Raabta: Low-cost Video Conferencing for the Developing World

Zubair Nabi
Information Technology University
Pakistan
zubair.nabi@itu.edu.pk

ABSTRACT

We present *Raabta*, a video conferencing system for the developing world, which is designed specifically to cater to the technological, social, and economic condition of its target audience. Specifically, to minimize the cost and the energy footprint of the system, it uses the existing analog cable TV network as a communication substrate to connect Raspberry Pi end-hosts. The network stack is designed to operate over a largely unmodified broadcast-based coaxial network. The architecture is completely decentralized and can scale to hundreds of concurrent connections without requiring any backbone connectivity. To make smart use of the existing bandwidth, the video conferencing application uses multi-layered encoding and dynamically switches between selective retransmission and forward error correction based on the loss rate. Finally, the text-free user interface is simple enough to be used by low-literate users. To enable wider Internet connectivity, the architecture can easily be augmented using existing backbone technologies. While designed for direct communication between individuals, we also discuss how the same system can be used to enable community-based telemedicine and distance learning, among other applications.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*distributed networks, network communications*; H.4.3 [**Information Systems Applications**]: Communications Applications—*computer conferencing, teleconferencing, and videoconferencing*

Keywords

Developing World, Lowest Cost Denominator Networking, Video Conferencing, ICT4D

LCDNet'13, September 30, 2013, Miami, Florida, USA.

ACM 978-1-4503-2365-9/13/09 ...$15.00.
http://dx.doi.org/10.1145/2502880.2502886.

1. INTRODUCTION

Over the years, the Internet has established itself as a basic human right [24]. This is primarily due to its ability to act as a conduit for information and knowledge, and to bring people closer together. The latter property is attested by the near universal use of text messaging, telephony, and video conferencing in the developed world. The Internet has the potential to radically improve the human condition [5] and create new industry models [18]. So much so that according to the World Bank an even 10% increase in Broadband Internet penetration can lead to a 1.4% increase in the national GDP [4]. Unfortunately, a large fraction of humanity—predominantly in the developing world—is devoid of this facility. This problem is even more acute in rural areas, which account for between 70% to 85% of the developing world population [18]. To address this, researchers and social entrepreneurs have explored the use of WiMAX [17], satellite [18], long-distance WiFi [20], ZigBee [21], optical, and cellular [14] technologies to enable and/or improve backbone connectivity in these regions, along with a similarly rich set of last mile access technologies.

Unfortunately, even these remedial solutions are plagued by shortcomings. Satellite links are cost prohibitive, optical fibers cannot reach the rugged terrain of rural areas, and microwave links are expensive both in terms of monetary cost and engineering [9]. On the other hand, cellular, long-distance WiFi, and WiMAX based solutions require physical towers which substantially increases the cost of even a small deployment [21]. Above all, wide-spread deployment of all of these networks is limited by their requirement of new infrastructure deployment. In fact, most of the deployment cost (around 70%) is attributed to access networks as they require concentrated distribution to reach end users[5]. Moreover, the sustainability of such deployments is hampered by cultural, environmental, and technical challenges specific to the developing world [6]. Specifically, power is both intermittent and of bad quality which results in frequent equipment failure. The situation is aggravated by extreme weather conditions including humidity, heat, dust, and dirt. To deal with this, components need to be constantly replaced and expensive backup power sources need to be maintained. Additionally, unconventional monitoring needs to be both developed and rolled out [27]. Finally, the lack of qualified local staff further complicates the matter from an operational perspective [6]. This entire ecosystem requires a rolling investment by either a government or a third party, which is not always forthcoming.

