Title	DAPP: A Delay-Aware Packet Prioritisation Scheme for VoIP in Wireless Multi-hop Networks
Authors(s)	Olariu, Cristian; Hava, Adriana
Publication date	2016-07-13
Conference details	The 9th IFIP Wireless and Mobile Networking Conference (WMNC 2016), Colmar, France, 11-13 July 2016
Publisher	IEEE
Item record/more information	http://hdl.handle.net/10197/7692
Publisher's statement	© © 2016 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

DAPP: A Delay-Aware Packet Prioritisation Scheme for VoIP in Wireless Multi-hop Networks

Cristian Olariu
School of Computer Science, University College Dublin
Belfield, Dublin 4, Ireland
cristian.olariu@ucd.ie

Adriana Hava
School of Electronic Engineering, Dublin City University,
Glasnevin, Dublin 9, Ireland
adriana.havaolariu@dcu.ie

***Abstract*—In this paper we propose a mechanism entitled Delay-Aware Packet Prioritisation (DAPP) that works at WiFi packet queue level. DAPP enqueues packets according to their time spent in the network, such that more delayed packets are enqueued towards the head of the queue. We study the effects of such an approach in wireless multi-hop scenarios where VoIP packets with various delay impairments reach a congested node. Packets' delay builds up as congestion increases and we show that DAPP improves a network's VoIP capacity by distributing the negative effects of delay to those VoIP packets that so far travelled in good network conditions. This is to the advantage of those VoIP packets already negatively affected by network conditions, as they are allowed to pass a congested node with less further delay accumulation. We use extensive NS-3 simulations where a wide range of network impairments affect VoIP calls, and we demonstrate that DAPP is able to improve the VoIP capacity and fairness among VoIP calls in a wireless multi-hop network.**

I. Introduction

Quality of Service (QoS) degradation often occurs due to network congestion, and packet queues are put in place in order to address this issue. Packets are temporarily buffered in these queues until network conditions allow for packet transmissions to resume. Queue overflowing can happen when too many packets accumulate in the size-limited packet queue, causing these to be ultimately dropped. However there are situations where the packet arrival rate fluctuates around the maximum serving rate of a queue. As a result, packet queues will always have a number of packets waiting for the scheduler to forward them to the next hop. This waiting time ranges from a few microseconds to entire seconds, however some applications are resilient to this delay, e.g. file transfers, while other are heavily affected by it, e.g. VoIP calls.

In wireless multi-hop networks, there are forwarding nodes where IP traffic merges from different network paths, and these nodes are particularly prone to congestion. This mix of traffic is composed of packets that are affected by delay (e.g. queueing delay) along the travelled path, and the amount of this delay can greatly vary.

Such a situation is presented in Figure 1 which depicts an arbitrary topology of a wireless multi-hop network with various VoIP calls deployed on the nodes in the network. Every node depicts the number of VoIP calls it handles, and every subset of the network is capable of forwarding the VoIP traffic received. The red node in Figure 1 is a bottleneck as the VoIP traffic handled by that node exceeds its capacity. VoIP sessions belonging to a specific subset of wireless nodes are affected by delay (e.g. Δ_2) differently than the sessions belonging to another subset (e.g. Δ_3). Thus, when these sessions merge in the congested node, VoIP packets will already be differently affected by delays.

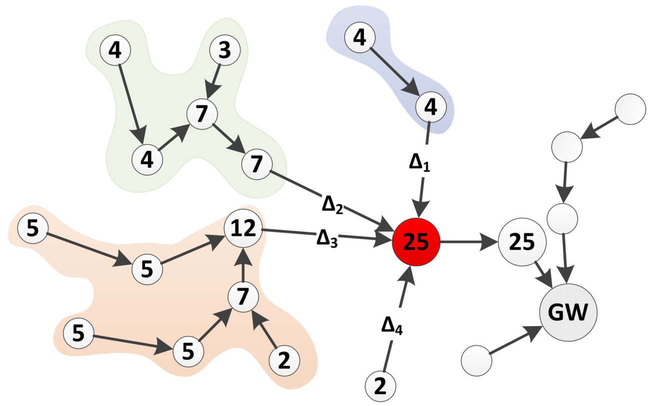

Figure 1: A wireless multi-hop network running VoIP calls

In this paper we make the argument that packets should be prioritised inside the same queue based on the delay suffered thus far in their network path, and this approach is implemented in DAPP, our proposed solution. Such a practice should allow 'older' packets to travel unimpeded until these packets meet even older packets in which case priority must be given. In this way, the negative effects of the queuing delay is shared among applications. More specifically we study such effects on the QoS of VoIP which is one of the most time sensitive IP applications.

The remainder of this paper is structured as follows: Section II details existing solutions for provisioning QoS for VoIP. Section III describes our proposed DAPP algorithm while Section IV describes the experiments carried out in NS-3 and the results that validate DAPP. The paper is concluded in Section V.

II. Related Work

VoIP capacity in wireless networks has been extensively investigated for various VoIP codecs and different wireless communication protocols [1, 2]. However, in a multi-hop wireless network the VoIP capacity problem is more complex and has been addressed mostly by proposing Call Admission Control (CAC) solutions that protect existing calls in the network against call quality degradation when the network is at capacity. The study on Wireless Mesh Networks (WMNs) nominal capacity in [3] shows that gateways introduce hot-spots that are creating bottlenecks, as nodes closer to the gateway have to forward more traffic than the nodes further away. Thus, actually, in a wireless multi-hop network the capacity is determined by these bottlenecks, which are located in the vicinity of gateways. Our present study is focused on such bottleneck scenarios, and our attention is on queueing events that happen in a congested node's packet queue.

