

QoS-Aware Inter-domain Multicast for Scalable Wireless Community Networks

Ahmed Y. Al-Dubai, *Senior Member, IEEE*, Liang Zhao, Albert Y. Zomaya, *Fellow, IEEE*, and Geyong Min

Abstract— Wireless Community Networks (WCNs) have emerged as a cost-effective ubiquitous broadband connectivity solution, offering a wide range of services in a given geographical area. QoS-aware multicast over WCNs is among the most challenging issues and has attracted a lot of attention in recent times. The existing multicast schemes in WCNs suffer in terms of several key performance metrics, such as, latency, jitter and throughput, particularly in large-scale networks. Consequently, these schemes cannot accommodate the desired performance levels, especially when dealing with high-bandwidth applications that require efficient gateway-based management. To fill in this gap, a new strategy for supporting QoS-aware multicast in large-scale WCNs is proposed in this paper. Specifically, a new Gateway based Multi-hop Routing algorithm (GMR) is firstly proposed to enhance the routing management capability of the network. Built upon GMR, a new Multicast Gateway Multi-hop Routing algorithm (MGMR) is devised to cope with high-bandwidth applications in WCNs. The MGMR is the first of its kind that considers both the capability of gateway-based management and the requirements of high-bandwidth applications. Extensive simulation experiments and performance results demonstrate the superiority of both GMR and MGMR when compared to other methods under various operating conditions.

Index Terms— Wireless community networks, multicast, routing algorithms, QoS provisioning



1 INTRODUCTION

It is anticipated that wireless applications will continue to grow including, but not limited to, Voice over Internet Protocol (VoIP), Video-on-Demand (VoD), online gaming, real-time multimedia streaming, and community-based applications. The provisioning of broadband access to citizens and communities has been a strategic objective for organizations and governments worldwide to avoid or mitigate the digital division and promote the quality of life. Wireless Community Networks (WCNs) have emerged as a cost-effective ubiquitous broadband connectivity offering a wide range of services in a given geographical area.

WCNs have attracted tremendous research efforts [1], [2], [9], [13], [35], [36], [37]. Although several network architectures have been studied for WCNs, the Wireless Mesh Network (WMN) is undoubtedly one of the most popular architecture [9], [15], [18], [19], [20], [27], [32] owing to its many attractive features including rapid rollout, low capital costs, low power consumption, self configuration and organization, high integration with other networks, and easy installation. Furthermore, a WCN is capable of using the basic radio frequency to provide robust, flexible mobile broadband communications to different communities through the readily attainable multi-

hop connection [9], [25].

A number of representative mesh-based WCNs have been deployed to operate in many cities and rural areas worldwide [1], [2], [23]. Among these is the Athens Wireless Metropolitan Network (AWMN), which is one of the largest community mesh networks [1], [24]. In addition, SeattleWireless, CUWiN, Wireless Leiden, Freifunk is at the forefront of the WCN movement that accommodates a wider range of community-based applications [1], [2]. Due to the use of unlicensed spectrum and the absence of centralized management, WCN is vulnerable to interference and congestion that can degrade the Quality-of-Service (QoS) greatly. Thus, there have been many attempts to improve the QoS provisioning in WCNs [1], [9], [28]. The majority of the offered solutions, however, are limited to the Physical or MAC layers. In particular, QoS-aware group communication algorithms, e.g., multicast, have not been explored yet within the domain of QoS-aware gateway-based routing.

To fill in this gap, this paper aims to propose a novel routing scheme with the capacity of offering adaptive quality guarantees for QoS-aware streams under the umbrella of both unicast and multicast communication patterns. Group communications, such as multicast are the essential component and among most important design goals for the next generation wireless networks because they are instrumental to facilitate important community-based applications, such as data distribution, video conferencing, distance education, games and administrative services [5], [6], [17], [20], [22], [26], [27], [34]. In addition, multicast is a fundamental and indispensable component for dissemination of control information in many routing

-
- A. Al-Dubai is with the School of Computing, Edinburgh Napier University, U.K. E-mail: a.al-dubai, l.zhao@napier.ac.uk.
 - L. Zhao is with College of Computer Science, Shenyang Aerospace University, Shenyang China. E-mail: lzhao@sau.edu.cn.
 - A. Zomaya is with the School of Information Technologies, the University of Sydney, Australia, E-mail: albert.zomaya@sydney.edu.au.
 - G. Min is with the Department of Mathematics and Computer Science, University of Exeter, U.K. E-mail: g.min@exeter.ac.uk.

protocols. It has been reported in the literature that QoS-aware multicast over WCNs is among the most challenging issues [9], [21]. However, the existing multicast schemes suffer in terms of several key performance metrics, including latency, jitter and throughput, particularly in large-scale networks. Consequently, this degrades the communication quality, especially when dealing with high-bandwidth applications that require efficient gateway-based management. In fact, the gateway is a de facto network node that can orchestrate the traffic flows between the WCN nodes and external networks. However, most of the previous studies [7], [14], [16], [19], [25] either did not consider the key role of the gateway or did not take into account the applications that require orchestrating efforts at the gateway node, especially within the context of scalable networks, high traffic loads and QoS-aware applications. Furthermore, efficient multicast protocols in WCNs cannot be achieved by adopting or slightly modifying the multicast protocols used in other types of multi-hop wireless networks because the concern in WCNs has been shifted from energy efficiency and route recovery to QoS provisioning, mainly due to the high reliability, limited mobility and the rechargeable characteristic of WCN nodes. Moreover, supporting potential emerging applications, such as VoD and real-time community-based applications, poses a significant challenge on the limited bandwidth of WCNs. These challenges necessitate the need for adopting a new strategy to facilitate efficient multicast and community-based applications. To address these issues, this paper makes the following contributions:

- A new Gateway Multi-hop Routing algorithm (GMR) is proposed for WCNs in order to enhance the routing management capability of the network. Unlike the existing work, GMR shifts the role of the gateway from a simple packet forwarder to a routing orchestrating node. GMR is a hybrid routing in which both proactive and reactive components cooperate concurrently. The reactive capability offers great flexibility in changing environments and, on the other hand, proactive tree based routing is very efficient in fixed WCNs. This combination mode makes GMR suitable for implementation on a variety of different network configurations. This hybrid routing strategy adopts the leaf-to-gateway update mechanism to keep the gateway updated with the instant topology information. The traffic prediction method estimates the forthcoming traffic status on the path, which reduces the number of update processes and thus minimizes the traffic overhead. In addition, the time-to-live (TTL) setup mechanism is introduced to reduce the radius of update messages and further limit the flooding issue caused by the excessive update process. These mechanisms improve the communication efficiency by reducing both communication latency and jitter.
- Built upon GMR, a novel Multicast Gateway Multi-hop Routing algorithm (MGMR) is proposed to handle high-bandwidth applications. Unlike the existing work, the MGMR considers the QoS provisioning, load balancing and the capability of gateway-based

management in large-scale networks.

The remainder of this paper is organized as follows. Section 2 reviews the related work. Section 3 depicts the network model and presents the problem formulation. Section 4 proposes GMR and MGMR algorithms, respectively. In Section 5, the performance of GMR and MGMR is evaluated. Finally, Section 6 concludes this paper and provides pointers for future work.

2 RELATED WORK

Although multicast communication has been widely investigated in wireless and mobile networks, this problem in WCNs is still in its infancy due to the unique characteristics of WCNs that require new multicast strategies to facilitate the challenging community-based applications. A number of unicast and multicast routing schemes have been proposed for WCNs. To improve the mobility management, Li and Chen [23] proposed a Hierarchical Agent based Secure Multicast (HASM) algorithm to efficiently support secure mobile multicast in WCNs. However, this study did not include QoS provisioning within the gateway inclusion. Hwang and Lee [31] proposed a new multicast framework to facilitate video streaming in WMNs. However, this framework focuses on scalable video coding over time division multiple accesses (TDMA) only and the gateway management was not addressed. Although there is a great deal of research and proposals pertaining to the gateway-based routing designed for cellular networks, this issue is rarely considered in the existing routing studies on WCNs. For example, Hybrid Wireless Mesh (HWMP) protocol is a 802.11s-based WMN routing algorithm proposed in [8] to discover the optimal path in small/medium size networks. As a hybrid routing algorithm, HWMP consists of two different routing schemes, namely, reactive and proactive schemes. While a reactive routing scheme is implemented for the route discovery based on the AODV principle, a proactive tree building mode is used in which one of the nodes acts as ROOT *node_r*, which periodically broadcasts proactive PREQs. The address field of such PREQs is the broadcast address, thus every node receiving them then sends the path reply packets back to *node_r*. In this way, a proactive tree is built, and *node_r* has the routing table filled with all possible destinations. In fact, HWMP does not facilitate an orchestrating ability to gateways and suffers from excessive broadcasting messages, incurs extra overhead and degrades scalability in order to build the routing table. In [3], Chen *et al.* presented a multicast algorithm to enhance the throughput of wireless meshes. However, this algorithm is an MAC layer based scheme and introduces new channel feedback mechanisms. As an extension to the AODV protocol [33], Multicast On-demand Distance Vector Routing (MAODV) is an on-demand protocol for mobile ad hoc networks [12]. In other words, it only discovers the routes when having packets to send. A Route Request (RREQ) is originated if a node intends to join a multicast group or to send a message without a route to the multicast group. Every

multicast group is identified by a unique address and group sequence numbers for tracing the current status of the group situation. On-Demand Multicast Routing Protocol (ODMRP) is also an on-demand protocol [10] where a join query message is broadcast if a mobile node has packets to send without a route to the destination. Each node that receives the join query updates its routing table with the appropriate node id of the message sender and the reverse path back to the sender. It is worth indicating that, both MAODV and ODMRP do not consider the extra loads and central role of gateways in WCNs to facilitate efficient management for applications offered by the service provider side. More recently, Li and Chen [4] presented a hierarchical multicast algorithm that mainly focuses on mobility management and membership and does not, however, consider the QoS provisioning that can be handled by the gateway to meet the requirements of time-sensitive applications.