While Internet access is extremely useful, it is not always an underpinning technology. In fact, the requirements of some of these regions are very basic and necessitate the prioritization of simple access network communication over end-to-end Internet connectivity [17]. This coupled with the extremely high cost of Internet bandwidth [23], makes backbone connectivity a limiting factor in most deployments. This is also indicated by the considerably higher penetration of mobile phones in the developing world than the Internet [16]. This also suggests that people in these regions prefer simple communication mechanisms to keep in touch with others. In addition, systems designed for the developed world are a bad fit for the developing world due to a number of fundamental differences including cost, power, and usage [5]. Moreover, the situation is further exacerbated by the fact that networks designed for urban environments are ill-suited for rural areas due to the low population density [4]. Finally, general purpose computers designed for the industrialized world are out of the reach of poor users in the developing world due to their high price tag [1].

To remedy the situation, we present Raabta, a video conferencing system designed for the developing world. It uses the existing analog cable TV network—which has reasonable penetration in rural areas—as a last mile access network. The only change to the existing infrastructure is the replacement of unidirectional amplifiers with bidirectional ones. Instead of personal computers, Raspberry Pis are employed as end-user devices and are connected through a broadcast-based coaxial analog network, enabling decentralized communication. The video conferencing application and the network stack have been co-designed to make efficient use of resources and usage patterns. Finally, the text-free user-interface lowers the usage barrier for low-literate users. Overall, the entire system is low cost, power and bandwidth efficient, and flexible enough to be extended to support wider Internet connectivity. According to our estimates, around 3.9 million individuals—who live in rural areas and own a TV set with an analog cable TV connection—just in Pakistan can directly and instantly benefit from it.

The rest of the paper is organized as follows. We enumerate our design goals in §2. A brief introduction to analog cable TV networks, Raspberry Pi, and the architecture of video conferencing systems is given in §3. §4 presents the design of our proof-of-concept system and its various components. We discuss how Raabta can easily be connected with any existing backbone to enable Internet connectivity in §5. In the same section we also discuss some other community-wide applications of our system including telemedicine and distance learning. Related work is summarized in §6. We finally conclude in §7 and also discuss future directions.

2. DESIGN GOALS

In this section, we discuss our design goals and how they have inspired the design of Raabta.

Cost: Roughly half of the population of the earth lives on less than $2 a day [10]. Even basic electricity is a luxury for this cross-section. Therefore, low-cost is a prerequisite for widespread deployment and sustainability of any developing world initiative [5, 27].

Power: Most of the developing world is either without power [10] or experiences intermittent and bad quality electricity [5, 27]. Therefore, low-power consumption is also a first-class goal.

Fault-tolerance: Component failure rate is considerably higher in the developing world due to extreme weather and environment conditions [6, 27]. As a consequence, systems need to be designed which treat component failure as the norm rather than the exception (similar to cluster computing these days which relies on failure-prone, commodity off-the-shelf hardware). Decentralized peer-to-peer architectures are more suited to this environment than the client-server model.

User Interface: The literacy rate in the developing world is less than 50% [6]. As a result, even simple tasks such as mouse clicks for selection are counterintuitive for such individuals. This is primarily due to the heavy reliance on text in most user interfaces. A natural solution to this problem is the use of text-free interfaces, voice feedback, and visual on-screen help [19].

Spectrum: Most existing solutions rely on frequencies that fall within licensed bands, thus incurring an additional cost. On the other hand, as discussed earlier, technologies that make use of the unlicensed spectrum have their own set of shortcomings. While some spectrum in UHF and VHF is being opened up as "white space", the infrastructure required to operate over these bands is cost-prohibitive for smaller community-centric setups. In addition, bandwidth is an extremely constrained resource in the developing world [23]. Therefore, for most community-local applications, Internet connectivity should not be a requirement.

Infrastructure: Wired distributions require expensive cabling and signal amplifiers while wireless networks need physical towers and repeaters. Ideally, a new system should make use of existing infrastructure through resource pooling. This also enables power costs to be amortized across many services.