Some benefits of piggybacking QoS-related information along with VoIP packets is highlighted in one of our previous works [4]. We established that piggy-backing information in VoIP packets can be used to increase QoS of VoIP calls when used in conjunction with a CAC mechanism.

The importance of packet delay for improving VoIP capacity has been considered in previous works [5]. A MAC layer approach for balancing the uplink and downlink delay for VoIP traffic in a Wireless Local Area Network (WLAN) is proposed in [2].

In [6], the authors propose a WLAN solution for prioritising VoIP sessions in different MAC queues based on the end-to-end delay. NTP is employed in order to compute the mouth-to-ear delay. Flaithearta et al. [6] propose to define three queues for VoIP traffic and to map the VoIP sessions based on their MOS score. Each category has defined fixed Enhanced Distributed Channel Access (EDCA) parameters giving higher priority to queues containing VoIP sessions having a low MOS. In contrast to these works, our solution focuses on multi-hop networks and increasing the VoIP capacity by evenly distributing the queueing delay among the VoIP packets within the existing 802.11e VoIP queue. As our solution considers prioritising the packets based on their current delay instead of end-to-end delay, effective measures for improving VoIP QoS can be taken in a timely manner.

In this work, the 802.11e [7] standard is used to provide QoS differentiation for services. 802.11e defines four access categories to enable differentiated QoS through EDCA. By default, these queues operate in First-In-First-Out (FIFO) transmission mode. In FIFO mode the packets are sent in the order in which they arrived. In a previous work we show that the usage of a Push-In-First-Out (PIFO) queue can improve the QoS for multimedia traffic [4]. Typically, a PIFO queue requires more computational power than FIFO queues during the insertion phase, thus it has been considered practical only for theoretical analysis of QoS schedulers. However, the practical analysis carried by the authors in [8] proved that the current hardware is able to cope with the demands imposed by PIFO queues.

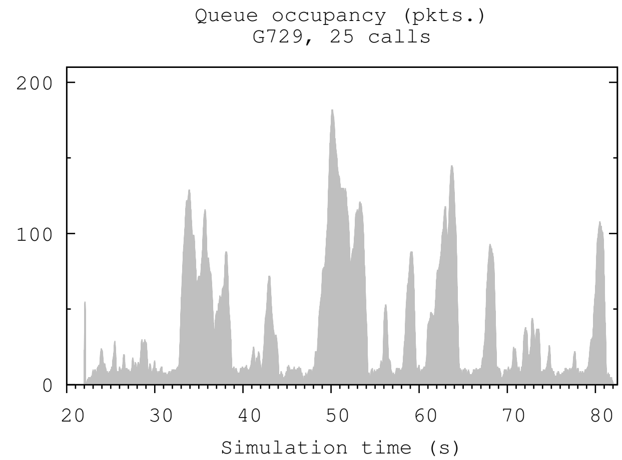

Figure 2: Queue occupancy

III. DAPP: The proposed Delay-Aware Packet Prioritisation Scheme

A. *Study of congested wireless multi-hop VoIP networks*

We study the effects of increased network load on wireless forwarding nodes and note that it is more likely that a small subset of nodes will become the network's bottleneck points, rather that finding all wireless nodes in a congested state. In a multi-hop scenario, the location of the bottleneck has been widely studied [3] and is not the scope of this paper. Hence we focus our attention on a particular wireless node belonging to a wireless multi-hop network, where multiple VoIP calls merge and their traffic amounts to a value beyond that node's capacity, as depicted in Figure 1.

We study the scenario where a node's packet queue is constantly busy storing packets and that situation can only occur around a node's forwarding capacity. We detected that for loads originated from a VoIP traffic mix with a fluctuating throughput, the queue occupancy is non-zero.

We simulated a scenario of 25 VoIP applications using G729 as codec over a WiFi link, with a call duration of 60 seconds, and the queue occupancy of the congested node is depicted in Figure 2. It can be observed that the queue occupancy never exceeds 200 packets thus no packet drop ensued, but neither was the queue completely empty at any time. For the duration of the simulation, the queue occupancy was on average 35 packets. This indicates that there are sufficient packets in the queue to create enough opportunities for older packets to get forwarded ahead of younger packets.

Figure 3 brings more arguments supporting our claim regarding non-boundary queue occupancy. We varied the number of calls in the same network configuration while using various VoIP codecs. The left-hand side plot in Figure 3 depicts the average queue occupancy against number of calls in the network, while the right-hand side plot presents the average queueing delay against the number of calls. The average queueing delay represents the amount of time a packet waited in the queue until it was transmitted.

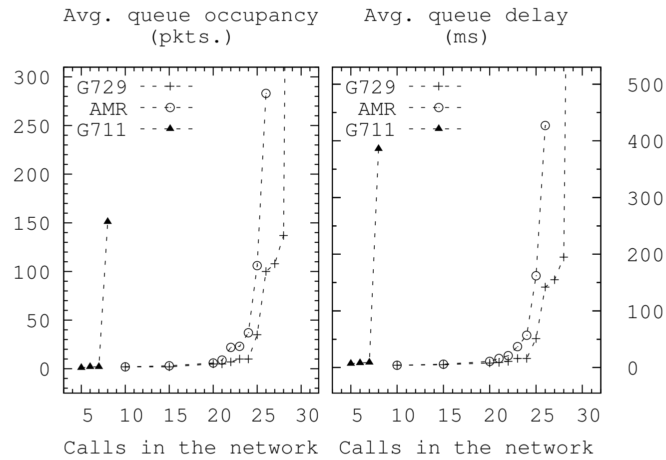

Figure 3: Queue Occupancy and Queueing Delay for various VoIP codecs

Algorithm 1: Inserting a Voice Over IP (VoIP) packet into the PIFO queue

```
input: p_in (incoming VoIP packet), Q (destination queue)
if Q == ∅ then
    Enqueue p_in;
    RETURN;
else
    for p ← Q.begin to Q.end do
        if getAge(p_in) > getAge(p) then
            if Q.length == Q.maxSize then
                drop Q.end;
            Enqueue p_in before this p;
            RETURN;
        else
            p++; // increment queue iterator
    if Q.length < Q.maxSize then
        Enqueue p_in, the youngest packet, at Q.end;
    else
        Drop p_in; //the youngest packet
    RETURN;
```

