QUALITY OF SERVICE SUPPORT FOR MULTIMEDIA APPLICATIONS IN MOBILE AD HOC NETWORKS - A CROSS-LAYERED APPROACH

S. Sivavakeesar, and G. Pavlou
University of Surrey, United Kingdom

Keywords: *Mobile Ad Hoc Networks, Quality of Service (QoS), Cross-Layering, Location-based Forwarding, Proportional Service Differentiation, QoS Routing*

Definition: *Quality of Service (QoS) is characterized as a set of service requirements to be met by the network while transporting a packet stream between a given source-destination pair, while a Mobile Ad hoc NETwork (MANET) is defined as an "autonomous system of mobile routers (and associated hosts) connected by wireless links – the union of which form an arbitrary graph".*

Introduction

Mobile ad hoc networks (MANETs) have recently received significant attention, and the eminent introduction of real-time applications in such environments has fuelled research activities in the area of Quality of Service (QoS) support. However, the unique features of MANETs, namely random mobility patterns of mobile nodes (MNs) and their bandwidth- and energy-constraint operations pose numerous challenges, and hence require cost-effective solution for any QoS provisioning mechanism. Given that QoS provisioning in MANETs is extremely challenging and is modeled as a multi-layer problem, this chapter takes a holistic view to this issue by identifying the required components of an overall MANET QoS framework. In this context, it mainly looks at the problem of QoS provisioning not only from the perspective of the network layer but also from the perspective of the medium access control (MAC) sub-layer. Accordingly, this chapter first proposes a QoS-aware MAC. This is followed by a justification for the use and proposal of scalable schemes of a hierarchical clustering and a location-management strategy in an attempt to devise a scalable routing protocol. The above aspect is necessary for the proposed QoS architecture that is developed subsequently and it mainly attempts to support a stronger notion of per-class service guarantees in terms of packet loss and delay in such networks. Since one of the key issues in providing QoS guarantees is how to determine paths that satisfy QoS constraints, this chapter finally studies the NP-hard delay-constrained least-cost path problem and presents a more distributed online heuristic solution that utilizes only local information. In this way, this chapter contributes in a number of vital areas as identified in Figure 1, and the key outputs of this research work are: i) a QoS-aware MAC facilitating spectrum-agile cognitive networks, ii) a novel clustering algorithm and protocol for topology control and scalable routing, iii) a new scheduling and buffer management strategy, and iv) an effective strategy for QoS routing (termed Stabilized Online Constrained-based Unicast Routing (SOCUR)).

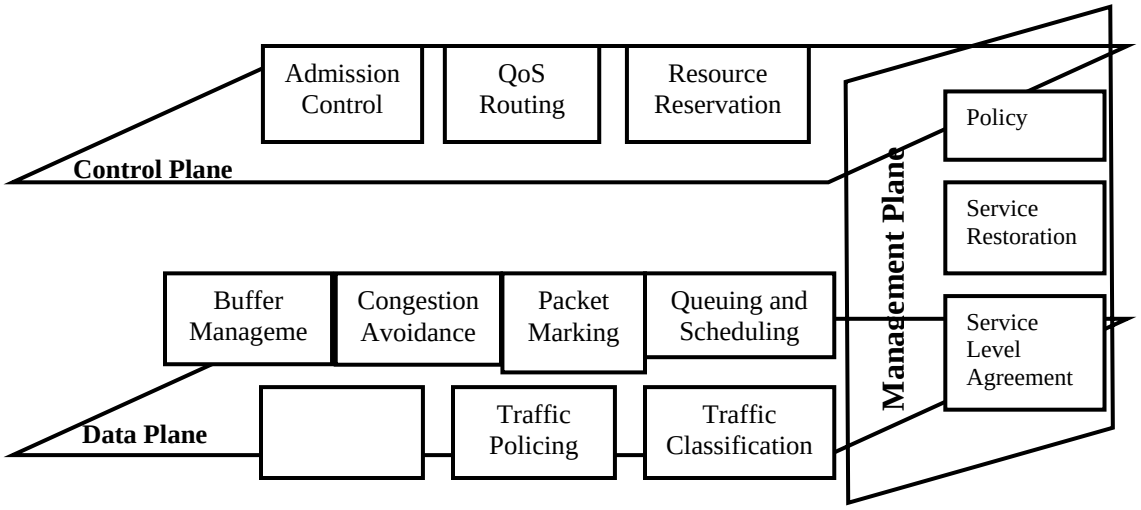

Figure 1. Necessary Components of Any Meaningful Quality of Service Architecture.

QoS-Aware MAC Protocol

Medium access control (MAC) plays a crucial role in the efficient and fair sharing of the common communication medium and hence in QoS provisioning. The importance for a QoS-based operation of MAC in the case of wired IP networks is not felt due to the nature of abundant bandwidth being available in such networks. As a result, although the available MAC mechanisms in the fixed IP networks are not QoS-aware, their impact/effect is not perceivable/significant due to the fact that the capacity/speed of wired links is tremendous with the advent of optical fiber. On the other hand, as discussed in [1][2][3][4], because of several unique characteristics that well distinguish multihop mobile ad hoc networks from their infrastructure-based wired and wireless counterparts, the MAC design in the case of MANETs becomes even more complicated. However, until now, a MAC based on the distributed coordination function (DCF) of IEEE 802.11 a/b is mostly prevalent in MANETs. Although multiple non-overlapping channels exist in the 2.4 GHz and 5 GHz spectrum, most IEEE 802.11 based MANETs today use only a single channel [5][7]. In addition, despite significant advances in physical layer technologies, today's IEEE 802.11 still cannot offer the same level of sustained bandwidth as their wired brethren. In addition, the advertised 54 Mbps bandwidth for IEEE 802.11 a/g is the peak link-layer data rate. When all the overheads – MAC contention, handshake packets such as request-to-send (RTS), clear-to-send (CTS) and ACK, packet errors etc. – are considered, the actual net bandwidth available to applications is almost halved per hop. Since this MAC is based on random access method of carrier sense multiple access with collision avoidance (CSMA-CA), its ability to support QoS especially when the contention rate is high is very small. Because of these reasons, a number of research works have questioned the suitability of DCF-based MAC for QoS support [3][4][5][6][7] in single radio multihop ad hoc networks.