Operational Sustainability: In addition to finances, long-term sustainability of such projects depends on the ability of local staff to take over maintenance [6, 27]. While it is possible to fix software problems remotely, hardware problems still require a visit to the field. Consequently, the life of a setup can be greatly enhanced by architecting a design that does not need any maintenance (such as decentralized systems) or requires only basic skills that are already possessed by the local workforce.

Quality of Experience: Studies have shown that the user Quality of Experience in video conferencing systems degrades if the video delay exceeds 350ms [31]. To enable an acceptable QoE, all the components within the system need to be co-designed.

Real Access/Real Impact (RA/RI) The RA/RI criteria [28] for ICT4D projects defines a checklist for long-term sustainability. It revolves around, among many factors, local needs, affordability, socio-cultural factors, and micro- and macro-economics to evaluate the design of an initiative.

2.1 Summary

In view of the above, the design goals of Raabta include a low-cost, power-efficient, video conferencing system that is completely decentralized and can scale to hundreds of concurrent users on the same medium while ensuring good Quality of Experience and conforms to the RA/RI criteria. In addition, it should make use of existing infrastructure, for which local maintenance already exists. Moreover, it should require no backbone connectivity and should not rely on any licensed spectrum.

3. BACKGROUND

In this section, we first describe typical analog cable TV networks, followed by an introduction to Raspberry Pi, and then finally a description of video conferencing software.

Analog Cable TV Networks.

Cable networks have a three-tier hierarchy with the cable operator at the top, followed by a number of *headends*, which are connected directly to subscribers [2]. Connectivity is provided by a Hybrid Fiber-Coaxial cable (HFC) network. The cable operator is connected to the headends through a digital optical fiber network while the headends in turn, are connected to subscribers through an analog coaxial medium. Each headend serves around 1000 subscribers. In the developing world, the hierarchy is reduced to two tiers with the cable operator directly connected to subscribers through an all-coaxial network. Regardless of the hierarchy, the cable operator uses digital satellite to subscribe to a number of channels which are then distributed across the cable network. The coaxial network is broadcast-based and all content on both the uplink and downlink is seen by everyone. Downstream traffic is between 4.9Gbps and 6.6Gbps, out of which around 3.3Gbps is used for cable television and the rest for other services. The same network can be enhanced to provide telephony and Internet services, but at present this service is mostly confined to urban areas. The uplink for these additional services is normally around 215Mbps, which also includes set-top box control traffic. In case of Internet connectivity, end-users are provided with analog cable modems, which act as MAC layer bridges [8]. At the other end, the Cable Modem Termination System (CMTS) is connected to an Internet backbone. The network stack at both ends conforms to Data-Over-Cable Service Interface Specifications (DOCSIS) [7].

It is important to quantify the number of individuals in rural areas who have cable TV access. According to the World Bank, the rural population of Pakistan is 112 million [30], out of which 11% or 12 millon people own a television set [12]. Out of these, 31% have a cable TV connection [11]. Therefore, approximately 3.9 million individuals in rural areas of just Pakistan own a TV set with a cable connection.

Raspberry Pi.

The Raspberry Pi is a general-purpose single-board computer, originally designed to teach Computer Science to school children by the Raspberry Pi Foundation in collaboration with the University of Cambridge and Broadcom [13]. It is powered by a Broadcom BCM2835 system-on-chip (SoC) multimedia processor, which consists of an ARM1176JZF-S 700MHz processor, a VideoCore IV GPU (24 GFLOPS), and integrated audio and video. Storage is provided through an external memory card such as SD or MMC. A number of video output options are present including Composite RCA, HDMI, and DSI (Display Serial Interface). Audio can either be obtained through HDMI or the universal 3.5mm jack. Furthermore, the Raspberry Pi can be interfaced with peripheral devices through 8 General Purpose Input/Output (GPIO) in addition to other standards including UART. It comes in two flavours: (1) Model A with 256MB of RAM and 1 USB port, and (2) Model B with 512MB of RAM, 2 USB ports, and 10/100 Ethernet. The former costs $25 while the latter $35. The software stack consists of ports of standard Linux distributions, including Debian and Fedora.