Load balancing in WCNs is a crucial objective. Usually, in WCN the gateway node and other network nodes on the shortest paths to the gateway can be extremely heavily loaded because the gateways and centre nodes relay both multicast packets and other background traffic. This can result in disastrous consequences on the overall network performance. However, with controlling the client-gateway registration, the load balancing can be improved dramatically, leading to a reduction in latency and transmission errors. Recently, a hybrid Gateway-cluster based Load Balancing Multicast algorithm (GLBM) [11] was reported to enable load balancing in multicast communications. As a hybrid multicast routing algorithm, the source node sends periodic multicast “Hello” messages to all the gateway nodes in an active way through the Internet backbone while the receiver nodes join the multicast group by sending requests to their gateway nodes on demand. When the requests are received, the gateway chooses the best path with the lowest traffic loads among the available paths and replies to the requested node. All the requests and data packets between the source and receiver are forwarded through the gateway. GLBM has shown a good performance with low mobility nodes.

Although the previous studies shed some light on various important issues, multicast communication in WCNs is still an open and challenging problem. To the best of our knowledge, there has not been any study reported on QoS-aware multicast with the efficient gateway inclusion in scalable WCNs so as to facilitate real-time and multimedia applications that require efficient gateway management. To fill in this gap, a new strategy is devised for supporting QoS-aware multicast in large-scale WCNs in this paper.

3 NETWORK MODEL AND PROBLEM FORMULATION

WCNs based on the mesh architecture are multi-hop networks that are composed of wireless links that typically include static wireless mesh routers/gateways and potentially mobile clients, forming a combination of both fixed and mobile networks. Each client device is outfitted

with a radio communication gear that acts as a relay point for other nodes as well as a central orchestrating gateway. There are three layers in a typical WCN: Internet Gateway Layer (IGW Layer) that covers a domain, Mesh Router Layer (MR Layer) and Mesh Client Layer (MC Layer) [8], [9], as illustrated in Fig. 1. Gateways relay the message between the backbone Internet and nodes inside WCNs. Under each gateway, routers are connected to a backbone network via the gateway in MR layer. To clarify the description of the proposed routing algorithm, we combine the MR layer and MC layer as both mesh routers and mesh clients are routing devices that use wireless radio and connect to the Internet backbone via two gateways IGW1 and IGW2 as illustrated in Fig. 1.

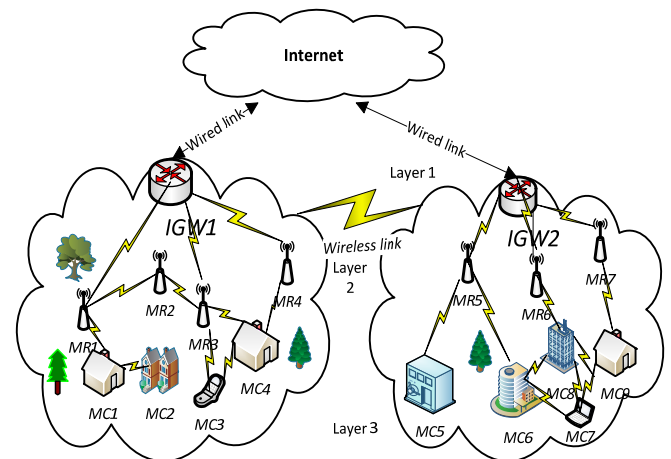


Fig. 1. The mesh architecture of a typical WCN with two domains

WCN consists of three types of nodes: mesh clients (MCs), mesh routers (MRs) and Internet gateways (IGWs) forming the layers of the typical structure of WCNs as depicted in Fig. 1. MCs are mostly static, and can be mobile end-users as well. To improve the flexibility of mesh networking, MRs nodes are usually equipped with multiple wireless (radio) interfaces built on either the same or different wireless access technologies.

Definition 1. Given a WCN modelled as a network graph $G = (V, E)$, where $V = \{v_1, v_2, \dots, v_n\}$ is a set of nodes n , including MCs, MRs and IGWs and $E = \{(i, j) : i, j \in V\}$ is the set of links (wireless and wired), a multicast set is a couple (v, \mathcal{D}) , where the source node is $v \in V$ and the destination set is $\mathcal{D} \subset V$, $\mathcal{D} = \{v_1, v_2, \dots, v_k\}$ and $k < n$.

To perform a multicast operation, a source node v , for instance, MC1 in Fig. 1 disseminates copies of the same message to all the destinations of \mathcal{D} , e.g. $\{MC2, MR1, IGW1\}$. Mesh routers have minimal mobility and thus form the mesh backbone for the MCs while the IGWs are fixed nodes to handle the external communication and enable the integration of WCNs with various other networks. Despite the capability of MCs to work as a router for mesh networking, the hardware platform and software for them can be much simpler than those of mesh routers [9]. WCNs are dynamically self-organized and self-configured with a high capability to grow as it can be

deployed incrementally as needed. Needless to say, more nodes are installed; the reliability and connectivity for the users increase accordingly, thus insuring a great deal of fault-tolerance.

4 THE PROPOSED SCHEMES

4.1 Gateway based Multihop Routing

The proposed GMR protocol offers high fairness in sharing network resources which leads to a better overall performance. In addition, the gateway gathers and stores real-time topological information and computes all the routing paths for the connected nodes in its domain. In what follows, the preliminaries are presented and followed by the registration and routing procedures of GMR.

Definition 2. Given a WCN $G = (V, E)$, a destination node, $d \in \mathcal{D}$, and a gateway GT that covers a set of nodes Ω , there are two main route discovery \mathfrak{R} processes between v and d : inner \mathfrak{R} and outer \mathfrak{R} , i.e., \mathfrak{R}_{in} and \mathfrak{R}_{out} , respectively:

$$\mathfrak{R} = \begin{cases} \mathfrak{R}_{in}, & \text{if } v \text{ and } d \in \Omega \\ \mathfrak{R}_{out}, & \text{otherwise} \end{cases}$$

For instance, if the source and destination nodes are MC1 and MC3, respectively, as shown in Fig.1, i.e., both can communicate with the Internet via the the same gateway, namely IGW1. However, $\mathfrak{R} = \mathfrak{R}_{out}$ if MC1 and MC5, for instance, would communicate. In GMR, a node can only register with one gateway. For example, although an adjacent node MC_i is within the transmission range of gateway $GT1$ and $GT2$, the route discovery process from MC_i to MC_j is $\mathfrak{R} = \mathfrak{R}_{out}$.

Definition 3. Given a WCN network $G = (V, E)$, $\forall v \in V$, v is said to be a leaf node if and only if it has a single one-hop neighbour; otherwise it is considered as a non-leaf node.

Definition 4. Given a WCN network $G = (V, E)$, a gateway GT , $\forall v \in V$, v is designated as a virtual leaf node if and only if there is no leaf node in a WCN or a Sub-WCN and $Dist(GT, v) = Max(Dist(GT, v_i))$. If there is more than one node satisfying the requirement above, a random one is selected.

Definition 5. Given a WCN network $G = (V, E)$, where $V_1 = \{v_1, v_2, \dots, v_n\}$ and $V_2 = \{v_{n+1}, v_{n+2}, \dots, v_m\}$ then $V1$ and $V2$ are two Sub-WCNs if and only if the gateway $IGW \in V_1 \cap V_2$.

Fig. 1, for example, shows two leaf nodes, MC2 and MC5 that are connected to the gateways IGW1 and IGW2, respectively. The rest of the nodes in Layers 2 and 3 are non-leaf nodes. In GMR, the gateway is fully aware of all the dynamic connection changes in the network. One of the innovative features of the GMR is its ability to operate efficiently in heavily loaded large scale multi-hop WCNs. This is achieved by enabling the route management in the gateway with the leaf-to-gateway update mechanism, traffic prediction method and TTL setting mechanism to

significantly reduce the flood caused by the requests of the broadcasted messages. In what follows, the registration and routing procedures of GMR are presented in Subsection 4(a). The route prediction and overhead minimisation mechanism are described in Subsection 4(b). The *TTL* setup mechanism is presented in Subsection 4(c). We discuss the significance of GMR along with the differences between GMR and HWMP within the involvement of gateway, QoS and flood control in Subsection 4(d).

(a) Registration and Routing

When a node intends to send or receive data, it must join a WCN. It broadcasts REQUEST with extended parameters in order to register with a gateway. For instance, node MC3 in Fig. 1 broadcasts REQUEST packets and finally registers with gateway IGW1 in order to communicate with node MR3. While an intermediate node, such as node MC1 or MC2 receives the REQUEST, it checks whether the current node is included in the path address field of the REQUEST packet. If it is the case, the packet is discarded. Otherwise, the intermediate node adds the node *id* and the QoS status to the packet header, and then rebroadcasts the packet. When a gateway receives the REQUEST from different paths, it records all the possible links in the Gateway Link Table (GLT) and then sends REPLY to the network nodes using the best path. Each intermediate node receives the REPLY, updates the route to the gateway and forwards the REPLY packet to the destination MC3. For building and maintaining the GLT, a leaf node that is defined as a WCN node with only one-hop neighbor is selected to broadcast periodic ROUTE_UPDATE messages in order to keep the GLT up-to-date.

