A Case for WiFi Relay: Improving VoIP Quality for WiFi Users

Amit Mondal
Northwestern University
Evanston, IL, USA
a-mondal@northwestern.edu

Cheng Huang, Jin Li
Microsoft Research
Redmond, WA, USA
{chengh, jinl}@microsoft.com

Manish Jain
Akamai
Cambridge, MA, USA
majain@akamai.com

Aleksandar Kuzmanovic
Northwestern University
Evanston, IL, USA
akuzma@cs.northwestern.edu

***Abstract*—Voice over Internet (VoIP) has been experiencing enormous growth in recent years. While posed to replace traditional PSTN for both enterprise and residential customers, VoIP has yet to achieve the same level of quality and reliability as PSTN. One key challenge is that a growing segment of customers is increasingly relying on WiFi connections. VoIP over WiFi (VoWiFi) experiences significant degradation in quality because of packet losses, mostly due to WiFi's low capacity, varying signal strength, interference, etc.**

To understand this problem, we have developed and deployed a comprehensive measurement platform in a global enterprise network. From large-scale real-world traces, we quantitatively analyze the impact of WiFi connections and study measures to mitigate such impact. Our results confirm that WiFi connections incur significantly more packet losses than wirelines, but these losses can be effectively concealed by sending each packet up to five times (*heavy replication*). Due to WiFi's inherent overhead, heavy replication only marginally increases WiFi airtime. To avoid the overhead on wirelines, we further propose a relay-based solution, where heavy replication only occurs between endpoints and nearby relays, and is removed before packets are transmitted on inter-branch long haul links or the public Internet. The solution has been implemented and deployed in the global enterprise network, and measurement results confirm that it can indeed greatly improve the performance of VoIP for WiFi users. In particular, it reduces the percentage of poor calls from 35% to 10%; and increases the percentage of acceptable ones from 45% to 70%.

I. INTRODUCTION

Individuals and companies are increasingly relying on VoIP (Voice over Internet Protocol) for personal and business communications. VoIP is not simply another new technology that makes traditional telephone calls less expensive, but rather a fundamentally disruptive one. It turns speech into digital data packets that can be stored, searched, manipulated, copied, combined with other data, and distributed to virtually any device that connects to the Internet. The flexibility and efficiency of VoIP quickly brings together the power of communications and business applications in a new and exciting manner. It is no surprise that the worldwide VoIP service revenue surged by 52% to 24.1 billion USD in 2007 [1].

Unlike traditional phone calls over public switched telephone networks (PSTN), VoIP calls do not traverse dedicated networks and are thus subject to uncontrolled delay, jitter, and losses. Even in an enterprise environment, where network infrastructures are normally well-provisioned and managed, VoIP calls can still experience significant quality degradation [2]. This is because an increasing number of users are connecting through WiFi links, which have unpredictable delay, jitter, and loss behavior. Indeed, the large-scale study in [2] shows that $5-10\%$ of the VoIP sessions with at least one WiFi endpoint experience more than 4% packet losses, producing poor audio quality and yielding a very poor user experience.

A recent survey [3] shows that 43% of small businesses provide only WiFi connections to their employees and 36% of organizations use VoIP over WiFi. Clearly, the number of enterprise users using VoIP over WiFi is large and will certainly increase over time. It is critical to design effective schemes to deal with packet losses occurring in VoIP over WiFi links.

To understand this problem, we have developed a comprehensive measurement platform – SureCall – to characterize packet losses and to evaluate their impact on VoIP performance. At a high level, SureCall emulates VoIP calls among volunteer machines and captures detailed packet-level traces. It is deployed within Microsoft's global enterprise network and has been running on 60 volunteer machines since September, 2008. After analyzing six months worth of packet traces — 4100 hours – we confirm that WiFi links can indeed experience high loss rate and cause significant quality degradation for VoIP. However, losses can be effectively concealed by *heavy replication* (sending each packet up to five times).