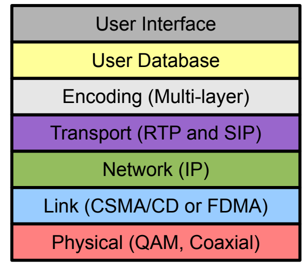

Figure 1: Raabta Design

Video Conferencing.

Over the course of the last one decade, a number of free video conferencing applications have found traction, with Skype and Google+ being popular choices. At a high-level, each application has 3 functional aspects: its overlay topology, encoding scheme, and packet loss recovery [31]. These systems require high-bandwidth and low-latency transmission. To enable good Quality of Experience, they are required to perform their operation in real-time. To this end, they are designed to tolerate device and network heterogeneity and wide-varying network conditions.

4. RAABTA

This section presents the design of Raabta in detail. We first present the physical end-hosts (§4.1) and then sketch the network design (§4.2). The network stack is described next (§4.3), followed by the video application and its interface (§4.4). Figure 1 shows the entire design at a high-level.

Component	**Cost ($)**
Raspberry Pi Model B	35
Camera Board	25
External Cable Modem	20
USB Sound Card	7
USB Keyboard	2
USB Mouse	2
Headset	1
Total	**92**

Table 1: Raabta End-host Cost Breakdown

4.1 End-host

For the end-hosts, we use Raspberry Pi (RPi)—as a realization of TinyPC [1]—because of its low-cost and low-power draw. For the video signal a 5MP camera, specifically designed for the RPi [29], is connected to the setup. The RPi does not come with a mic so one has to be interfaced externally. One option is to use a USB mic but that costs in excess of $100 so instead, we use a USB soundcard and connect a low-cost headset to it. Input is enabled through a USB mouse and keyboard and the output is through the RCA interface to the TV. Finally, an external cable modem is employed to interface the RPi with the analog coaxial network. Table 1 dissects the cost of each component. As opposed to desktops and laptops, which cost several hundred dollars, this entire setup costs $92, which we believe can be further reduced through the use of custom Silicon [10]. Although our original goal was to make the end-host accessible to every household within the rural community, in its current form the village kiosk model might be more applicable [5].

4.2 Network Design

Due to the use of the underlying analog coaxial network, the medium is shared and has a tree-and-branch architecture with a maximum radius of 100KM (RTT 1.6ms) [7]. The cable spectrum planning differs between North America and Europe. In this paper we focus on the North American standard but the principles are equally applicable to the European standard. The downstream spectrum spans 50-860MHz and each channel has a width of 6MHz, for a maximum of 135 channels using FDMA. Using 64-QAM or 256-QAM, each channel has a data rate of 32 or 42Mbps, respectively. If the network is used for Internet connectivity, the DOCSIS specification earmarks 5-42MHz as the upstream spectrum with an individual channel width between 200KHz-3MHz. Using QPSK or 16-QAM, a data rate of 320kbps-10Mbps is achievable which is shared across all users through TDMA. Each end-host is provisioned with a cable modem with an associated MAC address for link-layer access.

Internally, the modem consists of a tuner, modulator, demodulator, MAC, and Ethernet interface. The coaxial cable passes through a *splitter*, which splits it into two signals, one for normal TV channels and the other for the modem. The latter is connected to the tuner which consists of a *diplexer* which allows it to make use of any frequency within the spectrum. The tuner downgrades the signal to a lower frequency and passes it to the demodulator which does analog to digital conversion. The digital signal is then passed to the MAC which runs the link-layer protocol. It is important to highlight that this operation can also be delegated to the end-host. Finally, the MAC passes the actual data to the Ethernet interface which is connected to the end-host (or Raspberry Pi in our case). The entire process is reversed when data has to be transmitted.

