Blue Bridge and Linkage

Connecting Africa through 802.11, Bluetooth and Raspberry Pis

Curtis Sahd and Hannah Thinyane
Department of Computer Science
Rhodes University
Grahamstown, South Africa
{curtissahd@gmail.com, h.thinyane@ru.ac.za}

***Abstract*—Public Switched Telephone Networks (PSTNs) are predominantly comprised of proprietary protocols and are not well suited to integration with the ever expanding Voice over IP (VoIP) network. PSTNs have evolved from analogue circuit switched systems and purely fixed line services, to the digital and mobile realms, but still remain proprietary. With the advent of wireless technologies and the explosion in VoIP services, the combination of these two technologies results in a large number of integration possibilities and inter-connection of systems. In this paper, we show how existing technologies can be used in the creation of community telephone networks. Specifically, we provide an overview of our system (Blue Bridge), which performs the bridging of Bluetooth and IEEE 802.11 wireless. We also provide an overview of the associated lightweight hybrid protocol (Linkage), which delivers status updates and data transmissions.**

Keywords-Bluetooth bridging; Community telephone networks; Linkage; Raspberry Pi; Wireless mesh networks

I. INTRODUCTION

Wireless Mesh Networks (WMNs) and VoIP have become key players in IP-based telephony services and have revolutionized the industry in terms of cost, geographically constrained infrastructure and service, and interoperability between devices [6].

WMNs can be defined as dynamic self configuring networks in which all nodes have the ability to route traffic directly to the endpoint, or via a multi-hop path. The network is dynamic, which enables it to deal with nodes entering the network, as well as nodes leaving the network due to node failure or connectivity issues [8]. WMNs are comprised of two types of nodes: mesh routers and mesh clients [1]. Mesh routers provide similar functions to traditional wireless routers, except that they provide more routing functions which are suited to WMNs. Mesh routers achieve the same coverage as traditional routers, but with less transmission power, by means of multi-hop communications [1]. Mesh clients can also work as routers, but do not have gateway or bridge functionality as found with mesh routers. WMN architecture is an important consideration when determining how and where the WMN should be implemented. WMN architecture can be classified according to three primary implementations: infrastructure WMNs, client WMNs, and hybrid WMNs [2]. The backbone of an Infrastructure WMN is predominantly comprised of mesh routers with which mesh clients associate.

Client WMNs provide peer-to-peer network functionality, and are mostly comprised of mesh clients which perform the routing of packets.

Hybrid WMNs are a combination of Infrastructure and Client WMNs, with the infrastructure being comprised of both mesh routers and mesh clients [2].

The concept of WMNs has been around for quite some time, and the development of protocols which enable the efficient functioning of these networks has been a core focus area. Since the inception of VoIP, the way in which we communicate has drastically changed, enabling long distance phone calls at a fraction of the cost when compared to previous years. Although easily and universally accessible, VoIP communication has traditionally taken place with somewhat limited mobility, with instances such as calls being made from fixed landlines and scattered wireless hotspots. There still exists a heavy reliance on mobile networks utilizing proprietary protocols, and with the cost of data being close to that of calls, the reality of reliable VoIP communication over mobile networks is an optimistic idea. Over and above the cost of data communications on mobile networks, the filtering and throttling of VoIP protocols on these networks is not an uncommon practice [7] [12]. Extensive development of the IEEE 802.11 wireless protocol has taken place and drastically increased the ease with which we are able to communicate, as well as the general mobility of people and services. IEEE 802.11 wireless technology provides reliable mobile communication with equipment costs being a fraction of those encountered with traditional radio networks.

A community, city, and even country wide WMN running open standards can prove to be a feasible solution in providing cost effective communication along with ease of integration and expandability.