Due to the small size of VoIP payload and in light of the inherent overhead of WiFi links, our analysis indicates that heavy replication does *not* incur much overhead in terms of WiFi airtime. To avoid overhead on wired links, especially those inter-branch long haul ones, we propose a relay-based solution, where heavy replication occurs only between endpoints and nearby relays, and is removed before packets are transmitted on inter-branch long haul links or the public Internet. The WiFi relay solution is implemented and deployed on the SureCall platform. Experimental results show that our solution indeed improves the performance of VoIP for WiFi users – it reduces the number of poor calls with Mean Opinion Score (MOS) less than two from 35% to 10%, and increases the number of acceptable calls (MOS $\geq$ 3) from 45% to 70%.

The remainder of the paper is organized as follows. In Section II, we provide a brief background of the SureCall platform, and the impact of WiFi links on VoIP performance. In Section III, we study the effectiveness of the packet replication mechanism in recovering packet losses due to the wireless channel. We explore the overhead of packet replication over WiFi networks in Section IV. Section V presents the design of the WiFi relay solution. We evaluate and quantify the improvement of VoIP performance with WiFi relay in Section VI. We discuss related work in Section VII, and conclude in Section VIII.

II. Measurement Platform

In this section, we briefly describe the SureCall platform and discuss the impact of WiFi links on VoIP quality in the enterprise network.

A. SureCall Platform

We have developed a distributed platform – SureCall, which enables large-scale measurement studies and rapid prototyping of VoIP system designs. The SureCall platform comprises a light-weight master controller, which serves as a central coordinator, and clients, which run on volunteer machines. The master schedules clients to emulate audio calls by sending instructions to start audio sessions between them. The size and frequency of audio frames are specified by the master. Each client participates in one audio session at most once per hour. Each session lasts five minutes.[1]

We emulate G.722.1 audio codec for VoIP packet generation. This is an ITU-T standard for high quality, moderate bit rate (24 and 32 kbit/s) wideband ($50\,Hz$ - $7\,kHz$ audio bandwidth, $16\,ksps$) audio coding. In our current deployment, each SureCall client generates a 60-byte audio payload every $20\,ms$ thus operating at a bit rate of $24\,Kbps$.

B. Impact of WiFi Links

Using SureCall, we have gathered real-world VoIP packet traces with 60 endpoints, totaling 4100 hours worth of traces. We classify the traces into three categories based on the network interfaces used by the endpoints during VoIP sessions. WIRED-WIRELESS indicates scenarios where one endpoint uses a wired connection and the other endpoint uses a WiFi connection. We slice each 5-minute long trace into 5-second segments, and compute the loss rate for each 5-second segment. Figure 1 plots the CDF of loss rates in each category. The figure shows that there is almost no loss in the WIRED-WIRED scenario whereas there are significant packet losses in the WIRED-WIRELESS and WIRELESS-WIRELESS scenario. This confirms that WiFi links can indeed experience high loss rate, causing significant quality degradation for VoIP. For VoIP over WiFi, packet losses are more likely due to bad connectivity, weak WAN signal, high interference, etc. [4]. Adjusting communication rate, which is often applied to respond to congestion-related losses, is not a cure here. Thus, we explore the effectiveness of application layer replication in recovering loss in WiFi connections.

III. How effective can heavy replication be?

In this section, we analyze the SureCall traces to answer the following question: What percentage of the packet losses can be recovered by additional replication?

For VoIP traffic, the bit-rate is low and individual packets are small. Typically, an audio codec generates one packet every $20\,ms$, i.e., a frame rate of $50\,fps$. Assuming a typical