Raabta modifies this existing architecture in three key ways: (1) Any frequency band that is not being used for TV channels is utilized for video conferencing, (2) These bands are used for both upstream and downstream data, and (3) MAC operation is disabled in modem hardware and handled by the end-host instead. In case of cable-based Internet connectivity, the CMTS acts as an arbitrator for the medium to enable TDMA over the shared upstream channel. The lack of a central entity complicates the medium access control in case of Raabta. We discuss two possible solutions:

1. FDMA: All free channels, both within the 5-42MHz spectrum and the 50-860MHz spectrum, are sliced into equisized slots which are allocated to each end-host in advance by the operator. For instance, a 200KHz slot ensures a data rate of 700Kbps which can accommodate 512Kbps which is required for very good quality video [27]. On the downside, even if the entire upstream spectrum is used, the number of end-hosts is limited to 135. On the other hand, a 25KHz slot results in 84Kbps which enables reasonable Quality of Experience. This latter scheme can accommodate up to 1480 dedicated connections. Therefore, the choice of the slot size depends on the Quality of Experience and the number of clients that need to be accommodated. Regardless of the slot size, FDMA ensures contention free communication but limits the user base.

2. CSMA/CD: The entire upstream spectrum, 5-42MHz is treated as a large 37MHz channel. Using 256-QAM, this results in a data rate of 258Mbps. For an RTT of 1.6ms, a 258Mbps link requires a minimum frame size of 51600 bytes. DOCSIS uses a frame size of 204 bytes to mix video (MPEG) and data frames over the same channel. We replace this with a large 51600 byte MPEG frame. As the medium is broadcast-based, the cable modem at each end-host will only copy frames which are destined for its MAC address. On the downside, in case of CSMA/CD, collisions at scale will substantially degrade performance. At the same time, CSMA/CD makes efficient use of the available spectrum. In addition, as the medium is shared, the sender during a video conference can broadcast frames to multiple recipients.

Both these technologies explore different points in the design space and allow the operator to choose one based on the requirements of the network. For instance, if the goal is to provide each user dedicated network resources, then the FDMA scheme will be used. In contrast, if the goal is to dynamically scale the QoS of the network, then the CSMA/CD scheme will be desirable.

The rest of the link-layer is identical to the DOCSIS specification. Packets from the layer above it are sliced into 51580 byte long chunks. Each chunk is added as payload to an MPEG frame which has a 4 byte header and 16 byte FEC. Each modem is configured to listen for frames either destined for its MAC address or the broadcast address. DOCSIS also has MAC layer security enhancements for both privacy and authentication. The former enables DES/AES based encryption of frames while the latter ensures that unauthorized devices are not connected to the network. Unfortunately, both of these require a centralized server to store user credentials, which is in tension with our goal to keep the system completely decentralized. Therefore, we advocate the use of PGP and its decentralized "web of trust", to enable end-hosts to self-sign their frames.

4.3 Communication Protocol

While in principle it is possible for a local network to function without a network layer, we use an unmodified IP layer. This has two key advantages: (1) The network can potentially be connected to the wider Internet or other such networks, and (2) IP multicast can be used to efficiently implement video conferencing between more than 2 hosts. For dynamic address allocation any distributed protocol [26] can be used. On top of this network layer, RTP [25] atop UDP is used for transport. RTP provides a timestamp, sequence number, and payload format for each packet. In conjunction with RTP, RTCP is used for synchronization across streams. Similar to Google+ [31], for each call, 4 individual flows are initiated to the same destination port. This port selection is enabled through SIP [22]. Out of the 4 flows, 2 carry video and audio over RTP, respectively, while the other 2 carry synchronization information for the prior two flows atop RTCP.