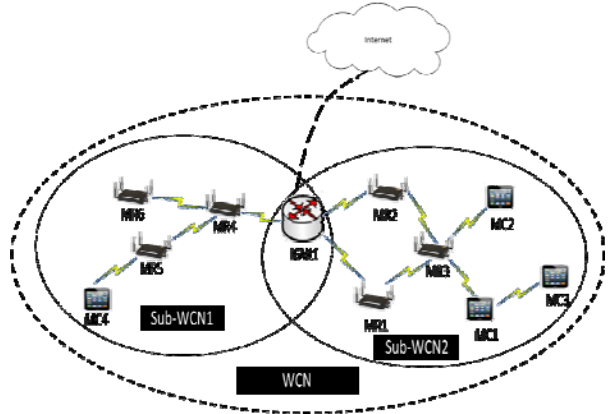


Fig. 2. Sub-WCNs of WCN.

If there are Sub-WCNs in a WCN, the gateway will select one leaf node for each Sub-WCN for reducing the duplication of sending update messages. Otherwise, only one leaf node is selected by the gateway. The ROUTE_LEAF message is sent to the chosen leaf node by the gateway. As soon as the leaf node receives the ROUTE_LEAF message, it will start the broadcasting of the ROUTE_UPDATE messages. In other words, unlike the existing schemes, GMR builds the GLT based on the proactively broadcast of periodic ROUTE_UPDATE mes-

sages from the leaf nodes in order to ensure that the routing table contains only the valid up-to-date routes.

GMR uses this type of leaf-gateway association scheme to update the real-time network topology and reduce the traffic overhead of sending out update packets from all the nodes to the gateway. For instance, Fig. 2 shows two Sub-networks, Sub-WCN1 and Sub-WCN2 of WCN. In Sub-WCN1, leaf nodes MC4 and MR6 are designated for sending the ROUTE_UPDATE messages, while MC2 and MC3 are leaf nodes designed by IGW1 in Sub-WCN2. This mechanism keeps the IGW1 has the latest topology of the network with minimizing the duplication broadcast of ROUTE_UPDATE messages. In addition, Algorithm 1 also gives the detail of the construction and maintenance of GLT in GMR. In Algorithm 1, $\text{Do_Dijkstra}(E, ST)$ computes the shortest paths of all possible pair nodes in E by considering the cost in ST , while $\text{run_Dijkstra1}(v, ST)$ computes the shortest paths from source v to all other nodes in V by considering the cost in ST .

Algorithm 1. Construction and maintenance of GLT in GMR

- 1 **Data** : E : edges; V : nodes; P : paths; ST : QoS status of links; e : edge; v : node; st : QoS status of a edge;
- 2 **Result**: P : gateway link table
- 3 $E(\text{WCN}) = \emptyset$; $V(\text{WCN}) = \emptyset$; $ST(\text{WCN}) = \emptyset$; $P = \emptyset$;
- 4 **for** \forall ROUTE_UPDATE recvd in IGW **do**
- 5 **for** $\forall e, \forall v$ and $\forall st$ included in this packet **do**
- 6 $E(\text{WCN}) = E(\text{WCN}) \cup \{e\}$; $V(\text{WCN}) = V(\text{WCN}) \cup \{v\}$;
- 7 $ST(\text{WCN}) = ST(\text{WCN}) \cup \{st\}$
- 8 **end**
- 9 **end**
- 10 $P = \text{Do_Dijkstra}(E(\text{WCN}), V(\text{WCN}), ST(\text{WCN}))$
- 11 **function** $\text{Do_Dijkstra}(E, V, ST)$
- 12 **for each** $v \in V$
- 13 $P = \text{run_Dijkstra1}(v, \text{all}, ST) \cup P$
- 14 **end**
- 15 **return** P
- 16 **end function**

If no ROUTE_UPDATE is received for a period of time, each non-leaf node broadcasts ROUTE_UPDATE-ERR until the gateway updates its GLT. In this case, the gateway considers that either an in-charge leaf node moves away or a non-leaf node is not reachable by ROUTE_UPDATE from the existing relevant leaf nodes. If there is no eligible leaf node (or there is initially no suitable leaf node), the gateway selects a virtual leaf node to send ROUTE_UPDATE. A virtual leaf node is the end node with the longest distance to the gateway. Thus, a virtual leaf node generates ROUTE_UPDATE packets that pass all the possible routes across the error node. For instance, in Fig.3, the gateway IGW1 has no eligible leaf nodes, while MC2 can act as a virtual leaf node and generate ROUTE_UPDATE packets.

Therefore, a path is discovered as follows. As shown in Fig. 3, MC4 intends to send packets to MC5 as both MC4 and MC5 are registered with the same gateway, IGW2. A registered MC4 sends ROUTE_REQUEST-A to its gateway for requesting a path to MC5. When a gateway receives ROUTE_REQUEST-A, it finds MC5 is inside of its WCN, i.e., this routing request is \mathfrak{R}_{in} . Then the best paths from MC4 to MC5 ($\text{MC4} \rightarrow \text{MR6} \rightarrow \text{MR7} \rightarrow \text{MC5}$) and from the gateway IGW2 to MC4 ($\text{IGW2} \rightarrow \text{MR6} \rightarrow \text{MC4}$) are calculated respectively, based on the GLT by running Dijkstra's algorithm [29], and are inserted to ROUTE_REQUEST-T. Then, IGW2 sends ROUTE_REQUEST-T to MC4 by following the path ($\text{IGW2} \rightarrow \text{MR6} \rightarrow \text{MC4}$). As soon as the ROUTE_REQUEST-T arrives at MC4, MC4 then encapsulates the packet to get path ($\text{MC4} \rightarrow \text{MR6} \rightarrow \text{MR7} \rightarrow \text{MC5}$) and start to send data packet along this path. Nevertheless, when the gateway receives a ROUTE_REQUEST-A packet and it could not find the requested targeting receiver node in its WCN, it assumes the targeting receiver node is out of its range and considers this routing request as \mathfrak{R}_{out} . It is worth indicating that for selecting the best path, the *Load Count* routing metric is considered instead of using Hop Count only to determine the QoS status of a node. This is primarily due to two main reasons.

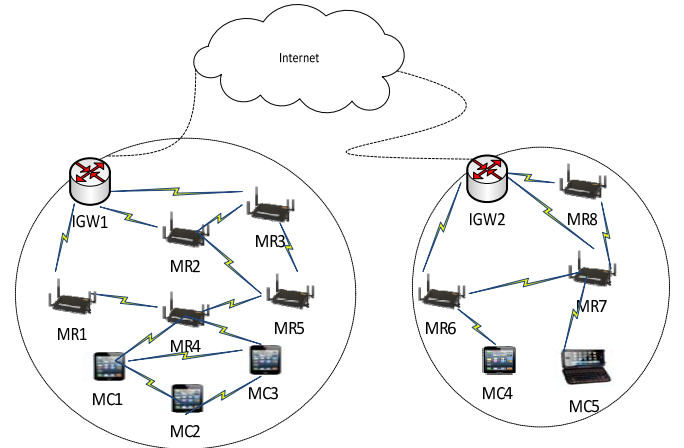


Fig. 3. Leaf nodes (MC4 and MC5) and virtual leaf node (MC2)

Firstly, *Load Count* is a widely used routing metric to achieve load balancing in WCNs. Secondly, compared to ETX [9] and ETT [10], *Load Count* here does not generate extra traffic for exchanging the information with neighbouring nodes in calculating the metric.

(b) Route prediction and overhead minimization

The Traffic Prediction Method [9] is implemented to further limit the flood of broadcasting ROUTE_UPDATE messages. We extend the waiting time of sending out ROUTE_UPDATE to $t \times \Delta$. The value of Δ should be changeable for different WCNs in order to obtain the best trade-off. However, for a specific network with a given working condition, Δ is a constant. In our simulation, Δ is set to be 2. This is found as the ideal value for network

sizing-50 and 100 nodes. However, for larger networks, Δ is adjusted to a smaller value to ensure the updates of WCNs. During the waiting period, while the gateway receives a route request from a network node, it estimates the future communication quality (QoS status) of the link from a MC m to destination d . It then updates the GLT before the destination node informs the source node of the path. The increment value of QoS status of each node is obtained as follows:

$$QoS_Incr = Routing_factor \times \frac{d_{infer}}{d_{node}} \quad (1)$$

where $Routing_factor$ denotes the impact value of routing metric caused in a single transmission, i.e., it is the average communication quality; d_{infer} represents the average interference distance in the network and d_{node} denotes the average distance between two neighbouring nodes, i.e., the average interference in hops in the network. The $Routing_factor$ is calculated by

$$Routing_factor = \frac{\sum_{i=1}^n Q_i}{n} \quad (2)$$

where Q_i represents the status based on the communication quality of $node_i$ on a given path. When the gateway receives the route request with inner-communication flag (ROUTE_REQUEST-I) message, it predicts the future link quality of the path as QoS_Incr and then adds the QoS_Incr value to the communication status of the nodes on the selected path.