The queue occupancy plot in Figure 3 shows that for G711-based VoIP calls, when the number of VoIP calls in the network increases from seven to eight calls, the queue occupancy increases abruptly from three to around 150 packets in the queue, while the average queueing delay of those 150 packets is around 400 ms, which is a value unacceptable for VoIP calls [9]. For Variable Bit-Rate (VBR) codecs, such as AMR and G729, we observe that the queue occupancy increases in smaller steps, depicting a 'knee' trend around 23 to 25 VoIP calls. It can be seen that the queue occupancy varies between 10 and 100 packets in the queue with an average queueing delay lower than 160 milliseconds. This delay value corresponds to the threshold between acceptable and unacceptable in terms of end-to-end delay for VoIP calls. Figure 3 shows that for VBR traffic the node's transition to a congested state happens gradually with every added call, as opposed to G711 traffic which generates an abrupt transition to a highly congested state by adding just one extra call. Due to this abrupt transition pattern for CBR VoIP traffic, our analysis focuses on VBR VoIP calls.

B. DAPP enqueueing algorithm

DAPP's enqueueing algorithm is depicted in Algorithm 1, it runs on every wireless node and it is executed every time a new VoIP packet is received. The algorithm requires two parameters as input: the incoming VoIP packet p_{in} that has to be enqueued, and the VoIP queue Q where p_{in} will be enqueued. Q's existing packets are denoted with p.

If Q is empty, then p_{in} is simply enqueued, thus leading to a queue with one packet; else, the insertion position is determined by iterative comparisons between the delay of p_{in}'s and that of p's using $getAge()$. The $getAge(p_{in})$ method returns the delay of the incoming VoIP packet. $getAge(p)$ returns the delay that p arrived with to which it adds the queueing delay p accumulated so far in Q.

When p_{in}'s delay is higher than the delay of p's, then p_{in} is enqueued right before p, but not before Q's current occupancy is verified; if Q's current occupancy is equal to its maximum size, then the last packet of Q is dropped as to make space for the more delayed packet p_{in}. If after successive comparisons the end of Q is reached, it means that all the packets in Q are older than p_{in}. Then, if Q's current occupancy is less than its maximum allowed size, p_{in} is enqueued at the end of Q (as the least delayed or youngest VoIP packet in this queue). Otherwise, p_{in} is dropped as Q is full and all its packets are older than p_{in}.

As a results of this mechanism, at any moment the packets in Q are always ordered from the oldest to the youngest, where the oldest packet will be served first by the scheduler. DAPP ensures that older packets receive higher priority of transmission as compared to younger packets.

C. Discussion

DAPP relies on the fact that the cumulative queueing delay of VoIP packets is known or computable. Further we have a discussion how this can be achieved. We propose two possible ways for placing the cumulative delay value into a VoIP packet.

The first option is to use an existing packet header to accommodate the delay value, and we call it a delay field. The value of the delay field is initialised to zero at the ingress node, i.e. the first wireless forwarding node, which receives the VoIP packet. Just before the node's scheduler dequeues a VoIP packet for transmission, the current queuing delay is added to the total delay in the delay field, and the delay field is rewritten. Hence, at each wireless hop the delay field contains the total delay experienced by the VoIP packet since entering the multi-hop wireless network. This delay is the information used by Algorithm 1 to place VoIP packets in their designated queue.

The second option is creating a new proprietary packet header. The first option is more desirable, as there are many packet headers which support experimenter fields, allowing

Parameter	Value
Simulator	NS-3.22
Distance between nodes	125 m
WiFi Data Rate	1 Mbps
Propagation Model	LogDistancePropagationLossModel
Error Rate Model	YansErrorRateModel
Remote Station Manager	ConstantRateWifiManager
Call Type	Half-duplex
Call Duration	60 seconds
Voice codec	AMR and G.729a+VAD
Call Quality Assessment Method	E-Model [9]
Speech Model	ITU-T/P.59 [11]
VoIP capacity threshold	R=70 (MOS=3.6)

TABLE I: Simulation Setup

researchers to piggy-back some information over the network using existing packet headers.

Considering that the delay field stores the sum of the amounts of time VoIP packets spent in the voice queues of each wireless node, our solution does not rely on NTP synchronisation between wireless nodes. Previous research [10] identified that the propagation delay is deterministic, as it depends on the medium characteristics between the wireless nodes, while the queueing delay (i.e. delay due to the time packets spent waiting in the queue) is stochastic and predominant in comparison to the propagation delay in congested network conditions.

If a NTP solution is employed in the wireless network for synchronising the nodes, as in [6], the delay field can actually store the creation time of the packet. This means that every wireless node receiving a VoIP packet will be able to accurately compute the accumulated delay.