Taking the above into consideration, we propose a new QoS-aware MAC protocol that works in conjunction with the location-based forwarding strategy for this purpose [11]. This novel protocol is based on the legacy IEEE 802.11, and thus can be relatively easily integrated into existing systems. It is adaptive and network-aware depending on the type and intensity of traffic and relative mobility patterns of nodes. In addition, it does not necessitate network-wide clock synchronization. Our strategy enables two-way admission control for improved performance, whereby the next-hop selection algorithm allows previous hop nodes to perform implicit admission control using locally available information, while a selected next-hop performs explicit admission control depending on its current load [3][5]. In order to support both asynchronous and time-sensitive multimedia traffic,

our MAC approach is based on a hierarchical strategy as depicted in Figure 2. It utilizes the DCF- and PCF-based operations of the IEEE 802.11 for the first time in multihop MANETs after being modified to accommodate MAC-level service differentiation. Our MAC protocol has the following three components in order to support QoS for real-time traffic as depicted in Figure 2: i) Admission control, ii) QoS-mapping, and iii) Resource reservation. Due to the fact that both the proposed MAC and location-based forwarding strategy work on the same principles (i.e., both use the local behavior to achieve a global objective), in this section we combine our MAC scheme with a location-based forwarding strategy [11].

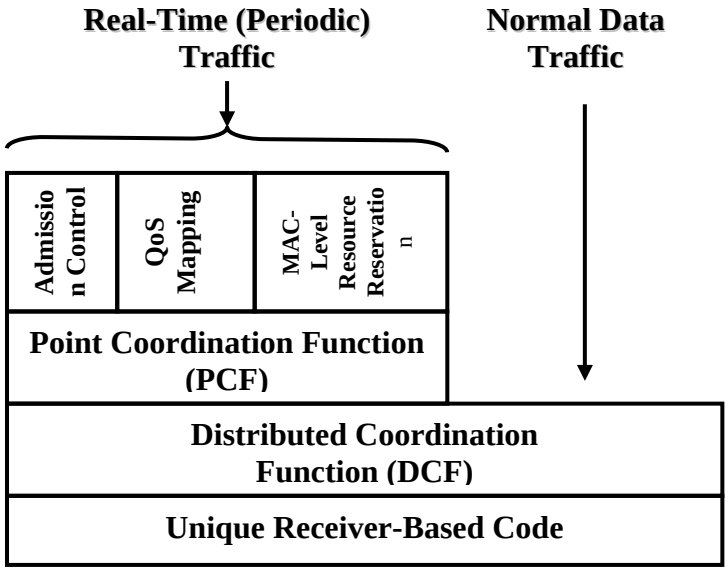

Figure 2. High-Level Functional Model of Our QoS-Aware MAC Framework.

As mentioned before, current MANETs operating on in the Industrial Scientific and Medical (ISM) band lack enough bandwidth to satisfy the above due to interference from various sources, heavy reliance on random medium access control (MAC) technologies, the need to use a handshaking mechanism to minimize hidden- and exposed-terminal problems, channel errors/uncertainties, etc. [2][3][4]. However, the use of single channel degrades network performance when the network size increases, and fails to meet the increased throughput and delay requirements of new applications [7]. On the other hand, supporting real-time applications in any network necessitates the availability of predictable resources. This is possible by harvesting additional resources and employing a central agency, which can have a control over the scarce channel resources for efficient and fair sharing. On the other hand, the very basic requirement of an ad hoc network is that it should not rely on any central node. However, some form of an agency to manage the channel resources is still required for QoS support. In order to accommodate the above mentioned mutually conflicting requirements, multiple parallel channels (multiple radios) are used in our scheme in order to improve capacity and scalability. Accordingly, each node is assigned a unique receiver-based channel [6], and each node behaves as a central node (AP) with respect to its own unique channel (medium).

This is possible because the current IEEE 802.11 a/b standard can support multiple non-overlapping channels. However, at the same time, there does not exist any mechanism to use any currently-idling licensed spectrum. It is, hence, understandable that this artificial spectrum scarcity is due to the complicated current regulatory structure and lack of innovative technologies.

However, with the advent of Opportunistic Spectrally Agile Radio (OSAR) systems, the real deployment of QoS-enabled MANETs will soon become a reality. This technique is becoming increasingly attractive particularly in the domain of MANETs in order to ensure their future success, because through intelligent spectrum sharing this technique can provide large amount of resources for delay sensitive and bandwidth greedy multimedia applications. In such systems, each mobile node assigns dynamically itself a unique channel depending on the intensity of traffic it handles by applying a universal hash-function in a conflict-free manner [8][4]. We envisage that such a unique hash-function takes the address, location, etc. of a given node as input and determines the channel to be used. This chapter, though, does not delve into the problem of finding a suitable hash-function, but the same technique employed in [8] is assumed here. The same hash-function can be used by any node to determine the unique receiver-based channel utilized by any of its one-hop neighbors.

In this receiver-based channel-assignment scheme, any sender has to transmit data using the receiver's unique channel, and hence, under normal circumstances each node uses its own channel to receive data from other nodes [4][6]. In addition, there is a common channel, which all nodes can use to disseminate and acquire mostly neighbor and routing related control messages. Accordingly, under normal circumstances each node in our scheme has to monitor its own unique channel and also the common channel for the reception of data and control frames respectively. However, there may be an exceptional case, where any node may be required to transmit data on the common channel as it is explained in [3][4][6]. These channels are assigned to nodes dynamically in a conflict-free manner using the common channel [7]. Since the unlicensed spectrum using IEEE 802.11 is extremely limited, an intelligent channel assignment scheme can lead to a proper coordination of the spectrum utilization which in turn mitigates coexistence/interference problem and increases the spectral efficiency. For a complete explanation of the way the novel QoS-MAC protocol operates and how it enables two-way admission control through localized promiscuous listening, and localized mobility and load predictions, the reader is referred to [3][4][6].