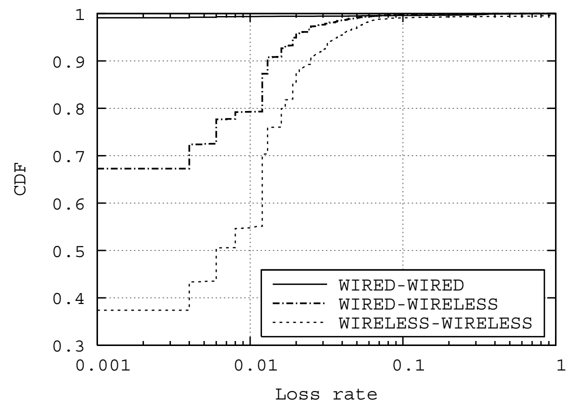

Fig. 1. Impact of WiFi on loss

fixed de-jitter buffer size of $100\,ms$, there can be at most five packets in the de-jitter buffer. Therefore, a common practice of Forward Error Correction (FEC) for VoIP is not to encode across packets, but rather to simply replicate each data packet. In such a scheme, each audio packet is sent out r times (referred as *replication ratio*): when it is first generated, as well as together with $r-1$ future data packets, each d-packet away (d is referred as *replication distance*). For instance, say $r=3$ and $d=2$, then packet i is sent out 3 times: when it is generated, together with packet $i+2$, as well as with packet $i+4$.

It is worth noting that our analysis in this section is *passive* and based on a simplified assumption. For low bitrate VoIP traffic, we assume that increasing its packet size (say from 60 bytes to 240 bytes per packet) will *not* alter the congestion behavior of the underlying communication channel. Hence, we simply take each SureCall trace and evaluate various replication ratios assuming the same send/receive timestamps and packet losses. In Section VI, we present experimental results from real implementations on the SureCall platform, where the size of each packet is set faithfully based on a given replication ratio.

To keep the comparison simple and illustrative, we use a fixed replication distance $d = 1$, and compare various replication ratios, from $r = 2$ to $r = 5$. We assume a fixed de-jitter buffer of 100ms, that is, a packet is treated as lost *either* when all its replica are lost *or* even the first received replica has jitter more than 100ms.

Figure 2 compares the replication performance of various replication ratios. To interpret the figure, we select a data point on the "replication ratio = 2" curve, where the x-axis value is 20% and y-axis value is also 20%, which means that for 20% of the audio sessions, sending each data packet twice will leave no more than 20% of the losses unrecovered (in other words, 80% or more losses will be recovered). This recovery ratio is high, but, unfortunately, only for a small percentage of sessions (20% here). Next, we examine the performance of sending each data packet 5 times. For the same x-axis value (20%), the y-axis value is now 87%, that is, for 87% of the sessions, sending 5 times will leave no more than 20% unrecovered. This is a huge improvement over sending twice

[1]The SureCall clients emulate calls even if the host machines are *not* idle. Fortunately, the enterprise machines tend not to run bandwidth intensive applications, such as streaming video or peer-to-peer file sharing.

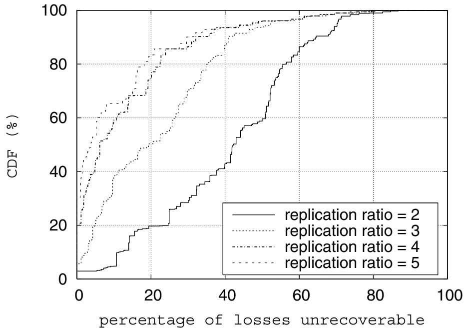

Fig. 2. Effectiveness of packet replication.

and even bigger compared to no replication case (no loss can be recovered). This concludes that 1) packet losses can be mitigated through replication, and 2) heavy replication is very effective in hiding losses (the remaining losses can then be concealed using signal processing techniques [5]). Note that the performance of $r = 4$ and $r = 5$ is close.

IV. Overhead of Heavy Replication

In this section, we analyze the overhead of the packet replication scheme for VoIP over WiFi networks. Given the small size of voice payload, the overhead of various protocol headers in a VoIP packet becomes nonnegligible. For example, a G.722.1 voice codec generates 60 bytes of audio packets every $20ms$. Such voice payload is typically encapsulated with RTP (12 bytes), UDP (8 bytes), IP (20 bytes), 802.11 MAC (28 bytes), and PHY (20μs of airtime for 802.11g) headers. In addition 4 bytes of CRC and 4 bytes FCS are appended. Thus, a G.722.1 packet may take around 70μs[2] to transmit at the maximum rate of 54Mbps in 802.11g. If acknowledgment mode of communication is used, then each data transfer is followed by a synchronous 802.11 ACK frame (14 bytes) that increases the air time to 102μs[3]. Table IV shows the air time with replication ratio = 1,2,3,4. This indicates that replicating the voice payload at the application layer causes only a marginal increase in air time. Even with 4 times replication, the air time increases from 102μs to 128μs, which is merely 26%.