Finally, a note on the complexity of DAPP. Finding the position where a VoIP packet is to be enqueued is an operation of complexity O(N), where *N* is the total number of packets enqueued. In the worst case scenario (i.e. congested node) the algorithm has to make N trivial number comparisons. However, when the network is not congested, the queue will operate almost empty with sporadic occupancy bursts, hence the algorithm has to make fewer than N comparisons most of the time.

IV. Performance Evaluation

A. Simulation Settings

The simulation parameters are presented in Table I. We use the topology depicted in Figure 1 and focus on the congested node. The VoIP call quality is assessed in the next hop node in the same figure. The physical link is using a data rate of 1 Mbps in order to provoke queueing and being able to demonstrate its effect. We enabled the wireless MAC of the nodes to be QoS-aware by employing 802.11e. Our VoIP applications tag VoIP packets with the appropriate Type of Service (TOS) value as to ensure these VoIP packets will be enqueued in the AC_VO queue of the 802.11e MAC layer, and this AC_VO queue is given to Algorithm 1.

We establish a number of VoIP calls with a call duration of 60 seconds. The VoIP calls are modelled as VBR traffic with 32 bytes frame payload size and 20 ms packet generation[1], for AMR codec, and 20 bytes frame payload size and 20 ms packet generation for G729 codec[2], respectively. Each call is simulated realistically following the conversation model from ITU-Ts P.59 report [11]. The speech model proposed by ITU considers the amount of time the speaker is in a specific state and the probability to switch to another one, and these values are extracted from actual human conversations. This model was implemented and incorporated into NS-3[3], in order to generate realistic speech activity for VoIP calls deployed in the network.

We focused our analysis on the scenario where the queue occupancy follows a 'knee' trend as depicted in Figure 3. The 'knee' trend corresponds to network loads where the queue occupancy is non-zero, thus allowing DAPP to prioritise packets inside of the same queue. For our setup it means that we focus on scenarios with 23, 24 and 25 G729-based VoIP calls and, 23 and 24 AMR-based VoIP calls deployed in the network.

The VoIP call quality assessment is done using the E-Model [9], which is the most popular model to estimate speech quality and is based on network transmission parameters and the voice codec choice. E-Model produces a scalar in a range from 0 to 100 called the *Transmission Rating Factor (R)*, which is computed as described in [12, 13]. However, the Mean Opinion Score (MOS) value is more commonly used by researchers [14, 15] for assessing VoIP quality and its values are in a range from 1 to 5. Traditionally, the MOS values are obtained by subjective testing. As subjective testing is expensive and time consuming, a conversion formula is provided in [9], which converts R values to MOS values and some important thresholds are highlighted in Table II. Typically, from a QoS perspective a MOS of 3.6 or higher is considered as satisfactory for a VoIP session to be valid [16]. Thus, in this work, the capacity of the network is calculated counting the number of VoIP session with a MOS equal or higher than this threshold.

R-value (lower limit)	MOS (lower limit)	User Satisfaction
90	4.34	Very satisfied
80	4.03	Satisfied
70	3.60	Some users dissatisfied
60	3.10	Many users dissatisfied
50	2.58	Nearly all users dissatisfied

TABLE II: R to MOS correspondence from the E-Model for estimative assessment

B. Simulation results

Our algorithm shows benefits when delayed packets are mixed with less delayed packets in the same VoIP queue, as

[1] https://tools.ietf.org/html/rfc4867#section-3.6

[2] http://www.cisco.com/c/en/us/support/docs/voice/voice-quality/7934-bwidth-consume.html

[3] http://www.nsnam.org

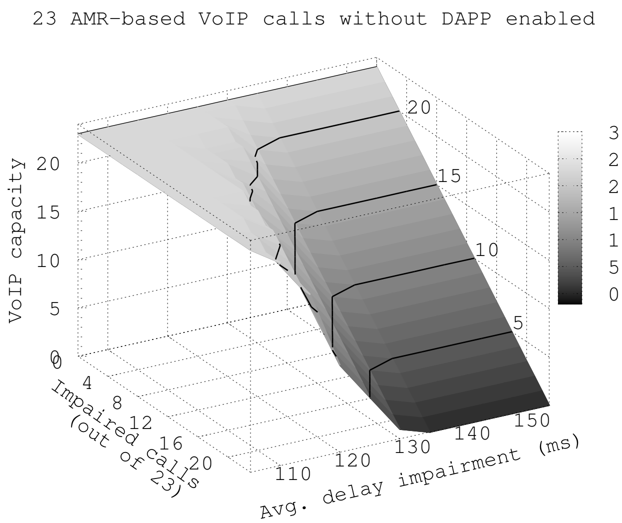

Figure 4: VoIP capacity for 23 simultaneous AMR calls

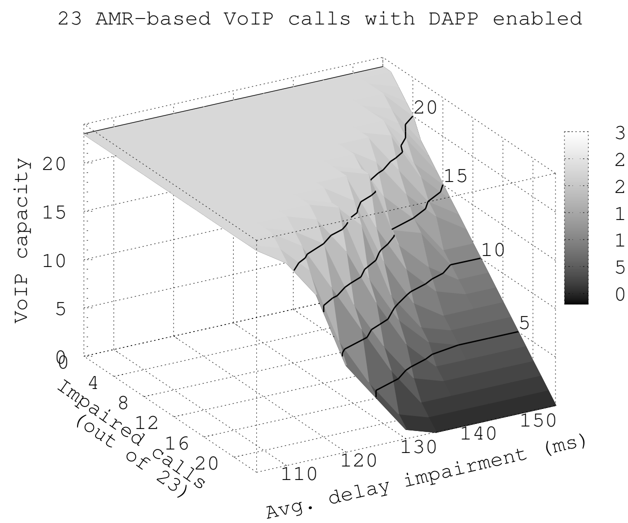

Figure 5: VoIP capacity for 23 simultaneous AMR calls with DAPP employed

discussed in section III-A. In order to simulate a mix of possible network delay impairments that can occur simultaneously on VoIP calls, we affect a subset of the total number of VoIP calls with a variable delay. This delay is in the range of 50 to 150 milliseconds. Additionally, we varied the number of calls affected by this delay from none to all calls affected by the delay. For each combination of delay impairment and number of calls affected by it we compute the network's VoIP capacity, which is the number of calls having an overall $MOS \geq 3.6$. We compare the VoIP capacity when our DAPP solution is employed against the baseline case, i.e. when packets are not prioritised based on the accumulated delay. We evaluate the delay fairness among sessions using Jain's fairness index.