As such, we propose the implementation of a WMN-based Community Telephone Network (CTN), which enables communication for Bluetooth and IEEE 802.11

 ISBN: 978-1-61208-299-8

wireless clients by means of distributed bridging nodes (*Blue Bridges*) and a lightweight protocol enabling Push To Talk (PTT) communication and signaling (*Linkage*). Section II provides an overview of the Bluetooth protocol and the benefits it provides in the South African context, as well as the associated constraints. Section III introduces PTT systems and provides a brief overview of how they aid in minimizing scalability concerns and constraints inherent in the Bluetooth protocol. Section IV, then introduces CTNs and outlines the benefits of such networks as well as a brief literature review of the common issues encountered with these networks. Section V builds on the ideas introduced and discussed in the aforementioned sections and discusses the various components of our prototype system (*Blue Bridge*). Section VI provides an in depth discussion of the functioning of our hybrid notification and communication protocol (*Linkage*). Future work is discussed in Section VII and this paper is then concluded in Section VIII.

II. Bluetooth overview

Bluetooth is a low powered, low cost, and short range wireless Radio Frequency (RF) technology, which operates in the unlicensed 2.4 GHz Industrial Scientific Medical (ISM) frequency range [9]. Bluetooth is comprised of three classes ranging from 1m to 100m. Bluetooth was originally developed to alleviate the problems caused by incompatible connectors, and also as a cable replacement technology [3].

Although, while very limited in terms of bandwidth and scalability, Bluetooth proves to be a widely adopted protocol in South Africa, with a large portion of the population not able to afford IEEE 802.11 wireless enabled mobile phones.

Bluetooth communication is typically used in Wireless Personal Area Networks (WPANs), which are best suited to ad hoc communication between devices within close proximity of one another. Piconets are the most common type of Bluetooth WPAN, and enable communication between a maximum of 7 active slave nodes and 1 master node. Master nodes control the ability of slave nodes to transmit on the channel [4].

The obvious drawback of Piconets is the maximum number of nodes which are able to simultaneously communicate. The total available bandwidth for Class 2 Bluetooth chips (commonly found in mobile devices) is 800 kb/s, which severely limits Bluetooth in terms of scalability [4]. PTT-based communication requires minimal bandwidth and also places a low memory footprint on constrained devices such as those commonly used throughout impoverished communities in South Africa. The next section provides a brief overview of PTT systems.

III. Push to talk communication systems

PTT communication is a well-known system used in two way radios, which eliminates the need for call signaling procedures encountered in traditional cellular calls. PTT calls can be established between two users, or alternatively, in group communication, between multiple users [2]. PTT-based communication systems prove to be considerably more cost effective than cellular or landline-based services in that charges are only deducted for the time used while speaking, and not for the elapsed time of the call.

Some of the benefits of IP-based PTT systems include: faster communication and less call setup overhead, group communication, integration with existing applications on LANs, unlimited range and improved costs [12]. PTT-based systems can be compared to instant messaging systems in that communication is instant and requires minimal call setup and signaling. Apart from faster communication than traditional systems, PTT systems provide group communication, which proves to be incredibly cost effective in organizations consisting of teams of employees. One of the major drawbacks of two way radio communication is the range at which communication can occur. LAN-based PTT systems overcome the range limitation of traditional radio network PTT systems and integrate with existing network infrastructure and software.

LAN-based PTT systems prove to be a cost effective solution to communication in the African context, and as such we have chosen a PTT-based system for the implementation of our CTN.

The next section introduces WMN-based CTNs and shows the relationship between the number of hops and supported calls on the network.

IV. Community telephone networks

The idea of VoIP communications over IEEE 802.11 wireless networks is by no means a novel one, and the benefits of such implementations are well researched with plenty of optimizations available [6]. For the purposes of this research, we will define CTNs as those which are run by the community, for the community, consisting of open protocols, and subscribed to with very little or no cost. We envision the architecture of CTNs, to be comprised of either fixed wireless access points spanning the area of the community concerned, or WMNs where clients participating in the CTN are an extension to the network and its functionality.