In case of packet loss, there are two options for recovery in case of real-time streaming: (1) FEC and (2) Retransmission. To avoid delay due to FEC encoding and decoding, blocks have to be kept short but this reduces its efficiency. On the other hand, retransmission adds redundancy. Fortunately, selective retransmission can be employed to keep bandwidth usage in check [31]. Therefore, the communication protocol in case of Raabta uses retransmission by default. It also exploits multi-layer encoding (described in the next section) which is used by the application to ensure that base-layers are always retransmitted. This allows the application to maintain good Quality of Experience even in the face of a high loss rate. As noted by Xu *et al.* [31] FEC is more efficient than retransmission if the loss rate is random and moderate, and the RTT is high. In such cases, our transport dynamically switches to FEC.

4.4 Video Conferencing

The video conferencing application has three components: (1) The user database, (2) Encoding scheme, and (3) User interface. We describe each in turn.

The first time the application is started, a special packet is broadcast across the network, which along with user identification and picture, contains the IP address of the host. Every host which receives this packet, caches the information within its user database. In addition, each host also directly sends its own information to the new host to enable it to quickly populate its database. Finally, after a configurable interval of time, called the *heartbeat* interval, (5 minutes by default), each node broadcasts its information. If a host is not heard from after 3 heartbeat intervals, it is marked as unavailable in the database.

When a call is initiated, the application captures video from the camera, encodes it, and passes it to the communication layer along with destination user information. In case of encoding, generally, three options are available: (1) One-version, (2) Multi-version, and (3) Multi-layer encoding. One-version encoding only generates a single video stream (typically based on the bandwidth of the slowest receiver) and each host within the session receives the same video. On the other hand, multi-version encoding generates a video for each destination based on the recipient's capacity. It is evident that the former scheme is too coarse-grained while the latter wastes precious bandwidth. In contrast to both these encoding algorithms, multi-layer encoding results in a video with multiple layers. This video consists of a base layer and additional layers which incrementally improve the video quality. As a result, the quality of video at the recipient is determined by its downlink capacity. In view of this, we employ multi-layer encoding.

The final component of the application is the interface which is completely text-free, although users have the option of sending text messages. The user database can be scrolled on screen and only the picture of each user on the network is visible. This picture is captured using the camera and as mentioned earlier, broadcast along with user information every heartbeat interval. A single click on a picture displays a dialog which allows the user to initiate a text, voice, or video session through an icon click. A double click by default initiates a video call. The same mechanism can be used to initiate a multi-party conference call. At each step the user is given audio feedback. This interface enables even low-literate individuals to use the system. Moreover, this mechanism can also incorporate voice based control.

5. DISCUSSION

This section discusses how the architecture can easily be extended to support Internet backbone connectivity. In addition, it also discusses some community-wide applications of the system and the incentive for cable operators to support this service.

5.1 Internet Connectivity

The existing Raabta architecture functions without any backbone connectivity but it can be extended to support it via low-cost solutions such as long-distance WiFi [20] or WiMAX [17]. Browsing will be made possible by a standard browser running on the RPi. As an added advantage, this will also enable video conferencing across the Internet. Furthermore, the local storage on the RPi can be used to cache popular content to reduce bandwidth usage [3]. It is imperative to highlight that the monthly bandwidth cost will need to be extremely small to make it affordable for the subscribers or it will have to be paid for by a benevolent NGO or a charity organization [24].

5.2 Community-wide Applications

In addition to person-to-person communication, Raabta can be used to enable telemedicine and distance learning. In the former, patients can consult a medical practitioner directly through video conference. Such initiatives have already been used to treat hundreds of thousands of patients in rural areas [5, 27]. Moreover, vocational lectures can also be provided through this medium. Furthermore, another natural application for Raabta is the dissemination of useful video content, such as "do-it-yourself" guides. Even without Internet connectivity, appropriate content can be brought in through "sneaker nets" [6]. Additionally, sensors connected to the RPi can note different attributes and enable more informed epidemiology.