1) VoIP capacity analysis for AMR calls: Figure 4 depicts the VoIP capacity for the scenario with 23 AMR-based VoIP calls deployed in the network where DAPP is not employed. Figure 5 depicts the VoIP capacity for the same setup, and DAPP is employed. The results in both figures are presented as 3D graphs, with the number of calls that are affected by delay out of a total of 23 calls on the Y-axis and on the X-axis the amount of delay incurred before the stream arrives at the congested node. The values on the Z-axis show the number of VoIP calls having a MOS higher than 3.6.

Comparing Figure 4 with Figure 5 it can be observed that DAPP extends the number of scenarios where all 23 VoIP calls are supported. The VoIP capacity improvement brought by DAPP over the baseline case is obtained by subtracting the values in these two pictures and it is depicted in Figure 6. Thus, in Figure 6 the Z-axis values show the number of additional calls supported when DAPP is employed as compared to the baseline case. The contour lines on the surface of the 3D graph highlight the VoIP capacity improvement ranging from one extra call supported to 14 extra calls. Thus, there are scenarios when the baseline case supports 9 calls (i.e. 40% of

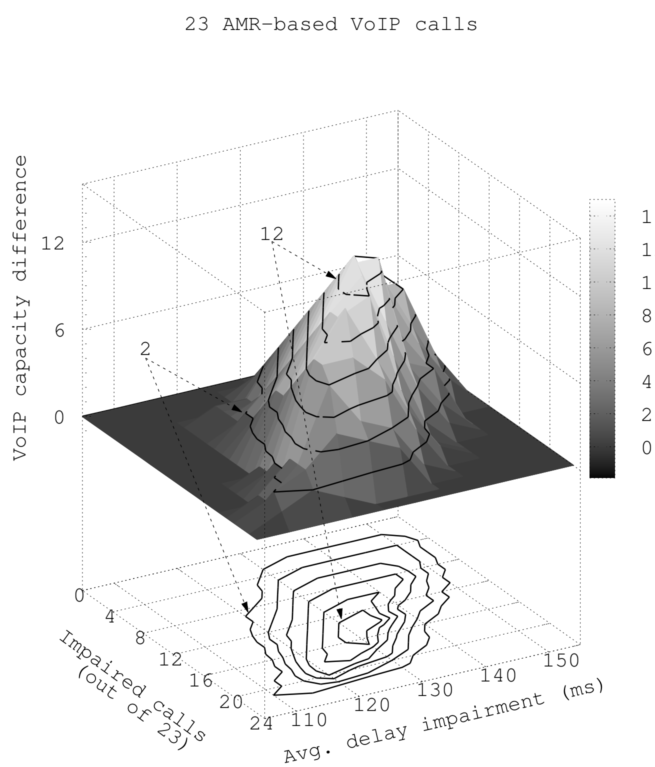

Figure 6: VoIP capacity improvement for 23 simultaneous AMR calls

all 23 VoIP sessions are supported), while employing DAPP the network supports all 23 VoIP calls (i.e. 100% VoIP calls