In the case of WMN-based CTNs, it is important to understand the effect of multiple hops between clients on the efficient functioning of the network. Ganguly et al. [6] found that eight calls are supported when each call utilizes a single hop on a 2 Mb/s wireless link en route its destination. They observed that as the number of hops increased from one to four on the 2 Mb/s wireless link, the number of supported calls decreased from eight to one. They suggest, that reason for the drastic drop in the number of supported calls is in converse relation to the number of hops. This can be attributed to the following: 1) a decrease in the User Datagram Protocol (UDP) throughput as a result of self interference; 2) large amount of packet loss due to the increased number of hops and the subsequent need for forwarding; 3) a high protocol overhead for the small VoIP packet sizes (20 bytes for IEEE 802.11 IP/UDP/RTP). They suggest two approaches which reduce the effects of multiple hops on the number of simultaneous calls supported: packet aggregation and packet header compression. They found a 200% - 300% increase in the capacity by implementing these methods.

Due to the dynamic creation and association capabilities of WMNs, they are ideal for peer-to-peer-based systems and are thus very suitable as the underlying platform for CTNs. As seen in [1] and [6], the main constraint regarding WMNs is the number of hops and the throughput each hop is capable of carrying.

Apart from being limited by the number of hops and available bandwidth, WMNs and Bluetooth often suffer from a large number of lost packets which are related to the interference caused by the plethora of devices operating in the 2.4 GHz frequency range. With an overview of PTT systems and CTNs, the next section introduces *Blue Bridge*, and demonstrates the important role it plays in the creation of cost effective CTNs in the South African context.

V. BLUE BRIDGE

Blue Bridge aims to combine the Bluetooth and IEEE 802.11 wireless protocols across a series of Raspberry Pi [11] computers, in an effort to create a CTN using existing technologies. Due to the transmission limitations of the Bluetooth protocol, we propose the implementation of *Blue Bridge* on multiple Raspberry Pi computers in order to create an array of Bluetooth hotspots. People within range of the Bluetooth hotspots will be able to connect to the CTN and in turn place calls to other parties connected via Bluetooth or IEEE 802.11 Wireless.

In order to avoid a large amount of broadcast traffic in an already bandwidth constrained network, we found it necessary to centralize client specific information on the Centralized Authentication and Accounting Server (CAAS).

Figure 1 shows architecture of our proposed CTN:

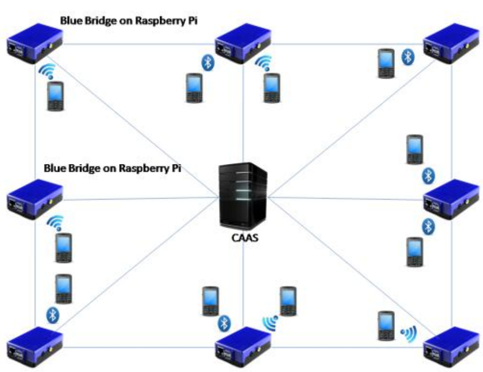

Figure 1. CTN Architecture

From Figure 1, it can be seen that multiple *Blue Bridges* form the infrastructure of the WMN. Bluetooth and IEEE 802.11 wireless clients connect to the CTN through nearby *Blue Bridges*. The CAAS minimizes broadcast traffic by centralizing the status of each client, as well as which router each client is currently associated with. In the event of clients moving between hotspot locations, updates are sent to the CAAS with the newly associated access point address and the status of the client. Multiple IEEE 802.11 wireless clients are able to associate with each *Blue Bridge*, however, a total of 7 active Bluetooth clients are able to communicate with each *Blue Bridge* due to constraints inherent in the Bluetooth protocol.

Connections can be made to and from *Blue Bridges* via the external Bluetooth and IEEE 802.11 Wireless interfaces attached through the USB ports on the Raspberry Pi. Figure 2 provides an overview of the internal structure of each *Blue Bridge* and shows how connections on the same interface as well as varying interfaces are bridged:

