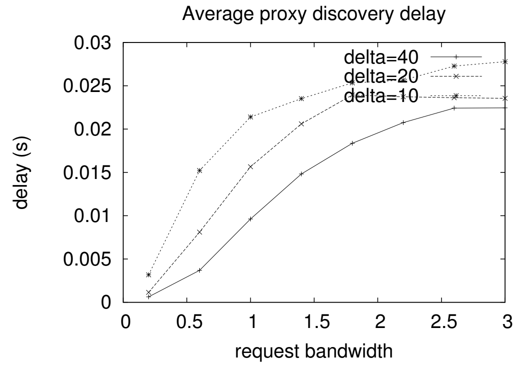
The average number of proxies in QAWBA is shown in figure 7(a). It increases nearly linearly with the increase of the requested QoS bandwidth. The more bandwidth requested, the more proxies are needed to contribute cellular bandwidth. As shown in figure 7(b), the delay of proxy discovery process increases with the increase of the requested bandwidth. Since more proxies should be found for a larger request, the delay of the proxy discovery is also increased.

5. Conclusion

Current cellular networks cannot meet the QoS requirements of many multimedia applications. Although



(a) Average number of proxies



(b) Average delay of proxy discovery

Figure 7. Number of proxies and delay of proxy discovery in QAWBA

the cellular interface provides "anywhere, anytime" network access, IEEE 802.11 based network interfaces have become the de factor interface for many mobile devices. We provide in this paper a QAWBA system that utilizes both the cellular network interface and the IEEE 802.11 ad hoc network for an integrated network architecture that provides QoS aware wireless bandwidth aggregation. Mobile nodes form a mobile ad hoc network via their IEEE 802.11 based network interface. The capacity of several low throughput cellular links are shared by all mobile nodes to provide better QoS support for the application. The issues of security (authentication, privacy) and billing are beyond the scope of this paper. The simulation result shows that QAWBA could significantly increase the utilization of the cellular resource and the admission rate of the QoS requests.

References

[1] S. Chen and K. Nahrstedt. Distributed Quality-of-Service Routing in Ad-Hoc Networks. *IEEE Journal on Special Areas in Communications*, 17(8):1–18, 1999.

[2] H.-Y. Hsieh and R. Sivakumar. A hybrid network model for cellular wireless packet data networks. In *IEEE GLOBECOM*, November 2002.

[3] S. Kang and M. Mutka. Efficent Mobile Access to Internet Data via a Wireless Peer-to-Peer Network. In *Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PerCom2004)*, pages 197–205, March 2004.

[4] S.-B. Lee, G.-S. Ahn, X. Zhang, and A. T. Campbell. INSIGNIA: An IP-Based Quality of Service Framework for Mobile ad Hoc Networks. *Journal of Parallel and Distributed Computing*, 60(4):374–406, 2000.

[5] W.-H. Liao, Y.-C. Tseng, S.-L. Wang, and J.-P. Sheu. A Multi-Path QoS Routing Protocol in a Wireless Mobile Ad Hoc Network. *Telecommunication Systems*, 19(3-4):329–347, 2002.

[6] Y.-D. J. Lin and Y.-C. Hsu. Multihop cellular: A new architecture for wireless communications. In *INFOCOM2000*, pages 1273–1282, 2000.

[7] H. Luo, R. Ramjee, P. Sinha, L. Li, and S. Lu. UCAN: A Unified Cellular and Ad-Hoc Network Architecture. In *MOBICOM*, pages 353–367, September 2003.

[8] M. Mirhakkak, N. Schult, and D. Thomson. Dynamic quality-of-service for mobile ad hoc networks. In *International Conference on Mobile Computing and Networking*, pages 137–138, Boston, Massachusetts, 2000.

[9] C. E. Perkins and E. M. Royer. Ad-Hoc On-Demand Distance Vector Routing. In *Proceedings of the Second IEEE Workshop on Mobile Computing Systems and Applications (WMCSA99)*, pages 90–100, February 1999.

[10] C. Qiao and H. Wu. icar: An intelligent cellular and ad-hoc relay system. In *Proceedings of IEEE IC3N*, October 2000.

[11] P. Sinha, R. Sivakumar, and V. Bharghavan. CEDAR: a Core Extraction Distributed Ad hoc Routing Algorithm. In *IEEE INFOCOMM '99*, 1999.

[12] A. Veres, A. T. Campbell, M. Barry, and L. H. Sun. Supporting Service Differentation in Wireless Packet Networks Using Distributed Control. *IEEE Journal of Selected Areas in Communications*, 2001.

[13] Q. Xue and A. Ganz. Ad hoc QoS on-demand routing (AQOR) in mobile ad hoc networks. *Journal of Parallel and Distributed Computing*, 63:154–165, 2003.

[14] D. Zhu and M. Mutka. Sharing Presence Information and Message Notification in an Ad Hoc Network. In *Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PerCom2003)*, pages 351–358, March 2003.