A Novel Clustering Scheme for Scalable Topology Control and Routing

Clustering in mobile ad hoc is adopted mainly for localizing the topology updates, limiting the scope of routing protocol's response to node-mobility and for scalability reasons [9][10]. However, we need to understand that such a process should not lead to increased control overhead. In order to achieve this, we need to make sure that maintaining cluster stability amidst nodes' random mobility patterns is paramount. Hence, for any clustering mechanism in mobile ad hoc networks to work economically and efficiently, it needs to consider node-mobility. In a MANET that uses cluster-based services, network performance metrics such as throughput and delay are tightly coupled with the frequency of cluster reorganization [3][9]. Therefore, stable cluster formation is essential for scalable routing and even for better QoS. However, almost all clustering mechanisms appearing in the literature fail to ensure this. Cluster formation should be achieved at minimal communication overhead and computational complexity. Consequently, in a highly dynamic environment, the algorithm should be distributed, operate asynchronously, and require minimal coordination among the nodes.

The primary step in clustering is the election of cluster heads (CH) and the formation of clusters around them. However, it is worthnoting that unlike in fixed networks, in multihop mobile ad hoc networks, the assignment of nodes to clusters is a highly dynamic process, as node mobility continuously alters connectivity and spatial relationships among nodes. Hence, any complete clustering framework in MANETs should specify an algorithm for dynamically assigning nodes to clusters, and for responding to node mobility.

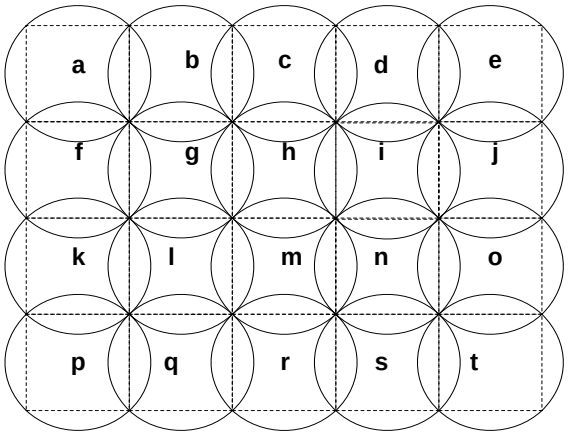

Figure 3. The Concept of a *Virtual-Cluster*.

Our strategy, on the other hand, differs from other similar approaches in two important aspects: a cluster head is elected based on spatial-associativeness, and it is based on the introduction of geographically-oriented *virtual-clusters* [3][10]. The idea is that a geographical area (or even the whole earth) is divided into equal regions of circular shape in a systematic way so that each mobile node can determine the circle it resides in if location information is known. The *virtual-cluster* is ideally a circular region centered on a *virtual-cluster* centre (VCC) as depicted in Figure 3. These VCCs are associated to a particular region in such a way that the resulting *virtual-clusters* are nearly overlapping. These circle area regions are our *virtual clusters*; a *virtual-cluster* becomes an *actual cluster* if MNs exist in it. Unlike in other clustering schemes, in our approach each *virtual-cluster* is supposed to have a unique identifier based on the geographic location, which can be calculated using a publicly known hash-function [3][8][10]. It is necessary that each *virtual-cluster* have a unique identifier for our concept of associativeness (and thus mobility prediction) to work in a scalable manner. Each MN is supposed to have a complete picture of the locations of VCCs. Our leader election heuristic takes the importance of cluster stability into consideration, and tries to elect stable cluster heads and thus form stable clusters. Having taken into account the common deficiencies of other approaches, our algorithm selects a MN as a CH, if it satisfies the following criteria: i) it has the highest spatial-associativity with respect to a specific *virtual-cluster*, in comparison to other MNs within the same cluster, and ii) it has the minimum distance from the respective *virtual-cluster* centre (VCC). The first requirement tries to ensure that a highly mobile MN is not elected as a CH. The second is to ensure that by being located very close to a VCC, the CH can have a uniform coverage over a specific *virtual-cluster*. This in turn ensures that in subsequent CH changes, the area covered would not be impaired. Our approach, being termed Associativity-based Clustering protocol, is motivated by the fact that link bandwidth and MN transmission power in MANETs are scarce, and any effective solution should take this into account and try to conserve them [2]. However, effective routing requires each MN to have up-to-date information on network topology, while keeping the control or signaling overhead as low as possible. In order to achieve a compromise between these two, accurate intelligent prediction of future state is necessary for the network control algorithms to keep pace with rapid and frequent state changes [2][3]. Hence, we propose a scalable mobility prediction scheme based on the physical associative nature of node with respect to its *virtual-cluster.*

As mentioned before, since mobility of nodes is the main cause of uncertainty in MANETs, our strategy considers mobility as the main criterion in the cluster head election process [9]. For this purpose, our CH election heuristic makes use of the concept of spatial-associativeness of a specific

mobile node with respect to a particular *virtual-cluster*. The concept of associativity was proposed and used as a routing metric for link reliability in [3]. In this work, the associativity-concept is used to reflect the degree of association stability between two mobile nodes over time and space. Nodes measure the connection stability by actively generating periodic beacons to signify their existence. In our scheme, however, every node tries to measure its spatial-associativity with respect to a specific *virtual-cluster* – as opposed to another node – by passively monitoring its presence in that cluster. It does not, however, involve periodic beacon transmissions. The CH election heuristic elects a node that has the highest associativity with respect to a specific *virtual-cluster* as the CH. With this technique, stable clusters are formed, and as a result the frequency of cluster re-organization is minimal. This in turn conserves scarce bandwidth and battery energy. Also, stability is an important issue, since frequent cluster head changes adversely affect the performance of other protocols such as scheduling, routing and resource allocation that rely on it. The key objectives of our strategy are to achieve stable cluster topology with minimal communications overhead, and to operate asynchronously in a distributed manner. In our proposal, CH does not take any extra workload, as it will otherwise become the bottleneck of the network [9][10]. For a detailed description about how our mobility prediction is performed and how this is used in our clustering protocol and algorithm, the reader is referred to [3][10].