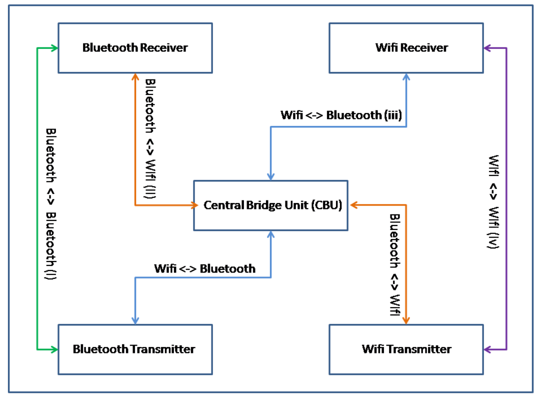

Figure 2. *Blue Bridge* Architecture

From Figure 2, it can be seen from step (ii) that incoming Bluetooth connections (via the Bluetooth Receiver) are sent to the Central Bridge Unit (CBU), which performs the necessary bridging and compression and forwards the data and control packets to the called party on the IEEE 802.11 wireless interface (IEEE 802.11 Wireless Transmitter). Since the concerned *Blue Bridge* is connected to other *Blue Bridges* via the WMN, incoming Bluetooth connections can be connected to any other client on the network without encountering typical range limitations inherent in the Bluetooth protocol. Incoming connections via the IEEE 802.11 wireless interface can similarly be bridged with clients on the attached Bluetooth interface.

In the event where two clients connected to the same *Blue Bridge* via the Bluetooth interface want to communicate, a connection from the calling party is first made to the *Blue Bridge* which then forwards the connection to the called party. This process can be seen in Figure 2 (i). In terms of enabling communication between clients communicating with one another via Bluetooth, there is no need for the *Blue Bridge* to perform any bridging functions or be involved with the communication process at all. With that said, a number of advantages of routing calls/communication via the *Blue Bridge* exist: the ability to perform authentication and accounting functions, which provides useful statistics for network and call monitoring; the ability to create and participate in group calls where clients are distributed across the CTN. There are of course disadvantages of routing local connections via the *Blue Bridge*: since the Bluetooth Master (Bluetooth interface of each *Blue Bridge*) communicates with other locally connected Bluetooth clients, the bandwidth is divided among each connected client and thus reduces the quality of calls;

 ISBN: 978-1-61208-299-8

and increased processing requirements are placed on each *Blue Bridge*.

An SDL diagram of the functioning of the Blue Bridge can be seen in Figure 5. The process can be outlined as follows: the *Blue Bridge* listens for and accepts new connections. Upon accepting a connection the *Blue Bridge* attempts to find the routing information and status of the called client locally (and if not found locally, the CAAS is queried, as seen in Figure 4). This process is illustrated by (a) in Figure 5. Upon finding the routing information for the destination device, the *Blue Bridge* determines the interface from which the data should be sent (as seen in b). If the destination device is connected via the IEEE 802.11 wireless interface, data is forwarded between the concerned devices (as seen in steps c and d). However, if the destination device is connected via the Bluetooth interface, the number of active connections needs to be determined (as seen in g). If the number of active Bluetooth connections is less than or equal to 7, then data is forwarded between the participating devices (as seen in h). In the event where the number of active connections is greater than 7, the *Blue Bridge* terminates the connection and sends the "calling" status to the CAAS as well as the client (not depicted in Figure 5).

With an overview of *Blue Bridge* and how it performs the bridging of the Bluetooth and IEEE 802.11 Wireless protocols, the next section introduces our signaling and voice payload protocol, *Linkage*.

VI. Linkage

Although each *Blue Bridge* performs the bridging between the IEEE 802.11 wireless and Bluetooth protocols, without the ability for *Blue Bridges* to communicate, the CTN would cease to exist. We therefore propose *Linkage*, which facilitates inter-communication between *Blue Bridges* and between each *Blue Bridge* and the CAAS. *Linkage* is a lightweight hybrid (UDP and TCP) protocol, which performs the following functions: determining status updates and which *Blue Bridge* each client is currently associated with; informing the CAAS of new clients connected at each *Blue Bridge*; informing the CAAS of existing clients associating with new *Blue Bridges*; status updates between clients, and between *Blue Bridges* and the CAAS; and the transportation of traffic between *Blue Bridges* and clients.