(c) TTL setup mechanism

The predicted communication status of the selected link is stored in the gateway link table. The ROUTE_UPDATE packets flow along all possible paths to record the QoS status of these paths from leaf nodes to the gateway. When the gateway receives a ROUTE_UPDATE packet, the GLT is updated with the information of the passed links. To reduce the probing overhead caused by broadcasting ROUTE_UPDATE messages from leaf nodes, the TTL (time to live) field of each message is set as:

$$TTL = \begin{cases} n & \text{if } \ell = 0, \text{ or } \ell = 1 \\ \max \left(h^{\max}_{l \rightarrow G}, \left\lceil \frac{h^{\max}_{l_i \rightarrow l_j}}{2} \right\rceil \right) & \text{if } \ell > 1 \end{cases} \quad (3)$$

where l and G are referred to as the leaf node and gateway, respectively. The number of hops between l and G along the longest path is expressed by $h^{\max}_{l \rightarrow G}$, whereas $h^{\max}_{l_i \rightarrow l_j}$ denotes the number of hops of the longest path from l_i to l_j . In addition, l_i is the nearest leaf node to l_j , n denotes the number of network nodes and ℓ represents the number of leaf nodes in the network. Eq. (3) shows that all the network nodes should be involved in disseminating ROUTE_UPDATE messages in case that either there are no leaf nodes or there is only one leaf node reg-

istered with the gateway. Alternatively, each leaf $node_l_i$ will compare two values and select the bigger one to set-up its TTL. These two values are i) the number of hops from l_i towards the gateway along the longest path, and ii) the half of the number of hops along the longest path between l_i and its nearest leaf node to l_j . It is worth indicating that this formula of TTL has been validated by our experiments pertaining to the dynamism that complicates the delivery of the ROUTE_UPDATE message by considering a number of factors, such as network size, traffic load, QoS provisioning and types of applications. By applying the TTL value, this minimizes the probing traffic caused by ROUTE_UPDATE messages and avoids duplicated update coverage areas in the network. Hence, without adopting such a mechanism, communication quality of one non-leaf node could be affected by updates caused by more than one leaf node. *For more clarity on using TTL in GMR, refer to the example in Appendix A).*

As $node_i$ receives the reply, it starts to send data packets by using the reverse route in ROUTE_REPLY. Note that, the WCN node assumes there is a broken link in the shortest path to the gateway, if it does not hear the reply of the ROUTE_REQUEST from the destination for a certain period. To find an alternate path, $node_i$ then broadcasts ROUTE_REQUEST-ERR towards the gateway, using the *Load Count* routing metric instead of using *Hop Count* only. This is to pay more attention to the communication quality by monitoring the current traffic load over a node. As a load balancing metric for wireless networks, *Load Count* of a path is given by

$$Load_Count = \sum_{i=1}^n Load_i \quad (4)$$

where $load_i$ is the traffic load on $node_i$ which is normally captured by counting the IFQ (Interface Queue) length. The IFQ is a drop-tail buffer at the MAC layer of 802.11 radios, which contains the outbound frames to be transmitted by the physical layer where the size of IFQ is calculated as the number of the remaining packets in the buffer.

(d) Comparison with the existing HWMP protocol

Compared to HWMP, GMR has distinguished advantages that can enhance the capability of WCNs, as outlined below:

- i) Gateway inclusion: Both GMR and HWMP involve gateway to relay the request. The gateway forwards only the request from the source node to the destination node in HWMP and may select a path from the cache. In contrast, the routes are selected based on the link information in the GLT which enables the central management of network routing. In addition, the gateway in GMR is required to handle the update packets.
- ii) Communication quality control: HWMP uses *Airtime Link Metric* to select a radio-aware path. However, the gateway cannot find the best paths all the time as the cached paths in the relay node are affected due to the

dynamic changes of the network. However, in GMR the gateway chooses the best paths based on the up-to-date information of the network. In addition, most of the existing routing metrics can be applied to examine the quality of links in the gateway.

- iii) Flood Control: A node without a valid path to the destination in HWMP broadcasts a route request over the entire network which may lead to heavy network overloads caused by the additional traffic.

However, GMR applies the leaf-to-gateway update mechanism which is an efficient one-way link maintenance mechanism assigning a limited number of leaf nodes to generate update packets. The update is motivated by the fact that some nodes did not join or left the WCN without notifying the gateway. The leaf-to-gateway update mechanism enables the gateway to maintain the dynamic changes of the network in a proactive mode where leaf nodes are required to send out periodic update messages. In addition, a mesh node finds the gateway on demand in GMR, unlike the periodical Root Announcement (RANN) that was used in HWMP to announce the gateway presence. In GMR, a node finds its gateway by broadcasting REQUEST if there is only a need for sending out packets. For example, in Fig. 5, when *node_3* firstly joins the mesh network for finding a path to send data packets to *node_1*, *node_3* acts reactively to broadcast REQUEST. Moreover, the setup of TTL in GMP further reduces the traffic flows caused by the periodic update message.

4.2 Multicast Gateway- Multi-hop Routing

The gateway can be the optimal candidate node that plays a key role in orchestrating the group communication paradigms, e.g., multicast. Thus, to involve the gateway in managing various routing tasks, MGMR is proposed to enhance the gateway routing capability and promote the routing efficiency of multicast communication in scalable WCNs. This is essential for the multicast based applications that require high quality management at the service provider side. The gateway IGW ensures that at least there is one leaf node to generate ROUTE_UPDATE within its coverage. Therefore, the GLT of the gateway is constructed and maintained as explained in GMR. Note that, the tree maintenance of MGMR benefits from the propagation of fresh ROUTE_UPDATE messages, i.e., thus the multicast tree is constructed based on up-to-date GLT information.

The load balancing issue is a cornerstone to facilitate efficient communications in scalable and QoS aware WCNs, especially under heavy traffic load conditions. It is worth indicating that in a WCN the IGW node that acts as a relay node in the multicast tree, the MCs and MRs nodes on the shortest paths can be heavily loaded. Although there is a number of multicast algorithms proposed for WCNs [8], [12], [23], [28], none of which considered the load balancing within the context of multicast communication in scalable WCNs. Unlike the existing studies, the load capture mechanism is implemented in MGMR to monitor the load status along each node on the

multicast tree so that uneven traffic load can be avoided and load balancing can be enhanced to maintain high QoS provisioning. This is a concrete step to minimise both latency and transmission errors. For clarity, the following three definitions and two theorems are presented to discuss the methodology of MGMR in dealing with load balancing.

Definition 6. The load balancing in a network $G = (V, E)$ is achieved, if and only if the traffic load of a node i LB_i approaches the average load of the network $LB_i \rightarrow \overline{LB}$, $i \in V$.

Definition 7. Given a network, $G = (V, E)$, the total load of a network G is: $LB(G) = \sum_{i \in V} LB_i$ and $LB(M) = \sum_{i \in M} LB_i$ is the total load of a multicast group M .

Definition 8. Given a multicast session M in a network, $G = (V, E)$ if the multicast session is load-balancing aware, then $\overline{LB(M)} \rightarrow \overline{LB(G)}$, where $\overline{LB(M)}$ refers to the average load of M and $\overline{LB(G)}$ is the average load of the whole network G .

Theorem 1. Let δ be the load caused on each tree node by handling a single multicast packet and v be the number of multicast packet transmissions in a node, the average load of a network with a multicast session, in a group N of n nodes is given by $\overline{LB(N)} = \frac{LB(N)_{pre} + \sum_{k=1}^j \delta_k}{n}$ (refer to the Appendix B1 for the proof).

Theorem 2. To perform multicast communication in a given network $G = (V, E)$, the MGMR algorithm outperforms the hop-count algorithms in terms of load balancing (refer to the Appendix B2 for the proof).

In what follows, the session initiation, joining a session, leaving a session and tree maintenance are in MGMR will be discussed in details in Subsections 5a, 5b and 5c, respectively.

a) Multicast session initiation

To start a multicast session, a mesh node *source_s* notifies the gateway by sending MULTICAST_SREQ. The gateway receives this message and creates a new Gateway Multicast Table (GMT) for *source_s* with the new multicast session *id*. This GMT stores the paths of multicast tree from the source to all receivers. Then, the gateway node broadcasts periodic Multicast Hello Messages (MHM) to all the gateway nodes through the backbone.

b) Joining a multicast session

The gateway acts as a routing management centre to construct the Shortest Path Tree (SPT) rooted at the node *source_s* using the Dijkstra's algorithm within the GLT. The process of multicast tree construction is described in Algorithm 2, where $P(i, j)$ runs the Dijkstra's algorithm and returns the shortest path from i to j in GLT, $P_{Backbone}(i, j)$ returns the path between gateways in the backbone internet, $run_Dijkstra2(s, r)$ computes the shortest paths from

source v to receiver r . If any node, for example, $node_m$, aims to join the multicast session, it sends MROUTE_REQUEST with the “join flag” and field of the message, towards the gateway. Upon receipt of this join request, the gateway checks whether there exists a GMT for this source node within the same multicast session. If so, the gateway confirms that the multicast join process is an inner one, \mathfrak{R}_{in} . The GMT of this multicast session is encapsulated in MROUTE_REPLY with “join flag” to be sent by the “join reply” to $source_s$ followed by a pre-calculated path from GLT. The gateway is not in charge of forwarding data packets for inner WCN, but handles the route request between a receiver and a source for the inner WMN with the aim of keeping the traffic load at the gateway to a minimum. Therefore, in this case, if there is an alternate path to bypass the gateway, this path will be selected during the construction of the multicast tree.