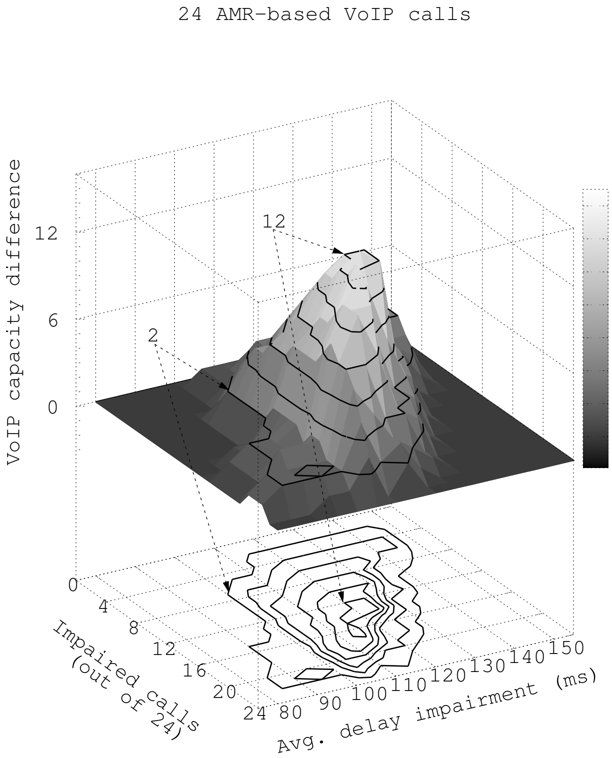

Figure 7: VoIP capacity improvement for 24 simultaneous AMR calls

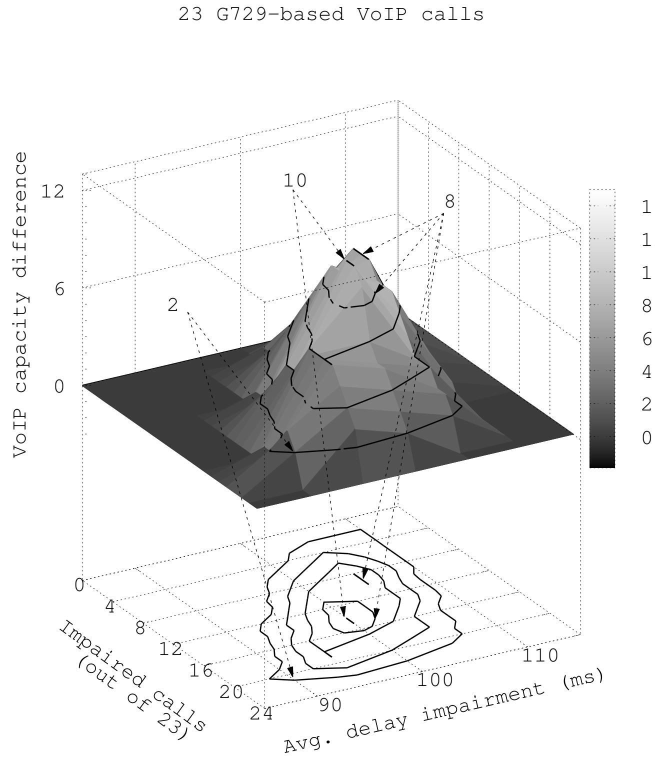

Figure 8: VoIP capacity improvement for 23 simultaneous G729 calls

supported). The surface contour lines are projected on the X-Y plane for a better visualisation of the levels of improvement in terms of VoIP capacity depending on the impairment delay and the number of VoIP sessions affected by it.

Figure 7 depicts the VoIP capacity improvement over the baseline solution when 24 AMR-based VoIP calls are running simultaneously in the network. We already established in Section III-A that for 24 VoIP calls using AMR codec a wireless node's queue occupancy is on average 37 packets accumulating an average queueing delay of 57 ms. Thus, VoIP packets that are already delayed due to travel in the network will be affected by an extra 57 ms delay before being transmitted, thus decreasing the QoS of the overall VoIP sessions. Due to the extra VoIP call added in the network the queuing delay doubled, compared with the previous 23 VoIP calls scenario, affecting negatively even more the VoIP session. Thus, the acceptable delay impairment of VoIP packets has to be smaller. However, for the scenarios considered, DAPP brings an improvement over a wider range of impairments the VoIP capacity in the network. In the best case scenario, the VoIP capacity is improved by 14 calls, from 10 supported calls using the baseline case to all 24 calls supported when DAPP is employed.

The 25 VoIP calls encoded with AMR scenario has not been considered in our analysis, as the queue occupancy is more than 100 packets, generating an average queueing delay of more than 150 ms, which is already unacceptable for VoIP calls, leading to all calls being negatively affected beyond any acceptable level. CAC is one solution to such a high degree of network impairments for VoIP calls.

2) VoIP capacity analysis for G729 calls: We also analysed the improvement in terms of VoIP capacity when VoIP calls encoded with G729 are deployed in the network. As presented in Figure 3 in Section III-A congestion starts to occur when 23, 24 or 25 calls are deployed in the network, leading to a queue occupancy of 10, 11 and 40 packets respectively. For these three transitional congestion scenarios the additional queueing delay affecting the packets is 20 ms, 21 ms, and 50 ms respectively. In these congested scenarios the queue occupancy is high enough for our solution to have an impact on VoIP capacity if more delayed packets are prioritised against less delayed packets. Scenarios with 26 or more G729 calls running simultaneously in the network are not considered, as the queuing delays, which add to the accumulated travel delay, are already too large (i.e. 150 ms) to bring any improvement through prioritisation.

Figure 8, 9 and 10 present the results for the transitional congestion scenarios when 23, 24 and 25 G729 VoIP calls,