5.3 Operator Incentive

The only support that is needed from the operator's side is the replacement of unidirectional amplifiers with bidirectional ones. This is just a one-off cost for the operator which can directly be incorporated into the initial connection/registration fee. In addition, the operator can add a minor surcharge to the monthly cable TV bill. Raabta uses the existing coaxial network and as such does not add an additional operational or power expense. Finally, because of its low-cost, any problem with the Raspberry Pi end-host can be rectified via replacement. The entire application stack can be made available through a stand-alone external memory card image that works out of the box.

5.4 Leveraging Existing Infrastructure

While this paper focuses on using the existing cable TV network as a communication substrate, we believe the general principles are applicable to a wide range of other media. The research community thus far has largely concentrated on using technologies that require new installations to provide low-cost services in the developing world, even though existing infrastructure holds much promise. For instance, using electricity transmission and distribution lines for data services is one such appealing option [15].

6. RELATED WORK

Our work is directly inspired by Poor Man's Broadband [23], which aims to make smart use of the limited backbone bandwidth in the developing world. To this end, it uses a BitTorrent-like protocol to directly connect machines through dial-up and share files. In contrast, we use the existing analog coaxial network deployed to support cable TV, to provide low-cost video conferencing. Our work is also complementary to Allen *et al.* [2] who augment the existing analog cable TV network to support video on demand. Specifically, set-top boxes are connected together to form a P2P topology wherein each set-top box caches a particular chunk of a program. If a user requests a program, it is first searched within the P2P network and if found, the request is locally served. While we also employ the cable network as a substrate to form a P2P topology between end-hosts, we focus on video conferencing instead of program caching. Raabta also benefited from the in-depth study of popular video conferencing systems conducted by Xu *et al.* [31]. In particular, the co-design of the system, the use of multi-layering encoding, and the dynamic switching between FEC and retransmission is a direct consequence of the insights they provided into the internals of Skype and Google+. Moreover, Raabta shares goals with the rich-set of work dedicated to providing Internet connectivity to the developing world [17, 4, 18, 20, 21, 14, 10]. But in contrast to Raabta, all of these initiatives require both the deployment of new infrastructure as well as Internet backbone connectivity. Finally, the use of Raspberry Pi in Raabta instead of regular computers is inspired by TinyPC [1].

7. CONCLUSION AND FUTURE WORK

We presented Raabta, a video conferencing system designed from the ground up to enable communication in developing world regions. It keeps both the cost and the power footprint of the system down by using the existing coaxial cable network deployed for cable TV. Furthermore, it achieves fault-tolerance by assembling Raspberry Pi powered end-hosts into a decentralized P2P topology without requiring any Internet backbone. Moreover, the video conferencing application uses multi-layer encoding and dynamic packet loss recovery to make optimum use of the existing spectrum. Finally, to reach a wider audience, the user-interface is completely text-free and makes extensive use of audio feedback.

Our future work includes the implementation and evaluation of the entire system. In particular, the medium access control strategy requires closer inspection and analysis. Furthermore, we plan on conducting a study to gauge the appropriateness of our user interface design. The eventual goal is to deploy Raabta in the wild.

8. ACKNOWLEDGEMENTS

The author would like to thank Anil Madhavapeddy and the anonymous reviewers for their insightful comments and suggestions.

9. REFERENCES

[1] M. Ali and K. Langendoen. TinyPC: enabling low-cost internet access in developing regions. In *Proc. NSDR '07*, 2007.

[2] M. S. Allen, B. Y. Zhao, and R. Wolski. Deploying video-on-demand services on cable networks. In *Proc. ICDCS '07*, 2007.

[3] A. Badam et al. HashCache: cache storage for the next billion. In *Proc. NSDI '09*, 2009.

[4] E. Blantz and B. Baikie. Case Study: The Haiti Rural Broadband Initiative. In *Proc. NSDR '12*, 2012.

[5] E. Brewer et al. The Case for Technology in Developing Regions. *Computer*, 38(6), June 2005.

[6] E. Brewer et al. The challenges of technology research for developing regions. *IEEE Pervasive Computing*, 5(2), Apr. 2006.