On the other hand, if all the replicated payloads were to be transmitted over wired links, where the MAC and PHY layer overhead is much less, it would increase the transmission time by a significant percentage. This is wasteful and especially undesirable on inter-branch long haul links, which are often costly and thus provisioned with limited capacity. Thus, we propose a relay-based solution, which allows heavy replication over WiFi links without incurring any overhead on wired links.

V. The WiFi Relay Solution

In the previous sections, we have shown that heavy replication can be a very effective way to recover packet losses due to wireless medium, and it incurs only marginal overhead in WiFi airtime. Thus, we advocate using heavy replication to improve VoIP quality whenever a WIRELESS endpoint is involved in a VoIP call. In addition, to remove wasteful overhead on wired links, especially to ease load on inter-branch long haul ones, we propose the following relay-based solution: the wireless endpoint sends packets with *heavy replication* to a nearby wired node, which is located in a nearby location (the same office building, campus, or city). The wired node removes the replication before forwarding towards the destination. Conceivably, when both endpoints are connected using wireless, the solution involves two-hop relays, where the second relay restores heavy replication before forwarding packets to the wireless destination.

Replication ratio	Air time (μs) w/o ACK	w/ ACK
1	70	102
2	79	111
3	87	120
4	96	128

TABLE I
Overhead of packet replication on WiFi air time

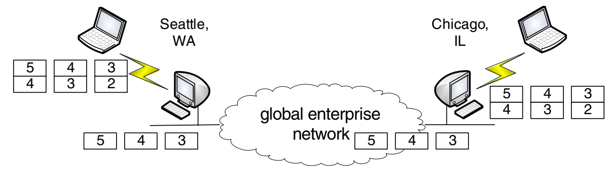

Fig. 3. Design of Enterprise P2P VoIP

Figure 3 illustrates two wireless clients, one located in the Chicago office and the other one in the Seattle office, participating in a VoIP call. The replicated packets from the Chicago wireless client is relayed through a wired node located in the Chicago office where the replications are removed before the packets are forwarded to the second relay node, located in the Seattle area. This second node then restores the replications before forwarding the packets to the wireless client in the local Seattle area, and thus relieving the Chicago-Seattle long haul link from carrying heavily replicated packets.

The requirement of a dedicated relay infrastructure can become prohibitive for the deployment of such a solution. An alternative approach is to use WIRED VoIP endpoints as relay nodes. A WIRELESS VoIP endpoint can redirect packets to a nearby WIRED VoIP endpoint, which removes the replication, and forward the packets to destination. This leads to the design of the proposed WiFi relay solution.[4]

VI. Evaluation

We have implemented the relay functionality on the SureCall platform, which is deployed through an update to all the SureCall clients. In addition to adding replication to protect VoIP calls, each client is also capable of relaying packets. The relay functionality involves receiving replicated packets from a WIRELESS endpoint, removing the replications, and

[2] DIFS + PHY header + (60+76)bytes/54Mbps.

[3] DIFS + PHY header + (60+76)bytes/54Mbps + SIFS + PHY header + 14bytes/54Mbps.

[4] Note that the relay functionality can also be implemented in a wireless access point (AP), as long as it is able to perform application specific processing.

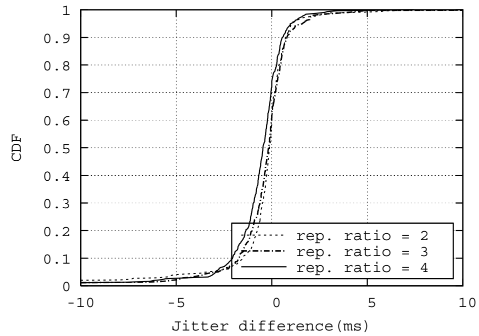

Fig. 4. CDF of jitter difference at 50^{th} percentile.