The main purpose of this clustering technique is to make only a set of nodes (dominating-set) to handle the routing related information exchange in MANETs and to bring in a number of benefits. For scalability reasons, location-based routing is adopted in our framework, and this class of routing protocol has two important ingredients, i) location service, and ii) forwarding strategy. The help of a location service is needed in order to learn the current position of a specific node [11][12]. The performance of location-based routing, however, heavily depends on how well its location service operates and hence we are interested in devising a scalable location service using our stable clustering strategy. Previous work in this area has shown that the asymptotic overhead of location-management is heavily dependent on the service primitives (location updates or registration, maintenance and discovery) supported by the location-management protocol of the location service [3]. However, the location-registration or update cost normally dominates other costs for all practical purposes, and thus novel schemes are required to limit this control traffic. In our location-management scheme, we try to achieve this with an introduction of our stable geographically-oriented clustering protocol as described briefly above. The adoption of a hierarchical strategy together with the use of a dominating-set demonstrates as to how the control traffic is minimized without compromising route computation accuracy. This protocol does not involve any extra control traffic, and only periodic HELLO messages as in Ad hoc On-demand Distance Vector (AODV) routing protocol or other location service approaches are enough.

Our strategy is to address the scalability issue in both dense and large-scale networks. Scalability in dense networks is addressed efficiently by allowing only a few dominating set of nodes to make "summarized" composite periodic location-updates on behalf of a set of dominated nodes (the dominating-set in our context does not strictly follow the graph theory principles, and it refers to a set of CHs that can be reached by other neighbors not necessarily by single-hop but by single or k-hops at most, and dominated nodes are simply the members of a cluster). This is to minimize superfluous flooding by every node to the entire network, unlike in other location services [11]. Scalability in large-scale networks is addressed by strictly using geo-forwarding-based (location-based) unicasting as opposed to flooding even for the location-registration process, and preventing location-updates, queries and replies from arbitrarily traversing unnecessary parts of the ad hoc network. Hence, the performance of our geo-forwarding-based routing strategy is improved unlike other similar approaches where excessive control traffic, poor route convergence and routing-loops resulting from mobility degrade the performance. For a complete protocol specification and its evaluation, the reader is referred to [3][12].

A New Scheduling and Buffer Management Mechanism

The proposed model attempts to improve proportional and *soft* absolute service guarantees (absolute is used here in order to differentiate the service guarantee from proportional service differentiation, and it is used throughout this report to necessarily mean a *soft* guarantee and hence should not be mistaken for a *hard* guarantee) over multiple QoS metrics both in a single-hop and end-to-end manner [17]. This model assumes no communication (i.e., Resource reservation protocol (RSVP)-like signaling) between different nodes with regard to achieving optimal rate allocation and service guarantees. These tasks are performed independently at each node with sufficient information carried in each packet and with the use of location-based forwarding strategy. No explicit admission control or traffic policing is assumed, but similar effects are achieved using our next-hop selection algorithm.

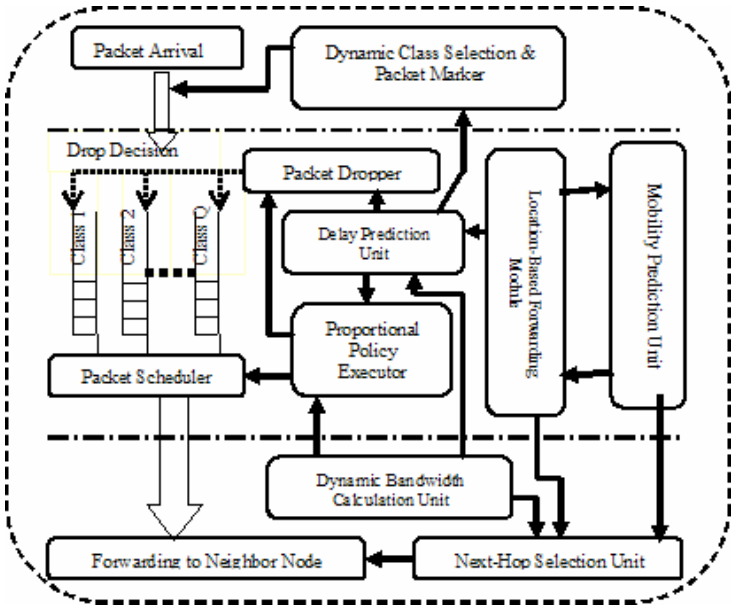

Figure 4. The Proposed Proportional Service Differentiation Framework.

A rate-based scheduler that allocates dynamic, time-dependent service rates to different traffic classes in order to improve proportional and absolute (*soft*) service guarantees is used to transmit traffic from the buffers [3][13][14]. Our rate-based scheduler is augmented with a mechanism in order to make delay predictions. Hence, it is termed here as prediction-based delay proportional (PDP) scheduler and it is based on a fluid traffic model. The dynamic rate allocation process of our system works as follows. For each packet arrival, new rates are calculated for each traffic class while satisfying the QoS and system constraints. In case there exists no feasible rate allocation that meets all the constraints, traffic is dropped with a careful selection of classes. Since the service rate allocation is defined in our approach as an optimization problem, the objective function aims at minimizing the average queuing delay and the number of packets being dropped per class. In order to satisfy system and QoS constraints, the ideal PDP scheduler has to make a prediction (projection) of the delays of all backlogged traffic upon each traffic arrival. Most of the time-dependent priority schedulers, especially Waiting-Time Priority (WTP), consider only the delay of the head packet in a queue, which is not fair to the other packets [3]. On the other hand, instead of considering only the delay of the head of a given class queue, our PDP scheduler tries to predict the average delay of all the packets. Since this is time consuming and necessitates high processing overhead, we only consider the average delays of the head and tail packets in our heuristic for convenience.