Algorithm 2. Multicast tree construction for MGMR

```

1 Data:  $R$ : multicast receivers;  $s$ : source node;
2 Result:  $T$ : multicast tree;
3  $E(T)=\emptyset$ ;  $V(T)=RU\{s\}$ ;
4 for  $\forall$  node  $r \in R$  do
5   if  $\mathfrak{R}_{in}$ 
6      $E(T)=E(T) \cup P_{GLT}(s, r)$ 
7   if  $\mathfrak{R}_{out}$ 
8      $E(T)=E(T) \cup P_{GLT}(s, r) \cup P_{Backbone}(IGWs, IGWr)$ 
9 end
10 function  $P(s, r)$ 
11   for each  $v \in V$ 
12      $E = \text{run\_Dijkstra2}(s, r) \cup E$ 
13 end
14 return  $E$ 
15 end function

```

TABLE 1
FEATURES OF MAODV, ODMRP, GLBM AND MGMR

Feature	MAODV	ODMRP	GLBM	MGMR
Rout-discovery	On demand	On demand	Hybrid	Hybrid
Multicast type	Tree based	Meshed	Tree based	Tree based
metric	Hop count	Hop count	Load count	Load count
Rep.to source	Unicast	Multicast	Unicast	Unicast
Member maintenance	Hard-state	Soft-state	Hard-state	Hard-state
Group maintenance	Hello message	Hello message	Hello message	Leaf update message
GI (request forwarding)	No	No	Yes	Yes
GI(compute SPT tree)	No	NA	No	Yes

If there is no entry of this source node in the GMT, the gateway treats the join request as an outer one, i.e., \mathfrak{R}_{out} . Consequently, it encapsulates the best path from $source_s$

to $node_m$ via the backbone in MROUTE_REPLY with the “join flag” and sends the join reply to $source_s$. During the selection of the best path for GMT, the multicast SPT is either constructed based on the GLT if there is no existing SPT, or is reconstructed to merge the current GMT with the path from $source_s$ to $node_m$. As SPT is received from the gateway reply at the source node, $source_s$ informs the next level tree nodes to forward MROUTE_REPLY with “notify flag” message to $node_m$. Fig. 4 shows an inner WMN with $node_{10}$ and $node_{11}$ being the leaf nodes to generate ROUTE_UPDATE. For example, in this joining session of Fig. 4, a multicast source S first sends MULTICAST_SREQ to the gateway G for registration and establishing the GMT.

Beforehand, the gateway, G , has already maintained the routing table, GLT, storing all the paths by calculating ROUTE_UPDATE messages received from $node_{10}$ and $node_{11}$. Thus, once G receives MROUTE_REQUEST with the “join flag” from S , it creates a GMT for the multicast session for S and then informs S that the multicast session is initiated. While the second node, $node_7$, joins this multicast session by sending the request to G , then G computes and sends the GMT of this multicast session including the path between S and $node_7$ ($S \rightarrow node_2 \rightarrow node_7$) to S . After that, S stores this GMT and starts sending data packets. When $node_8$ and $node_{10}$ join the multicast session respectively, G also merges the best paths ($S \rightarrow node_3 \rightarrow node_8$) and ($S \rightarrow node_2 \rightarrow node_7 \rightarrow node_{10}$) from GLT with the existing GMT to construct the new SPT, and sends the GMT to source S . Then, this multicast tree SPT is built with two leaf_receivers ($node_8, node_{10}$), one forwarder_receiver ($node_7$) and two forwarders ($node_2, node_3$).

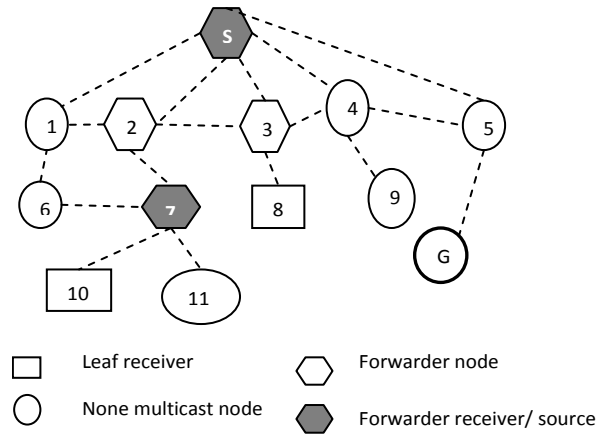


Fig. 4. Multicast tree constructed by MGMR inside a WCN

Fig. 5 depicts an example of multicast session among two gateways $G1$ and $G2$. When $node_9$ intends to join the multicast session, it sends MROUTE_REQUEST with the “join flag” associated with the multicast id towards its gateway $G2$. When $G2$ receives this request, it broadcasts this request in the Internet to find the right multicast session. When $G1$ receives the request, it notifies $G2$ through the Internet confirming the inclusion of the multicast session within its coverage and control.

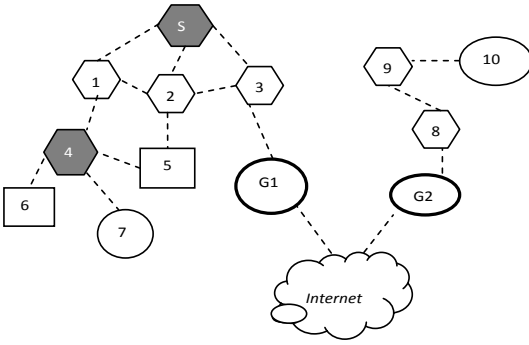


Fig. 5. Multicast session between two WCNs

At the same time, $G1$ also forwards the outer join request of $node_9$ to source S with the path ($S \rightarrow node_3 \rightarrow G1$). After that, S receives the request then stores the path in its GMT, and starts sending the data packets to $G1$. This turns $node_3$ to *forwarder* in the multicast session. As $G1$ receives the data packets, it relays the packets to $G2$ through the Internet backbone. Finally, $G2$ creates its GMT and forwards the data packets to $node_9$.

c) Leaving a multicast session and tree maintenance

A multicast receiver can leave the multicast session at any time by sending MROUTE_REQUEST with “leave flag” message towards the multicast source node, $source_s$, via a source-registered gateway. Once this gateway receives the leave request, it prunes and removes the node from the tree. Then, it updates the GMT and informs the source node, $source_s$, to delete the corresponding entry in the GMT of $source_s$.

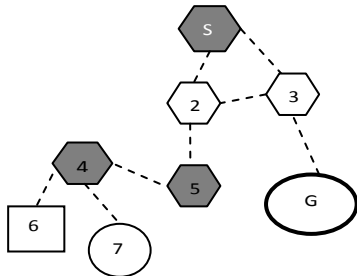


Fig. 6. Multicast tree reconstructed by MGMR inside a WCN

Based on the up-to-date GLT, the gateway checks the connectivity of all forwarder nodes on the multicast SPT tree upon the reception of each ROUTE_UPDATE. If a forwarder or a forwarder_receiver is found disconnected, the gateway reconstructs the SPT tree and passes the new paths to the source using one or multiple MROUTE_REPLY with “error flag”. The multicast source then contacts the affected nodes as it receives the error report, also by sending MROUTE_REPLY with “notify flag” message to complete this repair process. Fig. 6 illustrates an example of multicast tree repair. When a forwarder $node_1$ moves away and becomes disconnected, the tree is rebuilt. Hence, compared to Fig. 5, a former leaf_receiver, $node_5$, turns into a forwarder_receiver to

take over forwarding packets to $node_4$ in this particular case. Table 1 lists the major features of the proposed MGMR algorithm with comparison to the other popular multicast algorithms, such as MAODV [13], ODMRP [11], and GLBM [12]. In MGMR, the gateway actively broadcasts MHM messages towards the Internet backbone, and receivers join the multicast tree on demand. Further, as most nodes in WCNs exhibit low mobility, the cost to setup the proactive maintenance for the multicast tree or network topology is much lower than that of ad hoc networks, which necessitates the implementation of hybrid multicast. The hybrid mechanism of MGMR reduces the route discovery overhead, which is not the case in the existing on-demand driven multicast algorithms.

5 PERFORMANCE EVALUATION

The performance of GMR and MGMR has been evaluated and analysed thoroughly by using the NS-2 simulator [30] and considering the important QoS evaluation metrics including the *average end-to-end delay*; *average delay jitter*; *average packet delivery ratio* (PDR); *throughput*; and *probing overhead number*. Both the average end-to-end delay and average jitter are used to measure the algorithm’s capability in handling instant communications. Table 2 shows the simulation parameters of three network scenarios to be investigated: a small-size network of 50 WCN nodes uniformly distributed over $1000m \times 1000m$ area, a medium-size network of 100 WCN nodes over a $2000m \times 2000m$ area and a large-size network of 200 WCN nodes over $4000m \times 4000m$ area. Among all WCN nodes, 80% of them are set to be stationary and the rest are mobile. The gateway is placed relatively in the middle of the network. Each algorithm is simulated on 10 different randomly generated topologies, for each of which the simulation was run for 400 seconds. Notice that the same simulation environment/parameters adopted here were widely used by many exiting studies [9], [10], [11]. In the simulation of small size and medium size networks, the source of the multicast session transmits at a constant bit rate (CBR) from 50 to 200 packets/s (50, 100, and 200) in order to generate the data rate from 25 to 100Kbytes/s (25Kbytes/s, 50Kbytes/s, 100Kbytes/s). To examine further the capability of the proposed algorithm in large size networks, the source of the multicast session transmits at a constant bit rate (CBR) from 50 to 200 packets/s (50, 100, 150, 200) in order to generate the data rate from 25 to 100Kbytes/s (25Kbytes/s, 50Kbytes/s, 150 Kbytes/s, 100Kbytes/s). Therefore, the various data rates are used to verify whether or not the algorithms are capable of driving the applications with both high bandwidth and low bandwidth demand. The transmission power of the routers is set constant at 20 dBm; the data transmission rate at the physical layer is set to 54 Mbit/s; the two-ray propagation model is used as the radio propagation range and is set to 250m; the data packet size excluding the header size is 512Kbytes. The average performance results of MGMR, ODMRP, and GLBM normalized with respect to those of the original MAODV over all topolo-

gies are shown. In the figures below, the simulation results are plotted through the increase of data rate, i.e., the discriminative performance of each algorithm is examined versus various data rates.