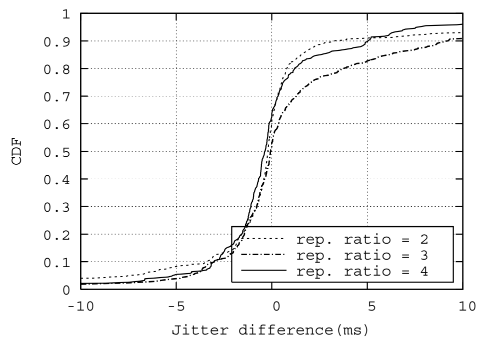

Fig. 5. CDF of jitter difference at 95^{th} percentile.

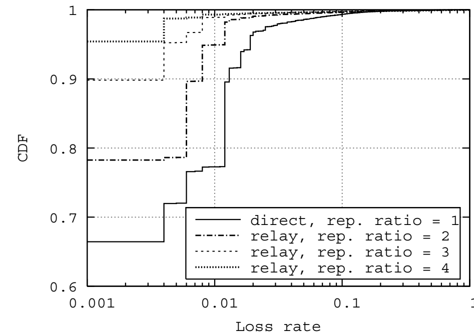

Fig. 6. CDF of loss distribution

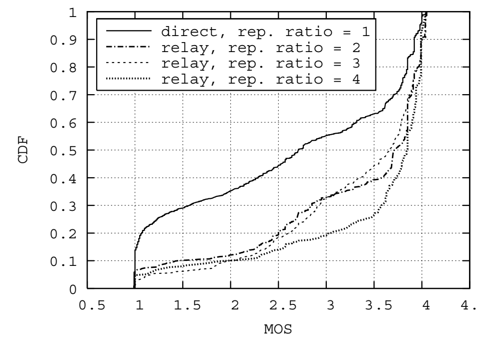

Fig. 7. CDF of MOS difference.

forwarding the packets to a destination or a second relay node (if the destination is also a WIRELESS endpoint). In the latter case, the second relay node buffers recent packets, restores the data replication and forwards protected packets to the WIRELESS destination.

To evaluate the system, SureCall schedules direct VoIP sessions between either two WIRELESS clients or one WIRELESS and another WIRED client. In addition, it finds idle (i.e., not participating in any VoIP session) WIRED endpoints located in the same building (or city/campus) and schedules simultaneous relay VoIP sessions (one-hop-overlay if only one endpoint is WIRELESS, and two-hop-overlay if both endpoints are WIRELESS). SureCall uses an enterprise internal location database to find geographically nearby endpoints.

A. Does Relay itself Introduce Jitter?

In the WiFi relay solution, relay nodes are not dedicated, but rather ordinary endpoints. One would wonder whether relay nodes themselves can introduce jitter and thus degrade VoIP performance. To this end, we compare the jitter values between each direct session and its corresponding simultaneous relay session (between the same pair of endpoints). If relay *does* incur significant jitter, then the distributions of the two jitter values should be far apart. Otherwise, they should be very close.[5] To compare two distributions, we calculate the difference between the 50^{th} (or 95^{th}) percentile values. Finally, we plot a distribution of all the difference values. Figure 4 shows three CDF curves – one for each replication ratio. It shows that for more than 99% of the cases, the jitter difference between a direct session and its corresponding relay session is less than 5ms at 50^{th} percentile. Figure 5 shows the corresponding CDF plot for 95^{th} percentile. For more than 90% of the time, the jitter difference at 95^{th} percentile is less than 10ms. This shows that using relay itself has a negligible impact on end-to-end jitter.

[5]For relay sessions, we exclude packets recovered from replications, whose large jitter values do *not* have counterparts in direct sessions.

B. Advantage of the WiFi Relay Solution

Figure 6 plots the loss rate distributions of the direct sessions, where the replication ratio is one, and the relay sessions, where the replication ratio equals to two, three or four. Clearly, the relay sessions show much improved loss characteristics compared to the direct ones.