As long as the predicted delays for any two classes satisfy the system as well as the QoS constraints, the service rate or the loss rate for any class will not be altered. On the other hand, if the predicted delays do not satisfy any constraint, then either the service rate or the loss rate need to be altered for each backlogged class until the constraints are met as explained below. If there are no buffer overflows, the projections for delay violations are made in our heuristic only once for every Y packet arrivals (not upon every arrival). The selection of Y (a system parameter) represents a tradeoff between the runtime complexity and performance improvement with respect to satisfying the constraints. On the other hand, when there is a buffer overflow, packets need to be dropped while still maintaining the constraints. Since certain absolute constraints may lead to an infeasible system, some constraints need to be relaxed in a specific precedence order until the system becomes feasible. For this purpose, system constraints have priority over absolute constraints, which in turn have priority over relative constraints [13]. In our heuristic, the new service rate of each class is estimated either in one step or two steps, depending on whether any absolute end-to-end delay constraint needs to be satisfied. Step-I strives to ensure that the per-hop proportional (especially delay) constraints of the optimization problem are satisfied [14]. Step-II of the estimation process begins only if does there exist any end-to-end delay absolute QoS constraint to be satisfied as part of the optimization problem. If such constraints exist, each node should check whether the delay of a packet belonging to any of the delay-sensitive classes has already exceeded its delay bound (i.e., violated). If it is the case, then that packet needs to be dropped, as forwarding it is meaningless. In this process, not only does each node check whether the packet has already been violated, but also predicts whether the delay would be violated by the time it reaches its destination. As shown in Figure 4, the location-based forwarding mechanism provides our PDP scheduler with enough information to make this type of prediction.

Since this prioritized scheduling is performed independently at each node with the locally available information in a more distributed manner, this process is called Early Deadline First (EDF)-based distributed priority scheduling. When initiating time-sensitive flows, each source performs two operations: i) checking its own bandwidth availability, i.e., a source node should measure its bandwidth availability in order to decide whether it can accommodate its own time-sensitive traffic (i.e., source-based admission control). If there is non-availability of sufficient bandwidth, any time-sensitive flow should not be accommodated, and ii) class selection, i.e., the source should make the above end-to-end delay prediction for each available priority class in order to decide the selection of the appropriate class to satisfy the end-to-end requirements of its flows. This end-to-end delay prediction by any source is made based on the average queuing delay incurred for each traffic class in its own queues (i.e., based only on the local information) with our reasonable consideration that the burden of handling each delay sensitive packet should be equally shared by all downstream transit nodes. Having selected the appropriate class and initiated a flow, if any source node decides that it is impossible for a packet to meet its end-to-end deadline, then it should adaptively select a higher priority class so that the end-to-end delay is satisfied as shown in Figure 4. If, however, the priority class selected is the highest available class, then packets that are not expected to meet their deadline need to be dropped. On the other hand, if a source node perceives that selecting a lower priority class is enough to satisfy the flow's end-to-end delay requirement, this flow could be moved to the appropriate lower class [3].

Packet drops are inevitable when buffer overflow occurs. The proportional loss rate dropper we used is a simple dropper having two objectives: i) trying to minimize the number of packets being dropped, and ii) when there needs to be a packet drop, picking a packet from a certain class in order to keep the loss rate proportional, while satisfying the other constraints [3][13]. The concept of weighted loss rate is used in order to make the packet dropping decision. The full description of our proportional service differentiation model achieved through novel rate-based scheduling is presented in [3].

Quality of Service Routing

Constraint-based (or quality of service (QoS)) routing is an invaluable part of a fully-fledged QoS architecture as identified in the introduction section [3]. Our aim and hence the objective of this section is to delve into this research area by exploiting the experience we gained in other areas particularly in the design of scalable routing and rate-based scheduling. This section, hence, studies this problem (more specifically quality of service (QoS) routing or the NP-hard delay-constrained least-cost path problem) and presents a more distributed on-line heuristic solution that utilizes only local information such infrastructure-less networks [16][18]. The heuristic is termed stabilized on-line constraint-based unicast routing (SOCUR). The delay guarantees provided are *soft* as opposed to *hard* – although SOCUR attempts to improve the end-to-end delay bounds of applications.

Let the ad hoc network be represented as a directed and connected graph G = (V, E), where V is the set of nodes in the graph, and E is the set of edges in the graph. As mentioned before, each node uses only the local information for SOCUR to work. In this respect, let N(U) be the one-hop neighbor-set of node U. The local routing information that needs to be present at any node are i) the bandwidth availability, ii) current velocity, and iii) remaining battery energy of node U when a packet (say) m to be forwarded arrives. The above information is maintained by any node U ∈ V pertaining to itself and its every one-hop neighbor I ∈ N(U). Any node U learns the above-mentioned information pertaining to its one-hop neighbor I ∈ N(U) through periodic HELLO packets, and for this purpose every HELLO packet should accommodate this three-piece information. Each node U ∈ V maintains this information for every I ∈ N(U) in the form of: bandwidth-vector, Link Expiration Time (LET)-vector and cost-vector. In addition to the above information maintained locally at each node, certain information carried by each packet relating to the type-of-service, packet creation time, packet generation rate, minimum rate to be allocated, and details in the location-header are used in the constrained path construction process. The type-of-service (tos), packet creation time and packet generation rate for every flow k belonging to higher priority classes are inserted only by the source node S of flow k, while the minimum rate to be allocated is determined and inserted by both source and the intermediate nodes the packet traverses – the details follow later.