[7] Cable Television Laboratories, Inc. Data-Over-Cable Service Interface Specifications. Online, 2009. `http://www.cablelabs.com/specifications/CM-SP-RFIv2.0-C02-090422.pdf`.

[8] P. Colleman. Cable Modems. Online. `http://home.scarlet.be/~pcoleman/extra_info/Cable%20modem.pdf`.

[9] R. Flickenger et al. Very long distance WiFi networks. In *Proc. NSDR '08*, 2008.

[10] Z. Foo et al. A case for custom silicon in enabling low-cost information technology for developing regions. In *Proc. ACM DEV '10*, 2010.

[11] Gallup Pakistan. Media Cyberletter. Online, 2012. `http://www.gallup.com.pk/News/Gallup%20Media%20Landscape%202012%20Cyberletter.pdf`.

[12] Gallup Pakistan. Trends in the use of Electric goods in Pakistan. Online, 2012. `http://www.gallup.com.pk/ITMarket.php`.

[13] G. Halfacree and E. Upton. *Raspberry Pi User Guide*. Wiley, 2012.

[14] K. Heimerl et al. Expanding rural cellular networks with virtual coverage. In *Proc. NSDI '13*, 2013.

[15] H. Hrasnica, A. Haidine, and R. Lehnert. *Broadband Powerline Communications: Network Design*. Wiley, 2005.

[16] B. E. Kolko, E. J. Rose, and E. J. Johnson. Communication as information-seeking: the case for mobile social software for developing regions. In *Proc. WWW '07*, pages 863–872, New York, NY, USA, 2007.

[17] A. Lele, M. Raj, and D. Das. Providing voice connectivity to rural India using WiMAX: issues and solution. In *Proc. NSDR '07*, 2007.

[18] K. W. Matthee et al. Bringing Internet connectivity to rural Zambia using a collaborative approach. In *Proc. ICTD '07*, 2007.

[19] I. Medhi, A. Sagar, and K. Toyama. Text-free user interfaces for illiterate and semiliterate users. *Inf. Technol. Int. Dev.*, 4(1), Oct. 2007.

[20] R. Patra et al. WiLDnet: design and implementation of high performance WiFi based long distance networks. In *Proc. NSDI '07*, 2007.

[21] B. Raman and K. Chebrolu. Lo3: Low-cost, Low-power, Local Voice and Messaging for Developing Regions. In *Proc. NSDR '09*, 2009.

[22] J. Rosenberg et al. SIP: Session Initiation Protocol, 2002.

[23] U. Saif et al. Poor man's broadband: peer-to-peer dialup networking. *SIGCOMM Comput. Commun. Rev.*, 37(5), Oct. 2007.

[24] A. Sathiaseelan and J. Crowcroft. LCD-Net: lowest cost denominator networking. *SIGCOMM Comput. Commun. Rev.*, 43(2), Apr. 2013.

[25] H. Schulzrinne et al. RTP: A Transport Protocol for Real-Time Applications, 2003.

[26] Y. Sun and E. M. Belding-Royer. A study of dynamic addressing techniques in mobile ad hoc networks. *Wirel. Commun. Mob. Comput.*, 4(3):315–329, 2004.

[27] S. Surana et al. Beyond pilots: keeping rural wireless networks alive. In *Proc. NSDI '08*, 2008.

[28] W. D. Tucker and E. H. Blake. Abstractions for designing and evaluating communication bridges for people in developing regions. In *Proc. ACM DEV '10*, 2010.

[29] E. Upton. Raspberry Pi Camera Board. Online, 2013. `http://www.raspberrypi.org/archives/3890`.

[30] World Bank. Rural population. Online, 2012. `http://data.worldbank.org/indicator/SP.RUR.TOTL`.

[31] Y. Xu et al. Video telephony for end-consumers: measurement study of Google+, iChat, and Skype. In *Proc. IMC '12*, 2012.