Next, we compare the call quality, which is often represented as a Mean Opinion Score (MOS) ranging from 1 (bad) to 4.5 (good). We compute the MOS score for each VoIP session using network level metrics with the approximation in [6]. We assume a fixed de-jitter buffer size of $100\,ms$, so packets experiencing jitter of more than $100\,ms$ are interpreted as lost. We slice each 5-minute-long session into 10-second segments, and compute the MOS score for each segment. We take the minimum MOS value as the MOS score for the session, as the worst 10 second can dominate user perception about the entire session.

Figure 7 shows the CDF of MOS values for direct and relay sessions. It shows that the relay solution reduces the percentage of poor calls (MOS $\leq$ 2) from 35% to 10%, and increases the number of acceptable calls (MOS $\geq$ 3) from 45% to 70%. Therefore, we conclude that it is viable to design and deploy a WiFi relay solution, without extra investment in infrastructures, and yet achieve significant improvements on VoIP quality for WiFi users.

VII. Related work

Forward Error Correction (FEC) schemes [7] have been proposed as a method of reducing the effective packet loss rate of lossy links. These schemes add redundant information and send it along with the original data, such that in case of a loss, the original information (or part of it) can be recreated. Huang et al. [8] propose the use of adaptive FEC for VoIP quality improvement. Dube and Altman [9] study the effectiveness of using a replicated packet for FEC. Sengupta et al. [10] propose using other link layer techniques such as automatic repeat request (ARQ), packet aggregation and variable size MAC protocol data unit combined with FEC to improve quality of a VoIP call over WiMaX. Kawata et al. [11] study the performance of VoIP over 802.11 networks. Patrick et al. [12] design *softspeak* which enables VoIP traffic to effectively share 802.11 channel capacity with background TCP/UDP flows to improve VoIP quality. In this paper, we use packet replication between an endpoint and a relay node to improve media performance when the endpoint is connected via a wireless link. Yet, the relay node removes any redundancy to avoid increases in inter-branch backbone traffic due to packet replication.

Overlay networks originally emerged to improve reachability by providing robust routing around Internet path failures [13]. OverQoS [14] proposes an overlay link protocol that uses both retransmissions and FEC to provide loss and throughput guarantees to improve QoS in the Internet. Amir et al. [15] use overlays to enhance VoIP quality in the Internet. It segments end-to-end paths into shorter overlay hops and attempts to recover lost packets using limited hop-by-hop retransmission. Others show that, when possible, using multiple paths or dynamically-switching paths based on real-time metrics improves VoIP performance [15]–[18]. The key difference between our approach and the above ones lies in the fact that we propose using an overlay to improve media performance when the endpoint is connected via a wireless channel in an enterprise environment. Indeed, our measurement results show that the access wireless part of the end-to-end path generates most severe quality degradations.

The rise of Skype has led to studies that try to understand peer-to-peer VoIP networks. Baset et al. [19] dissect the Skype protocol, in term of its login, NAT/firewall traversal algorithm, codec, call establishment, and media transfer. They report that Skype's overlay is mainly used to by-pass NATs, and not to deal with performance problems. Guha et al. [20] study the Skype node dynamics and churn, and the network workload generated by the Skype users. Chen et al. study the QoS of Skype VoIP system [21], and propose the use of a user satisfaction index that is derived from bit rate, bit rate jitter and one-way delay. Others compare the performance of Skype with MSN VoIP clients [22]. They report that Skype performs significantly better due to its codec and rate control mechanisms. Contrary to common wisdom, we have shown that the VoIP quality in enterprise networks can be quite low. Hence, we propose a scheme to address the problem in this particular environment.

VIII. Summary

In this paper, we have shown that wireless connections can cause significant degradation of VoIP quality due to random packet losses, and heavy replication can be a very effective way to hide these losses. We advocate using heavy replication to improve VoIP quality for WiFi users, and propose a relay-based solution to remove the replication before data is transmitted on inter-branch long haul links. We have implemented and deployed the WiFi relay solution in a global enterprise network, and our experimental results demonstrate that the solution can indeed significantly improve the performance of VoIP for WiFi users.