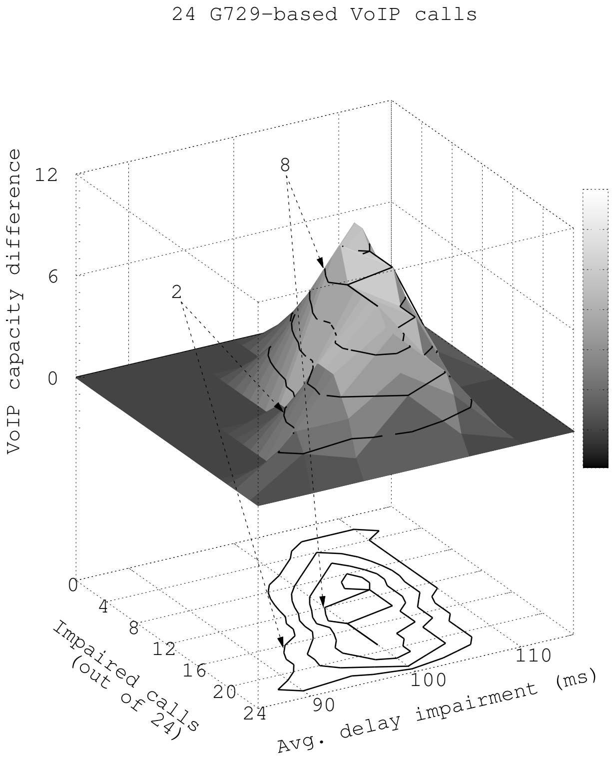

Figure 9: VoIP capacity improvement for 24 simultaneous G729 calls

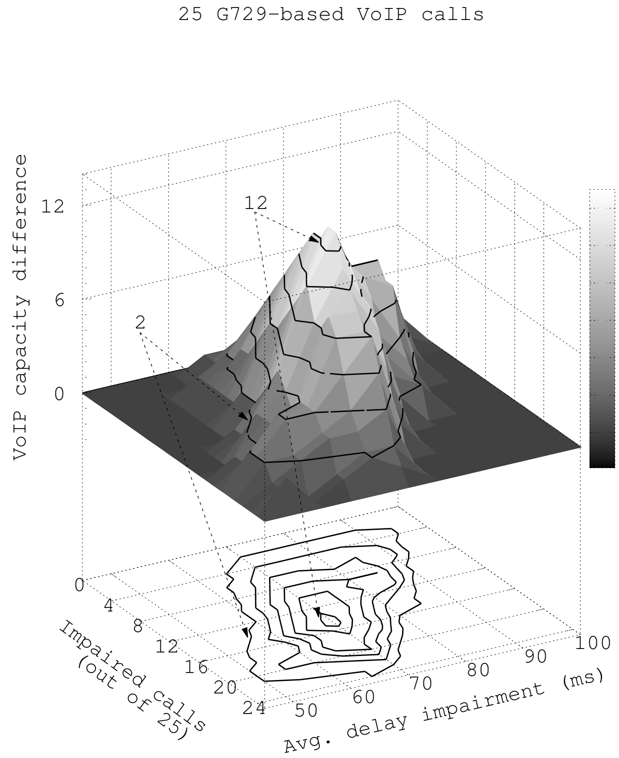

Figure 10: VoIP capacity improvement for 25 simultaneous G729 calls

respectively, are deployed in the network. Because the 23 and 24 VoIP calls scenarios have similar queue occupancy and queueing delays, the improvement brought by DAPP in terms of VoIP capacity is similar as it can be observed in Figure 8 and Figure 9. The improvement brought by DAPP ranges from one extra call supported up to 10 extra VoIP calls. In the best case scenario, the baseline case supports only 56% of all the calls (i.e. 13 VoIP cals), while DAPP can support 100% of the calls (i.e. all 23 VoIP calls).

For the scenario of 25 VoIP G729 calls deployed simultaneously in the network the queue occupancy is higher, around 35 VoIP packets on average, thus generating an average queue occupancy delay of 51 ms. Hence, DAPP brings an improvement in terms of VoIP capacity by increasing the quality of the VoIP sessions whose packets have an accumulated travel delay ranging between 50 ms and 95 ms, as it can be observed in Figure 10. Similarly, the improvement ranges from one extra call supported to 13 extra VoIP calls supported. In the best case scenario, when DAPP is not employed, only 48% of the calls are supported (i.e. 12 VoIP calls), whereas 100% of the calls are supported (i.e. 25 VoIP calls) when DAPP is employed.

We argue that in the non DAPP scenarios, VoIP users will hang-up their calls and retry due to bad network conditions, however that congested network scenario will not fade until a number of VoIP users will give up making their calls altogether. On the other hand, the side effects of DAPP is early detection of scenarios leading to total congestion, hence buying time for the network to adapt its forwarding settings to mitigate an imminent failure scenario. Examples of such settings are employing a CAC mechanism, codec-rate adaptation, or re-routing.

3) Delay Fairness Analysis: We showed in the previous section that our DAPP solution improves the VoIP capacity of the network by balancing the delays across VoIP sessions. This improvement comes from the effect of distributing the negative effects of the queueing delay on the least delayed VoIP calls, thus allowing the more delayed VoIP calls to preserve the quality they arrived with. Figure 11 demonstrates the delay fairness achieved by our mechanism, and depicts the Jain fairness index improvement of DAPP over the baseline case. The trend we observe is for DAPP to increase the overall network's fairness when the number of impaired calls is around the same with the non-impaired ones, in our case around 13 impaired calls versus 12 not impaired calls. The trend is also for DAPP-enabled networks to increase their delay fairness when the average delay impairment increases. In Figure 11 the improvement of the network's fairness is over 12% when 13 calls are impaired with more than 100

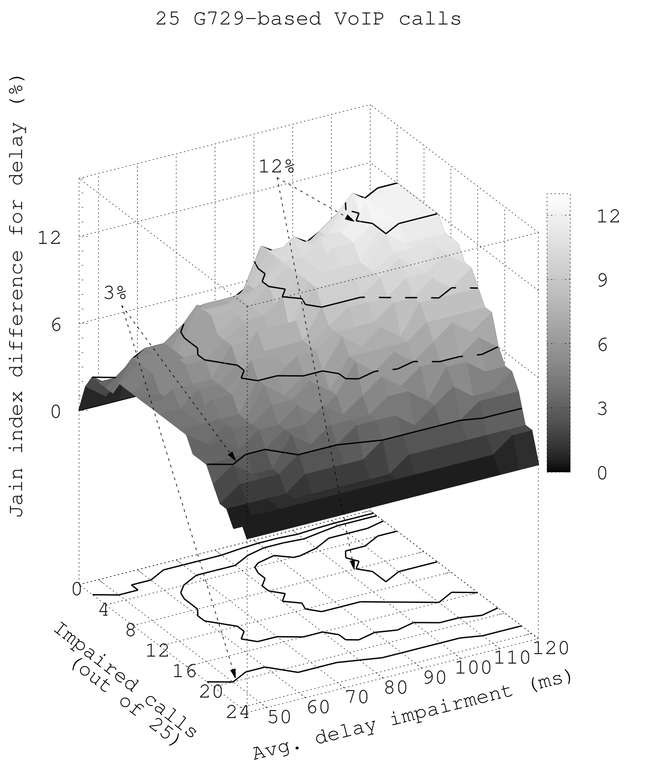

Figure 11: G729: Delay fairness improvement of our mechanism over the default case, in number of extra call accepted.

milliseconds. These improvements increase as the delay gap widens among the most and the least delayed VoIP calls, however for VoIP deployments this would mean impairing some VoIP calls at the point where users will drop their calls.