TABLE 2
THE SIMULATION PARAMETERS

Network size	50 nodes over a 1000m × 1000m area, 100 nodes over a 2000m × 2000m area 200 nodes over a 4000m × 4000m area
Router-trans-power	20 dBm
Radio propagation range	250 m
Transmission rate	54 Mbits/s
Physical layer protocol	PHY802.11g
Packet size (excluding header size)	512 bytes
Queue size at routers	50 Kbytes
Unicast sender's rate	{50,100,200} packets/s
Multicast receiver size	{10,20,30} receivers in 50-node network {20,40,60} receivers in 100-node network {40,80,120} receivers in 200-node network
Multicast sender's rate	{50,100,200} packets/s {50,100,150,200} packets/s (200-nodes)
Simulation duration	400 seconds

5.1 Unicast communication with GMR

The first set of experiments intends to examine the performance of GMR with 100 nodes in a 2000m × 2000m area. In the simulation experiments, GMR-IFQ, GMR-PRED, AODV-HOP, and HWMP are implemented. GMR-IFQ denotes GMR with *Load Count* routing metric to control the QoS status in the central gateway, GMR-PRED denotes GMR with the Traffic Prediction Method enabled and AODV-HOP is referred to as the AODV based on hop count metric. We investigate the performance results of GMR-IFQ, GMR-PRED, and HWMP normalized with respect to those of the original AODV-HOP.

TABLE 3
THE PERFORMANCE OF GMR, AODV AND HWMP

Algorithms	Average End-to-End Delay	Average Jitter	Average Packet Delivery Ratio
GMR-IFQ	0.497	0.601	1.245
GMR-PRED	0.484	0.589	1.258
HWMP	2.996	1.871	1.134

Table 3 reveals that applying the Traffic Prediction Method helps GMR to achieve the highest packet delivery ratio. This demonstrates that the GMR-PRED provides a higher percentage of the successful packet transmission from the source to destination. The results illustrate that the packet loss of GMR-IFQ is slightly higher than that of GMR-PRED. However, both GMR-PRED and GMR-IFQ exhibit good QoS provisioning compared to other routing protocols due to reduction of flood caused by the topology update. In GMR, the gateway maintains the up-to-date information of the dynamic network topology and selects the optimal path by using the advanced routing metrics. The average end-to-end delay is largely decreased in GMR compared to HWMP and AODV-HOP as shown in Table 3 where GMR-PRED provides the lowest

delay among the four routing algorithms. The results confirm that both the leaf-to-gateway update mechanism and Traffic Prediction Method are used to help GMR avoid the busy path. In contrast, a cache-and-forward mechanism is functionalized in HWMP as each node receives route reply from the destination node and stores the path for future use until the expiration time is reached. This may yield the outdated path that can be selected for transmitting data since the forwarder node does not always keep the up-to-date best path. A good path provides stable traffic relay services continuously, whereas a poor path provides unsteady services in data packet transmission. A high average jitter is intolerable in services such as gaming and online conference applications because the instant communication always dominates in these applications. The results of the average jitter in Table 3 show that GMR-PRED exhibits the lowest jitter time which is followed by GMR-IFQ closely. In both GMR-PRED and GMR-IFQ, the results of the average jitter are reduced at least 40% compared to those of AODV-HOP and HWMP.

5.2 Multicast communication with MGMR

This set of experiments aims to compare MGMR with GLBM, MAODV and ODMRP in terms of the efficiency of multicast communications. The simulation results for 50-node WCNs are presented in Figs. 7 to 10. Figs. 11 to 14 depict the performance results for medium-size networks (100-node WCNs). In addition, Figs. 15 to 18 also illustrate the performance results for large-size networks (200-node WCNs) with sender's data rate at {50,100,150,200} packets/s which can further show the scalability of MGMR.

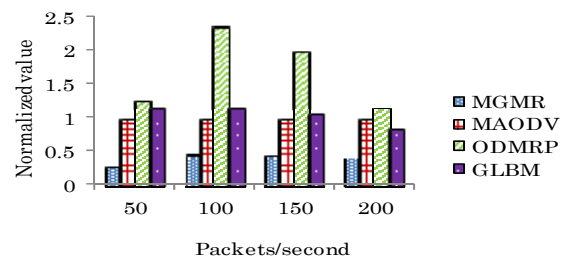


Fig. 7. Average rationed-to-end delay in 50-nodes

First, we can see that the delay of GLBM and MAODV is slightly lower than that of MGMR subject to the data rate 50 and 200 packets per second, respectively. As shown in Figs. 7, 11 and 15, MGMR behaves stably in all situations and generally achieves the lowest average average end-to-end delay, thus ensures the primary prerequisite of real-time communications and further confirms that MGMR provides a low delay in building the multicast tree. More importantly, a high average jitter is intolerable in services such as gaming and video application because the instant communication is always highly sensitive to latency. On the other hand, Fig. 8 shows that the average jitter is re-

duced 50% by MGMR compared to its counterparts which guarantees the consistency of the instant packet transmission.

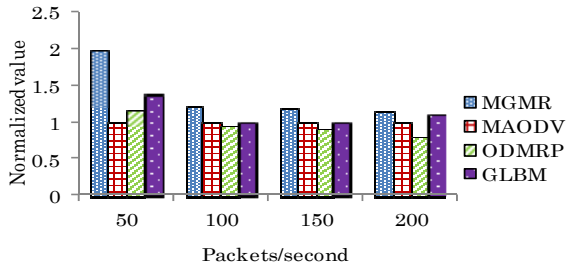


Fig. 8. Average jitter in 50-nodes WCNs

It was found that the low delay and jitter exhibited by MGMR is due to the leaf-to-gateway update mechanism that can avoid the busy paths. Figs. 9 (50 nodes), 13 (100 nodes) and 17 (200 nodes) also confirm that MGMR has the highest PDR. This is particularly important to guarantee the packet delivery quality in multicast video based applications. As plotted in Fig. 10, MGMR achieves the higher throughput than its counterparts. Figs. 9 (50 nodes), 15 (100 nodes) and 19 (200 nodes) also confirm that MGMR has the highest PDR.

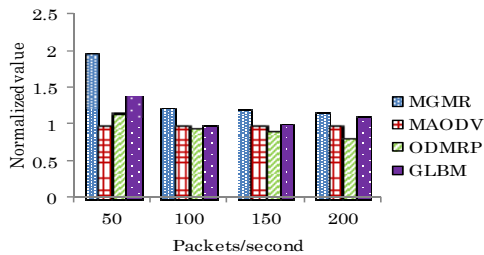


Fig. 9. Average packet delivery ratio in 50-nodes WCNs

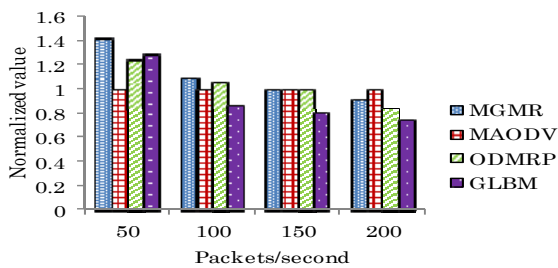


Fig. 10. Average throughput in 50-nodes WCNs

This is particularly important to guarantee the packet delivery quality in multicast video based applications. As plotted in Fig. 10, MGMR achieves the higher throughput than its counterparts. In addition, we have considered different network sizes and found that MGMR outperforms its counterparts, especially in large-size networks, *i.e.*, MGMR has shown high scalability in different operating conditions. As for the probing overhead, Table 4 shows different routing overhead values incurred by

the four routing schemes and reveals that MGMR causes additional probing overhead, but this does not affect its extraordinary performance.

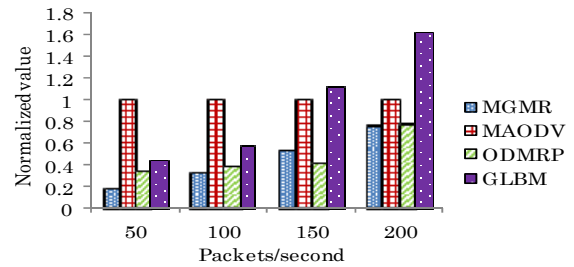


Fig. 11. Average end-to-end delay in 100-nodes WCNs

In the second row of Table 4, we observe that MGMR for small-scale networks only has 4.3% and 3.2% higher overhead than MAODV and GLBM, respectively. However, MGMR generates much lower overhead compared to ODMRP. We consider the total overhead size as the total probing packets in bytes and observe that MGMR has higher overhead than MAODV and GLBM respectively, whereas MGMR generates much lower overhead compared to ODMRP as shown in Table 4. Since the total overhead size is very small against the total network throughput, the leaf-to-gateway update of MGMR does not affect significantly the throughput gains. More importantly, it is shown in Table 4 that in three network sizes, 50 nodes, 100 nodes and a large network with 200 nodes, the probes overhead caused by MAODV, GLBM, ODMRP and MGMR increases, from 50 to 100 nodes, with the percentage 127%, 68%, 47.7%, 33.3%, respectively.