References

[1] M. Machowinski, "VoIP services and subscribers: Annual market share and forecasts," Infonetics Research, Inc., Aug. 4, 2008.

[2] V. Vasudevan, S. Sengupta, and J. Li, "A first look at media conferencing traffic in the global enterprise," in *Passive and Active Measurement Conference*, 2009.

[3] C. MacGillivray, "Mobile Internet access in the SMB: Demand-side analaysis of WiFi and 3G mobile broadband," IDC, Dec. 2008.

[4] S. Rayanchu, A. Mishra, D. Agrawal, S. Saha, and S. Banerjee, "Diagnosing wireless packet losses in 802.11: Separating collision from weak signal," in *IEEE INFOCOM*, 2008.

[5] C. Perkins, O. Hodson, and V. Hardman, "A survey of packet loss recovery techniques for streaming audio," *IEEE Network*, vol. 12, no. 5, 1998.

[6] R. Cole and J. Rosenbluth, "Voice over IP peformance monitoring," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 31, no. 2, Apr. 2001.

[7] J. Bolot, S. Parisis, and D. Towsley, "Adaptive FEC-based error control for Internet telephony," in *IEEE INFOCOM*, 1999.

[8] C. Huang, S. Sukittanon, J. Ritcey, A. Chindapo, and J. Hwang, "An embedded packet train and adaptive FEC scheme for VoIP over wired/wireless IP networks," in *IEEE ICASSP*, 2006.

[9] P. Dube and E. Altman, "Utility analysis of simple FEC schemes for VoIP," in *Networking*, 2002.

[10] S. Sengupta, M. Chatterjee, S. Ganguly, and R. Izmailov, "Improving R-Score of VoIP streams over WiMax," in *IEEE ICC*, 2006.

[11] T. Kawata, S. Shin, A. G. Forte, and H. Schulzrinne, "Improving the capacity for VoIP traffic in IEEE 802.11 networks with dynamic PCF," in *IEEE WCNC*, 2005.

[12] Y. A. P. Verkaik, R. Gupta, and A. Snoeren, "Softspeak: Making VoIP play well in existing 802.11 deployments," in *USENIX NSDI*, 2009.

[13] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris, "Resilient overlay networks," in *USENIX OSDI*, 2001.

[14] L. Subramanian, I. Stoica, H. Balakrishnan, and R. Katz, "OverQoS: An overlay based architecture for enhancing Internet QoS," in *USENIX NSDI*, 2004.

[15] Y. Amir, C. Danilov, S. Goose, D. Hedqvist, and A. Terzis, "An overlay architecture for high-quality VoIP streams," *IEEE Trans. on Multimedia*, vol. 8, no. 6, Dec. 2006.

[16] R. K. Rajendran, S. Ganguly, R. Izmailov, and D. Rubenstein, "Performance optimization of VoIP using an overlay network," in *IEEE INFOCOM*, 2006.

[17] Y. Liang, E. Steinbach, and B. Girod, "Multi-stream voice over IP using packet path diversity," in *IEEE Multimedia Signal Processing*, 2001.

[18] S. Tao, K. Xu, A. Estepa, T. Gao, R. Guerin, J. Kurose, D. Towsley, and Z. L. Zhang, "Improving VoIP quality through path wwitching," in *IEEE INFOCOM*, 2005.

[19] S. Baset and H. Schulzrinne, "Analysis of the Skype peer-to-peer Internet telephony protocol," in *IEEE INFOCOM*, 2006.

[20] S. Guha, N. Daswani, and R. Jain, "An experimental study of the Skype peer-to-peer VoIP system," in *IPTPS 2006*, 2006.

[21] K. Chen, C. Huang, P. Huang, and C. Lei, "Quantifying Skype user satisfaction," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 36, no. 4, Oct. 2006.

[22] W. C. et al., "A performance study of VoIP applications: MSN vs. Skype," in *IEEE MULTICOMM*, 2006.