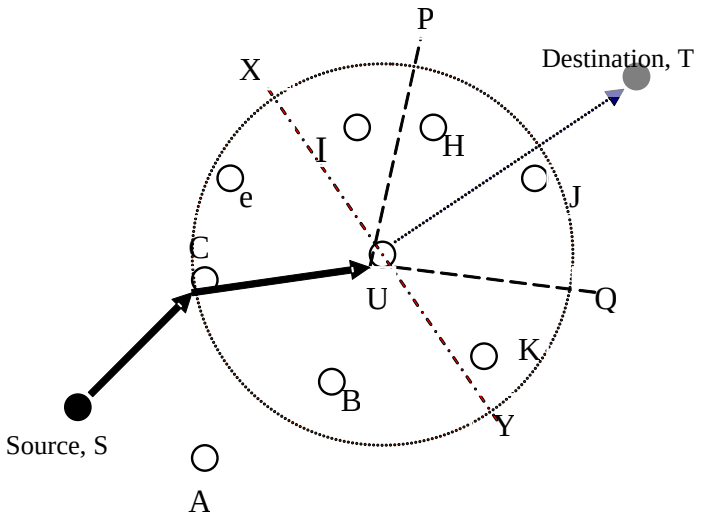

Figure 5. The Proposed Proportional Service Differentiation Framework.

SOCUR does not assume the use of any special control (signaling) packet to find or construct constrained paths. This is possible with an adoption of location-based forwarding strategy, which in turn improves scalability and more importantly enables high robustness to node-mobility and a distributed hop-by-hop routing. With this strategy any node U is able to choose any one-hop neighbor I ($\in$ N(U)) as its next-hop node for a given packet depending on the critical nature of packet that U needs to forward. We consider modified greedy forwarding, and SOCUR chooses the most suitable one-hop neighbor as its next-hop node through a process that undergoes four consecutive "filtering" processes. Consider the example illustration of Figure 5, where source S initiates a flow k belonging to high-priority with a given end-to-end delay requirement to destination T. Let node U currently hold a packet of flow k to be finally routed to destination T, and further assume that the source S has selected node C as its next-hop, and C has selected U as its next-hop. The circle ring around node U represents its transmission range. Also assume that the minimum service rate to be supported as indicated in the packet is satisfied by U – this aspect will be elaborated later in this section. Now let us see how node U selects its next-hop. The first filtering is intended for selecting a suitable next-hop node that can facilitate the dispatch of a given packet to its respective destination using the modified greedy forwarding principles. The input to this filtering is the one-hop neighbor-set. Since greedy forwarding normally tries to achieve shortest-path routing in a connected graph unless there is a "local-maximum" [3][11], node U considers nodes that are in the general direction of destination node T.

Once the first filtering is completed and the resulting one-hop neighbor-set is not null, the routing heuristic proceeds to the second filtering that takes the reduced one-hop neighbor set as the main input. The second filtering process takes the mobility of nodes and the underlying MAC into consideration. Unlike in fixed networks, in mobile ad hoc networks stability of links is important for successful data delivery. Since MAC needs an acknowledgement (ACK) frame from the selected next-hop for successful data delivery (otherwise it leads to unnecessary retransmission attempts, and hence energy and bandwidth wastage), a relatively stable neighbor node needs to be selected as next-hop. The second filtering process enables the SOCUR to find stable links in a more distributed hop-by-hop manner with a view to have stable routes – although no attempt is made to maintain end-to-end routes through any signaling mechanism. The link expiration time (LET) is used to find stable links [3]. Hence, a node U that tries to send a packet needs to first consult its LET-vector to identify each of its one-hop neighbors for which the minimum stability measure is satisfied. This way node U filters out unstable links, and hence unstable nodes with respect to itself from it input. Now the reduced neighbor-set is subject to the third filtering process, only when is it not null. Through this filtering, the neighbor nodes that have reached a critical condition in terms of the remaining battery energy available will be removed from one-hop neighbor set. Now the new neighbor-set of tagged node U after the third filtering process is considered for the fourth filtering, and assume that it is not null.

SOCUR now proceeds to the fourth filtering process, and only this part interacts closely with the rate-based scheduling mechanism. In this process, the information stored in the bandwidth-vector is going to be used. The critical nature of the current packet to be forwarded is considered in this process, provided that it has not violated its delay deadline. We explain next as to how the critical nature of a packet is defined, how it is determined, and how the rate-based scheduling works in conjunction with the constrained path computation in SOCUR. As mentioned before, in highly volatile MANETs, *hard* guarantees are difficult to achieve in certain situations. Hence, before forwarding, each node U normally checks whether the delay of a packet belonging to high-priority class-set has already exceeded (i.e., violated) its delay bound. If this is the case, then that packet needs to be dropped as forwarding it is meaningless. In this process, not only does each forwarding node U check whether the packet has already been violated, but also predicts whether the delay would be violated by the time it reaches its destination. It is the location-based forwarding

mechanism that provides the SOCUR and the rate-based scheduler with enough information to make such a prediction. Hence, the accuracy of such delay predictions depends heavily on how effectively the underlying location service works. At U, SOCUR estimates the shortest distance between the destination T and itself from the destination location it can find in the location-header of a packet in terms of number of hop-counts. This is the minimum number of hop-counts the packet under consideration may traverse when traveling from U to T. The denser the network, the more accurately this measure approximates the actual values taken by the number of hops. Nevertheless, in this local independent estimation process, each transit node U uses a rough estimate of transmission range in order to determine the minimum distance in terms of the number of hop counts only. Note that the transmission range of node U and that of each of the downstream nodes may not necessarily be equal. However, as it will be stated succinctly, every node U that forwards a packet performs this independent estimation process, so as to determine how quickly a given delay sensitive packet needs to be serviced at U.