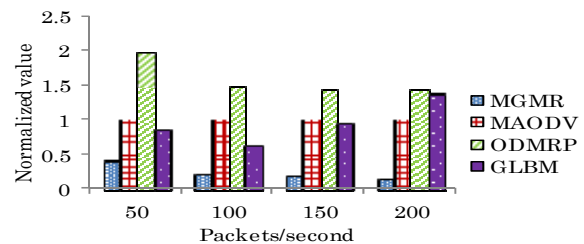


Fig. 12. Average jitter in 100-nodes WCNs

The increasing rate from 100 nodes to 200 nodes are 148.0%, 56.8%, 65.2%, and 23.6% respectively and the overall increasing rate from 50 nodes to 200 nodes are 463.6%, 163.6%, 144.0%, and 64.8%, respectively. This confirms that the probing overhead of MGMR is more stable through the changing of network size and traffic loads, where the maintenance cost of MGMR stays steadily even for larger networks which is not the case in MAODV, GLBM and ODMRP. The gateway based routing allows more possible paths for selection, offering more flexibility, and thus enhances the overall performance. This mechanism provides the low tree construction overhead and low communication time, *i.e.*, delay and jitter, confirming

the fact that both leaf-to-gateway update mechanisms enable MGMR to avoid the busy path.

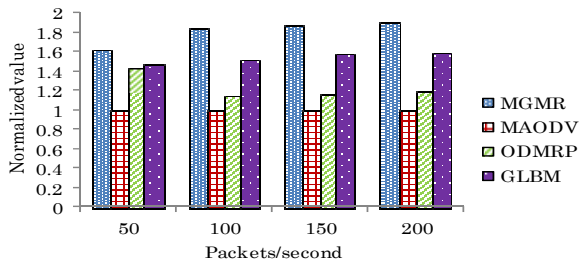


Fig. 13. Average packet delivery ratio in 100-nodes WCNs

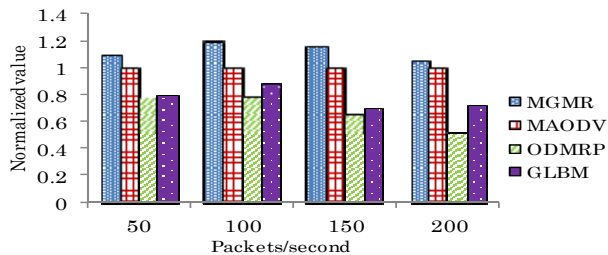


Fig. 14. Average throughput in 100-nodes WCNs

TABLE 4

THE COMPARATIVE PERCENTAGE OVERHEAD

Multicast algorithm	MAO-DV	GLBM	OD-MRP	MGMR
%Overhead -50 nodes	1.1	2.2	13.4	5.4
%Overhead-100 nodes	2.5	3.7	19.8	7.2
% Increasing rate 1	127	68	47.7	33.3
%Overhead-200 nodes	6.2	5.8	32.7	8.9
% Increasing rate 2	148.0	56.8	65.2	23.6
% Increasing rate 3	463.6	163.6	144.0	64.8

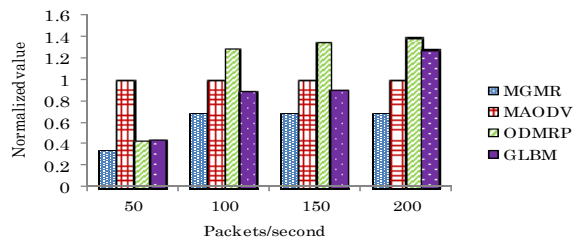


Fig. 15. Average end-to-end delay in 200-nodes WCNs

This onwards proves the less prominent throughput the results of MGMR in high data rate where the probes from leaf nodes are more likely to be dropped under high interference circumstances and the gateway does not have instant condition of fresh links from the mobile nodes. Additionally, the good performance of MGMR is also gained from the routing metric, *Load Count*. Figs. 11 and 15 show that the average end-to-end delay of MGMR is more stable and continuously the lowest among the four algorithms, demonstrating the capability of MGMR in medium-scale and large-scale networks. More im-

portantly, Figs. 12 and 16 show that the average jitter is reduced at least 115% and 171% respectively by MGMR in medium and large networks compared to its counterparts, thus guarantees high quality streaming. This is required to facilitate QoS aware applications. Besides, as plotted in Figs. 10, 14 and 18, MGMR achieves the higher throughput than its counterparts except MAODV with the data rate of 200 packets per second. This is quite interesting because intuitively one would expect MGMR to perform better than all others, since the PDR results of MGMR is always higher than MAODV. The reason for such an outcome is the high node density of small size mesh networks in which the high data rate typically generates high interference, resulting in more complexity and difficulty in updating the instant link status from leaf nodes in MGMR.

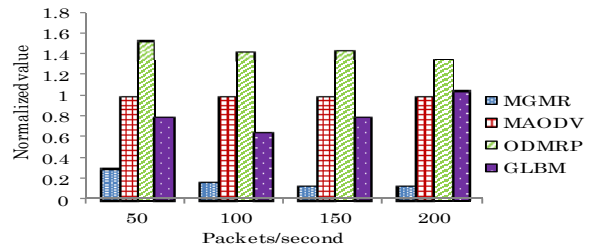


Fig. 16. Average jitter in 200-nodes WCNs

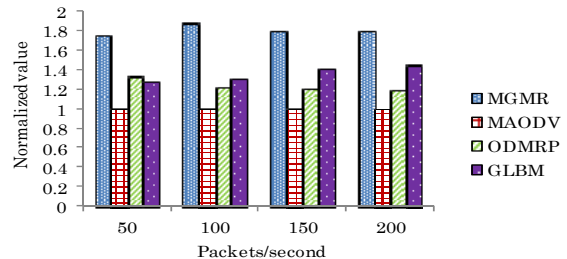


Fig. 17. Average packet delivery ratio in 200-nodes WCNs

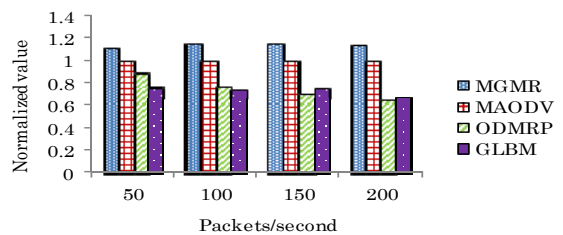


Fig. 18. Average throughput in 200-nodes WCNs

Moreover, we have also conducted experiments using 400 and 800 WCN nodes over $4000m \times 4000m$ area and we have obtained the similar results as 200 nodes.

6 CONCLUSION AND FUTURE WORK

This paper presents a new multicast strategy to handle QoS-aware applications in WCNs. Firstly, GMR is developed to promote the routing capability and gateway inclusion in WCNs. Based on GMR, MGMR is proposed as a new multicast routing algorithm. To the best of our

knowledge, MGMR is the first of its kind that considers both the gateway role along with the multicast based quality-aware applications. The MGMR allows the routing management in the gateway together with the leaf-to-gateway update mechanism to optimize the multicast communication. The performance results show that GMR and MGMR significantly outperform their counterparts, in terms of several important QoS performance metrics. In this study, the single-channel communication in WCNs is considered. Thus, for future research, we plan to investigate the capability of the proposed scheme at higher data-carrying capacity. We anticipate that our scheme can be implemented in multi-channel based multicast algorithms, which could provide a concrete basis for a number of interesting extensions.