There are other applications that can be affected by a larger delay than that applicable to VoIP deployments. The fairness trend depicted in Figure 11 represents a great benefit for applications where decisions are taken based on information gathered from multiple sources, e.g. sensor nodes. When a process uses measurement timestamps, it has to wait until all information with the same timestamp has been collected. Considering a congestion scenario where measurement nodes are affected by network delay unevenly distributed, then the decision process can only take a decision when the most delayed piece of information arrives. When using DAPP the amount of delay affecting the most delayed packets is lower than when not using DAPP, thus improving the speed a decision can be made. As an example, we observed that in our results when using DAPP the packets belonging to the most delayed VoIP session arrive on average 15% earlier than in the case where DAPP is not used.

V. Conclusion

In this paper, we investigated the scenario when VoIP packets which were previously affected by various network delays, merge in a congested wireless node. Using DAPP, our proposed solution, we proved that forwarding older VoIP packets ahead of younger VoIP packets, regardless of the order they arrived, increases the VoIP capacity, and the QoS and delay fairness across all VoIP calls.

Through extensive NS-3 simulations, which consider various VoIP codecs and a wide range of realistic network delay impairments, we presented results proving that DAPP improves the number of supported VoIP calls in a wireless multi-hop network where network conditions degrade.

Acknowledgements

This work was supported, in part, by Science Foundation Ireland grant 10/CE/I1855, and supported, in part, by Science Foundation Ireland grant 13/RC/2094, and partially funded by the European Union through the European Regional Development Fund (ERDF) 2014-2020 programme via Enterprise Ireland (EI) grant IP2014 0344.

References

[1] S. Garg and M. Kappes, "Can I add a VoIP call?" in *Communications, IEEE International Conference on*, vol. 2, May 2003, pp. 779–783.
[2] S. Shin and H. Schulzrinne, "Measurement and Analysis of the VoIP Capacity in IEEE 802.11 WLAN," *Mobile Computing, IEEE Transactions on*, vol. 8, no. 9, pp. 1265 –1279, sept. 2009.
[3] J. Jun and M. Sichitiu, "The nominal capacity of wireless mesh networks," *Wireless Communications, IEEE*, vol. 10, no. 5, pp. 8–14, Oct 2003.
[4] C. Olariu, J. Fitzpatrick, Y. Ghamri-Doudane, and L. Murphy, "Provisioning call quality and capacity for femtocells over wireless mesh backhaul," in *Personal Indoor and Mobile Radio Communications (PIMRC), 2013 IEEE 24th International Symposium on*. IEEE, 2013, pp. 3071–3076.
[5] L. Zheng, L. Zhang, and D. Xu, "Characteristics of network delay and delay jitter and its effect on voice over IP (VoIP)," in *Communications (ICC), IEEE International Conference on*, vol. 1, Jun 2001, pp. 122–126.
[6] P. O. Flaithearta, H. Melvin, and M. Schukat, "Optimising QoS of VoIP over wireless LANs via synchronized time," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2013 IEEE 14th International Symposium and Workshops on a*. IEEE, 2013, pp. 1–3.
[7] "Medium Access Control (MAC) Quality of Service (QoS) Enhancements (Replaced by approved draft 802.11e/D13.0)," *IEEE Std P802.11e/D13.0*, 2005.
[8] F. Wang and M. Hamdi, "iPIFO: A Network Memory Architecture for QoS Routers," in *High Performance Switching and Routing, 2007. HPSR '07. Workshop on*, June 2007, pp. 1 –5.
[9] ITU-T G.107, "The E-Model, A Computational Model For Use In Transmission Planning," 2005.
[10] G. Wei-Xuan and Y. Shun-zheng, "Inference of One-Way Queuing Delay Distribution Using Packet-Pair Probes without Clock Synchronization," in *Network and Parallel Computing Workshops, 2007. NPC Workshops. IFIP International Conference on*, Sept 2007, pp. 169–175.
[11] ITU-T P.59, "Artificial Conversational Speech," 1993.
[12] R. G. Cole and J. H. Rosenbluth, "Voice over IP Performance Monitoring," *SIGCOMM*, vol. 31, no. 2, pp. 9–24, Apr. 2001.
[13] A. D. Clark, "Modeling the effects of burst packet loss and recency on subjective voice quality," in *IP Telephony Workshop*, 2001.
[14] C. Olariu, M. Foghlu, P. Perry, and L. Murphy, "VoIP Quality Monitoring in LTE Femtocells," in *Integrated Network Management (IM), 2011 IFIP/IEEE International Symposium on*. IEEE, 2011, pp. 501–508.
[15] R. de Fréin, C. Olariu, Y. Song, R. Brennan, P. McDonagh, A. Hava, C. Thorpe, J. Murphy, L. Murphy, and P. French, "Integration of QoS Metrics, Rules and Semantic Uplift for Advanced IPTV Monitoring," *Journal of Network and Systems Management*, vol. 23, no. 3, pp. 673–708, 2015.
[16] M. Al Masri, A. Sesay, and A. Fapojuwo, "Session state aware handover procedure for VoIP sessions in heterogeneous wireless networks," in *Wireless Communications and Networking Conference (WCNC), 2014 IEEE*, April 2014, pp. 3011–3016.