If the packet creation time and the maximum end-to-end delay bound for the high-priority class that packet belonging to are known, then SOCUR enables the intermediate node U to estimate the total remaining time (maximum limit) for the given packet to reach its destination T. If this estimate is less than zero, the packet will be dropped. If it is not very small, node U proceeds to forward the packet. Since node U has a rough estimation about the number of hops the packet needs to traverse for it to reach the destination, U can roughly estimate how quickly that packet needs to be served at U and in subsequent downstream nodes. In this estimation process, it is fair to consider that the burden of handling the given packet should be equally shared among the present node U and the subsequent downstream nodes that the packet will traverse on its way from U to T. In other words, let us consider the case that the node U and the subsequent downstream nodes should allocate the same minimum service rate to the given packet depending on its critical nature, and hence the given packet is considered as incurring the same amount of delay in U and the subsequent downstream nodes (however, service delay in each subsequent downstream node may vary depending on the current load and the independent calculation performed in each node using the local information based on the packet's "critical" nature, i.e., the service rate needs to be increased/decreased in subsequent downstream nodes, and this time limitation is considered in the downstream next-hop node selection process). This consideration along with the use of a single transmission range to determine the number of hops as stated above are made in local computations only and does not necessarily introduce any significant undesirable effect, as they are used to judge as to how quickly a given time sensitive packet needs to be serviced primarily in the current node. Even if one transit node introduces an error with these considerations, it will be subsequently corrected to a certain extent by other downstream nodes – as each transit node estimates this independently using its local information. Note that the nearer the given packet to its destination, the more accurate the local information regarding the given destination, the lower the errors being introduced by each node and the more accurate the local estimation would be. On the other hand, in the absence of very accurate information regarding delay being available in each node due to bandwidth-constrained nature of MANETs, such rough estimations are inevitable and reasonable for any node to make in its local computations. As mentioned before, it is the effective operation of the underlying location service that provides accurate location information of desired nodes – this in turn enables each transit node U to perform highly accurate local estimation. However, care has been taken to adjust the inaccuracies associated with these estimations using as specified in [3]. With this the maximum allowable service time and hence the minimum rate at which the given packet needs to be serviced in the tagged node U is calculated.

Hence, in this fourth filtering process, the one-hop neighbors of node U out of the reduced neighbor-set resulting from the third filtering will be filtered-out based on whether they do have enough residual bandwidth available to support the current packet or not. For this purpose, node U

uses its bandwidth-vector. The resulting new neighbor-set of tagged node U after the fourth filtering process would contain nodes (provided that it is not null) that can make up a link, and hence a path, and can satisfy the given delay-constraint of the packet to be forwarded. Since our objective is to find a delay-constrained least-cost path, node U should now look for a possible next-hop node belonging to the reduced neighbor-set and that can result in a least-cost (LC) link. For this purpose, node U now looks for a possible candidate node for which the cost (which depends on how much the node/link is congested and how much battery energy remains in the nodes concerned) is the minimum. Note that any such possible next-hop candidate simultaneously satisfies the given delay-constraint.

While forwarding the given packet at the determined rate, the tagged node U also updates the minimum rate determined by itself based on the packet's critical nature in the minimum rate to be allocated field of the data packet header as in [15]. Assume that SOCUR of the tagged node U selects H of Figure 5 as the next-hop. Once node H has received the packet, it checks whether it can satisfy the service rate requirement as indicated by the previous hop node U. If it can, it admits this flow or else it will simply drop the packet. This situation might happen when the bandwidth information that node U had about H was wrong and stale. In order to accommodate such a situation, node U has to activate its promiscuous listening functionality as soon as it has forwarded a given packet. Accordingly, if node U decides that node H has not taken any attempt to forward the given packet that U had just sent within $\Delta_{Timeout}$, node U will try to find another suitable node belonging to the reduced neighbor-set of fourth filtering. Every node U that has just transmitted the packet needs to store the transmitted packet in its memory for at least $\Delta_{Timeout}$ from the time at which the last transmission is made – provided that the given packet has not violated its deadline. If, on the other hand, the given packet violates its deadline during $\Delta_{Timeout}$, it will be dropped. The value $\Delta_{Timeout}$ should take an optimal value, and it should always satisfy other conditions as specified in [3]. This way, promiscuous listening functionally helps to achieve the crank-back operation. If, on the other hand, the selected next-hop H forwards the packet within $\Delta_{Timeout}$ period, then the previous hop node U would no longer need to maintain the forwarded packet in its memory, and hence it can drop it.

Note that if for any reason the original or the reduced neighbor-set of tagged node U at each filtering process is null, SOCUR will return to the first filtering process and will try to increase the forwarding angle (given that greedy packet forwarding is used, the smaller the forwarding angle, the shorter the path that will result in). After each increase, the other filtering processes are performed to check whether the resulting new neighbor-set of each filtering process is not null. On the other hand, even after the maximum increase of the forwarding angle, if the resulting neighbor set of any subsequent filtering process is null, SOCUR will stop at the respective filtering stage. If this happens, the previous hop node C will initiate the crank-back operation facilitated through promiscuous listening as explained before. The value to be chosen for the step size of forwarding angle in SOCUR measures the tradeoff between the run time complexity and the accuracy of the optimal path. For the complete mathematical and simulation-based evaluation of the SOCUR model, readers are referred to [1].

Summary

Mobile ad hoc networks are rapidly evolving and the concept of mobile ad hoc networking has become one of the most challenging research areas of wireless communications. Given that quality of service (QoS) provisioning is an extremely challenging task and it is modeled as a multi-layer problem in such networks, this chapter took a holistic view to the issue of QoS provisioning by identifying the required components of an overall MANET QoS framework. It investigated the

problem of QoS provisioning not only from the perspective of network layer but also from the perspective of MAC sub-layer. Unlike similar QoS provisioning mechanisms that have mainly identified and studied the major challenges of mobile ad hoc networks, different architectural components of the QoS framework proposed in this chapter attempt to exploit some of the unique desirable features of MANETs, namely the use of location-based forwarding and promiscuous listening. The employment of location information becomes more and more realistic with the increasing availability of inexpensive positioning systems.