REFERENCES

- [1] P. A. Frangoudis and G. C. Polyzos, "Wireless Community Networks: An Alternative Approach for Nomadic Broadband Network Access," *IEEE Comm. Magazine*, vol. 49, no. 5, pp. 206-213, 2011.
- [2] S. Vural, D. Wei and K. Moessner, "Survey of Experimental Evaluation Studies for Wireless Mesh Network Deployment in Urban Areas Towards Ubiquitous Internet," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 223-239, 2012.
- [3] Ai Chen, OH Columbus, Lee Dongwook, G. Chandrasekara and P. Sinha, "HIMAC: High Throughput MAC Layer Multicasting in Wireless Networks," 2006 IEEE Int. Conf. Mobile Ad hoc and Sensor Systems (MASS), pp. 41 – 50, 2006.
- [4] Yinan Li and Ing-Ray Chen, Dynamic Agent-based Hierarchical Multicast for Wireless Mesh Networks, *Ad Hoc Networks*, Elsevier, vol. 11, no. 6, pp. 1683–1698, 2013.
- [5] N. Kumar, N. Chilamkurti and Jong-Hyouk Lee, "A novel minimum delay maximum flow multicast algorithm to construct a multicast tree in wireless mesh networks," *Computers & Mathematics with Applications*, vol. 63, no. 2, pp. 481–491, 2012.
- [6] S. Roy, D. Koutsonikolas, S. Das, and Y.C. Hu, "High-Throughput Multicast Routing Metrics in Wireless Mesh Networks," in *Proceedings of the 26th IEEE International Conference on Distributed Computing Systems*, pp.48, 2006.
- [7] G. Zeng, B. Wang, Y. Ding, L. Xiao, and M. Mutka, "Multicast Algorithms for Multi-Channel Wireless Mesh Networks," *IEEE Trans. Parallel and Distributed Systems*, vol. 21, no. 1, pp. 86-99, 2010.
- [8] J. Camp and E. Knightly, "The IEEE 802.11s Extended Service Set Mesh Networking Standard," *IEEE Communications Magazine*, vol. 46, no. 8, pp. 120-126, 2008.
- [9] I. F. Akyildiz and X. Wang, "Wireless Mesh Networks," Wiley, ISBN 9780470032565 (H/B), 1st edition, 2009.
- [10] S.J. Lee, W. Su, and M. Gerla, "On-Demand Multicast Routing Protocol in Multihop Wireless Mobile Networks," *J. Mobile Networks and Applications*, vol. 7, no. 6, pp. 441-453, 2002.
- [11] L. Zhao, Ahmed Y. Al-Dubai, Geyong Min, "GLBM: A New QoS Aware Multicast Scheme for Wireless Mesh Networks," *Journal of Systems and Software*, vol. 83, no. 8, pp. 1318-1326, 2010.
- [12] E. Royer, and C. E. Perkins, "Multicast Operation of the Ad-hoc On-Demand Distance Vector Routing Protocol," in *Proceedings of the 5th ACM/IEEE Annual Conference on Mobile Computing and Networking*, pp. 207-218, 1999.
- [13] M. Manshaei, et al, "On Wireless Social Community Networks," *IEEE INFOCOM*, 2008.
- [14] L. Zhao, A. Al-Dubai and G. Min, "Cross-Layer Neighborhood Load Routing for Wireless Mesh Networks," *Proc. the 24th Int. Parallel and Distributed Processing Symposium, IEEE/ACM-IPDPS*, Atlanta, USA, 2010.
- [15] G. Min, Y. Wu, A. Y. Al-Dubai, "Performance Modelling and Analysis of Cognitive Mesh Networks," *IEEE Transactions on Communications*, vol. 60, no. 6, pp. 1474-1478, 2012.
- [16] C. Szabo, Z. Horvath and K. Farkas, "Wireless Community Networks: Motivations," *Proc. 7th Int. Wireless Internet Conference on Design & Business Models (ACM WiCON)* 2007.
- [17] K. Oh and C.Y. Lee, "Multicast Routing Protocol with Low Transmission Delay in Multi-Rate, Multi-Radio Wireless Mesh Networks," in *Proc. of 2010 IEEE International Conference on Communications (ICC)*, pp. 1-6, May 2010.
- [18] Mesh Networks, Decentralized, Self-Forming, "Self-Healing Network", MOTOROLA Position Paper, 2005.
- [19] J. Sun, Y. Liu, H. Hu and D. Yuan, "On-Demand Coding Aware Routing in Wireless Mesh Networks," *the Journal of China Universities of Posts and Telecommunications*, vol.17, no. 5, pp. 80-86, 2010.
- [20] E. Pourfakhar and A. Rahmani, "A Hybrid QoS Multicast framework-based protocol for wireless mesh networks" *Comp.Communications*, vol. 33, no.17, pp. 2079-2092, 2010.
- [21] V. C. M. Borges, M. Curado, and E. Monteiro, "The Impact of Interference-Aware Routing Metrics on Video Streaming in Wireless Mesh Networks," *Ad Hoc Networks*, vol. 9, no. 4, pp. 652-661, 2011.
- [22] G. Zeng, B. Wang Y. Ding, L. Xiao and M. W. Mutka, "Efficient multicast algorithms for multichannel wireless mesh networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 21, no. 1, pp. 86–99, 2010.
- [23] Y. Li and I. Chen, "Hierarchical Agent-Based Secure Multicast for Wireless Mesh Networks", *Proc. IEEE International Conf. Communications (IEEE ICC)*, 2011.
- [24] List of wireless community networks by region: Available in: http://en.wikipedia.org/wiki/List_of_wireless_community_networks_by_region.
- [25] U. T. Nguyen, "On Multicast Routing in Wireless Mesh Networks," *Elsevier Journal of Computer Communications*, vol. 31, no 7, pp.1385–1399, 2008.
- [26] O. Baghban Karimi, J. Liu and Z. Li, "Multicast in Multi-channel Wireless Mesh Networks", *NETWORKING 2010, Lecture Notes in Computer Science*, vol. 6091, pp. 148-159, 2010.
- [27] Y. Andreopoulos, N. Mastrorade, and M. van der Schaar, "Cross-Layer Optimized Video Streaming Over Wireless Multihop Mesh Networks", *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 11, pp. 2104 – 2115, 2006.
- [28] J. Yuan, Zongpeng Li, Wei Yu and Baochun Li, "A Cross-Layer Optimization Framework for Multihop Multicast in Wireless Mesh Networks", *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 11, pp.2092-2103, 2006.
- [29] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to Algorithms," 2nd Edition, *The MIT Press*, 2001.
- [30] The Network Simulator NS2: <http://www.isi.edu/nsnam/ns/>
- [31] J. Hwang and C. Lee, "Effective Video Multicast Using SVC with Heterogenous User Demands over TDMA-BASED Wireless Mesh Networks, ", *IEEE Transactions on Mobile Computing*, vol. 12, no.5, pp. 984-994, May 2013.
- [32] W. Yang and W. Hong, "A cross-layer Optimization for Maximum-revenue-based Multicast in Multichannel Multiradio Wireless Mesh Networks," *Int. Journal of Communication System*, Article first published online in Wiley Online Library, DOI: 10.1002/dac.2535, March 2013.
- [33] C.Perkins, E. Belding-Royer and S. Das,"Ad hoc On-Demand Distance Vector (AODV) Routing," IETF. RFC 3561.
- [34] S. Surlees, S. Anand John Francis, "Secure Multicasting Protocols in Wireless Mesh Networks: A Survey," *Computational Intelligence, Cyber Security and Computational Models, Advances in Intelligent Systems and Computing*, Springer, vol. 246, pp 245-256, 2014.
- [35] Fu Weihuang and D.P Agrawal, "Capacity of Hybrid Wireless Mesh Networks with Random APs," *Mobile Computing, IEEE Trans. Mobile Computing*, vol.12, no.1, pp.136-150, Jan. 2013.
- [36] Weiyei Zhao and Jiang Xie, "DoMaIN, A Novel Dynamic Location Management Solution for Internet-Based Infrastructure Wireless Mesh Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol.24, no.8, pp.1514-1524, Aug. 2013.
- [37] Wei-Liang Shen, Chung-Shiuan Chen, Kate Ching-Ju Lin and Kien A. Hua, "Autonomous Mobile Mesh Networks," *IEEE Trans. Mobile Computing*, vol.13, no.2, pp.364-376, Feb. 2014.



Ahmed Al-Dubai is Reader of Networking and High Performance Computing in the School of Computing at Edinburgh Napier University, United Kingdom. He received the PhD degree in Computing from the University of Glasgow, United Kingdom, in 2004, and the MSc and B.Sc. degrees in Computer Science from Al Al al-Bayt University and Mutah University, Jordan in 1996 and 1999, respectively. His research interests include Communication Algorithms, Computer Networks, Mobile Communications, Ubiquitous Computing, Modelling and Performance Engineering and Future Internet. Dr Al-Dubai is Fellow of the British Higher Education Academy and Senior Member of the IEEE.

He received the PhD degree in Computing from the University of Glasgow, United Kingdom, in 2004, and the MSc and B.Sc. degrees in Computer Science from Al Al al-Bayt University and Mutah University, Jordan in 1996 and 1999, respectively. His research interests include Communication Algorithms, Computer Networks, Mobile Communications, Ubiquitous Computing, Modelling and Performance Engineering and Future Internet. Dr Al-Dubai is Fellow of the British Higher Education Academy and Senior Member of the IEEE.



Liang Zhao is currently a lecturer in the College of Computer Science at the Shenyang Aerospace University, China. Before that, he served as an associate senior researcher in Hitachi (China) Research and Development Corporation from 2012 to 2014. He received his PhD degree from the School of Computing, Edinburgh Napier University in 2011. His research interests include wireless mesh networks (WMNs), VANETs, MANETs, group communications, routing metric, QoS, Intelligent Transport Services, and Telematics.

He received his PhD degree from the School of Computing, Edinburgh Napier University in 2011. His research interests include wireless mesh networks (WMNs), VANETs, MANETs, group communications, routing metric, QoS, Intelligent Transport Services, and Telematics.



Albert Y. ZOMAYA is currently the Chair Professor of High Performance Computing & Networking in the School of Information Technologies, The University of Sydney. He is also the Director

of the Centre for Distributed and High Performance Computing which was established in late 2009. Professor Zomaya published more than 500 scientific papers and articles and is author, co-author or editor of more than 20 books. He is the Editor in Chief of the IEEE Transactions on Computers and Springer's Scalable Computing and serves as an associate editor for 22 leading journals, such as, the ACM Computing Surveys and Journal of Parallel and Distributed Computing. Professor Zomaya is the recipient of the IEEE Technical Committee on Parallel Processing Outstanding Service Award (2011), the IEEE

Technical Committee on Scalable Computing Medal for Excellence in Scalable Computing (2011), and the IEEE Computer Society Technical Achievement Award (2014). He is a Chartered Engineer, a Fellow of AAAS, IEEE, IET (UK). Professor Zomaya's research interests are in the areas of parallel and distributed computing and complex systems.



Geyong Min is a Professor of High Performance Computing and Networking in the Department of Mathematics and Computer Science within the College of Engineering, Mathematics and Physical Sciences at the University of Exeter, United Kingdom. He received the PhD degree in Computing Science from the University of Glasgow, United Kingdom, in 2003, and the B.Sc. degree in

Computer Science from Huazhong University of Science and Technology, China, in 1995. His research interests include Future Internet, Computer Networks, Wireless Communications, Multimedia Systems, Information Security, High Performance Computing, Ubiquitous Computing, Modelling and Performance Engineering.