Since medium access control has been identified as a component that plays a vital role in QoS support, this chapter initially concentrated on this to devise a QoS-aware MAC. Given that the routing protocol is the key to the efficient operation of multihop mobile ad hoc networks, research on scalable routing was required as a prerequisite for attacking the problems of QoS routing and QoS provisioning. A class of routing protocol that uses geographical locations of the participating nodes has been chosen as the best candidate, because of its robustness to mobility and for scalability reasons. Our task was to devise a scalable location management scheme for this location-based routing protocol to work effectively. The next task was to devise a viable QoS provisioning mechanism for MANETs. Accordingly, this chapter introduced a service architecture that attempts to support stronger notion of per-class service guarantees in terms of packet loss and delay in ad hoc networks. The architecture relies on distributed priority scheduling enabled proportional service differentiation (PSD) model. However, it does not involve explicit admission control, traffic policing or maintenance of per-flow state information in any intermediate nodes. It uses (per-hop) local behaviors to achieve a desired global objective. Finally, given that one of the key issues in providing QoS guarantees is how to determine paths that satisfy QoS constraints, this chapter proposed a practically efficient solution for the simultaneous optimization of constrained path computation and scheduling for connections with end-to-end delay requirements in the domain of mobile ad hoc networks. In this way, this chapter contributed in a number of vital areas spanning the MAC and network layers. A novel clustering algorithm and protocol, a QoS-aware MAC, a scalable location service, a new scheduling and buffer management strategy, and an effective strategy for QoS routing and load balancing are the key outputs of this research work, resulting in a scalable QoS framework for ad hoc networks. All these areas have been addressed in the process of building our overall QoS framework.

Links: Our Research Homepages

1. http://www.ee.surrey.ac.uk/CCSR/
2. http://www.ee.surrey.ac.uk/Personal/S.Sivavakeesar/
3. http://www.ee.surrey.ac.uk/Personal/G.Pavlou/

References

1. MANET Working Group. http://www.ietf.org/html.charters/manet-charter.html.
2. S. Chakrabarthi, and A. Mishra, "QoS Issues in Ad Hoc Wireless Networks", IEEE Communications Magazine, Feb. 2001. pp 142 –148.
3. S. Sivavakeesar, "Quality of Service Support for Multimedia Applications in Mobile Ad Hoc Networks - A Cross-Layered Approach", PhD Thesis, University of Surrey, Dec. 2005.
4. A.C.V. Gummalla and J.O. Limb, "Wireless Medium Access Control Protocols", IEEE Communications Surveys and Tutorial, vol. 3, No. 2, Second Quarter 2000, pp. 2 – 15
5. S. Sivavakeesar, and G. Pavlou, "Two-way Admission Control and Resource Allocation for Quality of Service Support in Mobile Ad Hoc Networks", ACM SIGMOBILE Mobile Computing and Communications Review (MC2R), Jan. 2006.
6. S. Sivavakeesar, G. Pavlou, "Quality of Service Aware MAC Based on IEEE 802.11 for Multihop Ad Hoc Networks", Proc. of the IEEE Wireless Communications and Networking Conference (WCNC), vol. 3, Atlanta, Georgia, USA, IEEE, Mar. 2004, pp. 1482 – 1487.

7. P. Kyasanur, and N. Vaidya, "Routing and Interface Assignment in Multi-Channel Multi-Interface Wireless Networks", Proc. of IEEE Wireless and Communications and Networking Conference (WCNC' 05), vol. 4, Mar. 2005, 2051 – 2056.
8. L. Blazevic, L. Buttyan, S. Capkun, S. Giordono, J-P. Hubaux and J-Y. Boudec, "Self-Organization in Mobile Ad Hoc Networks: The Approach to Terminodes", IEEE Communications Magazine, June 2001, pp 166 –173.
9. A.B. McDonald, "A Mobility-Based Framework for Adaptive Dynamic Cluster-Based Hybrid Routing in Wireless Ad hoc Networks", PhD Thesis, University of Pittsburgh, 2000.
10. S. Sivavakeesar, G. Pavlou, "Associativity-based Stable Cluster Formation in Mobile Ad hoc Networks", Proc. of the IEEE Consumer Communications and Networking Conference (CCNC), Las Vegas, USA, IEEE, January 2005, pp. 196 – 201.
11. S-C.M. Woo, and S. Singh, "Scalable Routing Protocol for Ad Hoc Networks", Wireless Networks, Kluwer Academic Publishers, Sep. 2001, vol. 7, no. 5, pp. 513 – 529.
12. S. Sivavakeesar, G. Pavlou, "Scalable Location Services for Hierarchically Organized Mobile Ad hoc Networks", Proc. of the Sixth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc), Urbana-Champaign, Illinois, USA, May 2005.
13. N. Christin, and J. Liebeherr, "A QoS Architecture for Quantitative Service Differentiation", IEEE Communications Magazine, vol. 41, no. 6, Jun. 2003, pp. 38 – 45.
14. S. Sivavakeesar, G. Pavlou, "Rate Allocation and Buffer Management for Proportional Service Differentiation in Location-Aided Mobile Ad Hoc Networks", Proc. of the 3rd International Conference on Wired/Wireless Internet Communications (WWIC), Greece, May 2005, pp. 97 – 106.
15. I. Stoica, and H. Zhang, "Providing Guaranteed Services without Per Flow Management", Proc. of ACM SIGCOMM, Sep. 1999, pp. 81 – 94.
16. F.A. Kuipers, and P.V. Mieghem, "Conditions that Impact the Complexity of QoS Routing", to appear IEEE-ACM Transactions on Networking.
17. C. Dovrolis, D. Stiliadis, and P. Ramanathan, "Proportional Differentiated Services: Delay Differentiation and Packet Scheduling", Proc. of ACM SIGCOMM, Aug. 1999, Boston, MA, pp. 109 – 120.
18. Z. Wang, and J. Crowcroft, "Quality-of-Service Routing for Supporting Multimedia Applications", IEEE Journal on Selected Areas in Communications, vol. 14, no. 7, Sep. 1996, pp. 1228 – 1234.