Traditional voice communication is generally session-based, which means that communication between clients and servers is request-response-based. Since voice communication is essentially the transmission of voice data from one client to another and the subsequent playback of this voice data at the receiving end, we decided to abandon the traditional session-based model of voice communication. As such, packets are transmitted from one client to another (via *Blue Bridges*) in a best effort attempt without the need for exchange of information between clients.

We envision a CTN, to be one in which all protocols are open and easily expandable. As such, we designed *Linkage* in such a way that it can be extended beyond its original function (transportation of voice data between *Blue Bridges* and clients) and transport various other types of traffic by means of custom field addition in the protocol structure.

In order for communication between clients to take place, the following processes are followed:

1. The calling client communicates with the CAAS to determine which *Blue Bridge* the destination client is currently associated with.
2. Once the calling client has knowledge of the destination *Blue Bridge*, it then sends data (voice data in the case of our CTN) via *Linkage* to the destination *Blue Bridge*.
3. Upon receipt of voice data the concerned *Blue Bridge* utilizes the information contained within *Linkage* to route the packet to the destination client.
4. The destination client can then respond to the calling client by utilizing information contained within *Linkage*.

In order to better understand how *Linkage* performs the necessary functions required for the implementation of a CTN we provide an overview of the protocol structure and the components used throughout the process above.

Figure 3 shows the structure of *Linkage* packet*:*

```
<packetId>

    <source>

        <sourceNumber> </sourceNumber>

        <sourceBB> </sourceBB>

        <state> </state>

        <totalPackets> </totalPackets>

        <currentProgress> </currentProgress>

        * <customFields> </customFields>

    </source>

    <destination>

        <destinationNumber> </destinationNumber>

        <destinationBB> </destinationBB>

        * <customFields> </customFields>

    </destination>

    <Data> </Data>


</packetId>
```

Figure 3. Linkage packet structure

Due to the nature of our CTN and for the purposes of efficiency, we utilized the UDP protocol as the underlying protocol for communication between the clients and *Blue Bridges*, and the TCP protocol for communication between *Blue Bridges* and the CAAS. UDP was chosen as the primary protocol between clients and *Blue Bridges* due to the fact that potential lost packets would only result in a decrease in audio quality and would not cause the CTN to cease functioning. TCP was chosen as the preferred protocol for communication between *Blue Bridges* and the CAAS to ensure reliable conveyance of client status information, new client registrations, and reporting of statistical data.

For the purposes of client and *Blue Bridge* communication, we propose the addition of the following fields to the UDP protocol: *Linkage* specific fields, and the payload data field which carries the voice data. In cases where *Linkage* is used merely for informational transfer, the payload data field is left empty.

From Figure 3, it can be seen that the *Linkage* fields are as follows: the packet ID field; client source information; and client destination information.

The PacketID field uniquely identifies each packet for the purposes of accounting, and returned routable responses.

The client source is comprised of the telephone number of the calling client; the IP address of the source *Blue Bridge;* the current state of the client; the total number of packets being transmitted during the voice communication session (if applicable); the current progress of the voice transmission (if applicable); and any additional number of custom fields. The current progress, and total number of packets for the voice communication session, play an important role in optimizing the CTN in that they allow the CAAS to change the status of communicating clients as soon as the voice transmission is complete. These fields also ensure that statuses are updated in the event of lost packets.

The client destination is comprised of the telephone number of the called party, the IP address of the destination router with which the called party is associated; and then any additional custom fields. Due to changing IP addresses, the client telephone numbers serve as the main identifier for communication in the CTN.

A. Statuses and Notifications

In order for the CAAS to be notified of new clients joining the CTN, the above *Linkage* packet is ideal in that new clients can encapsulate their telephone number, and the *Blue Bridge* with which they are associated, as well as their current status. In order to minimize network congestion and unnecessary strain on clients, *Blue Bridges* are required to inform the CAAS of new client associations as well as changes in client statuses. A client can have the following statuses: available; calling; and offline.

It is important to notify calling clients of the status of called clients, so as to avoid engaged calls being mistaken for non-existent clients. An example of this would be, where Client A attempts to call Client B and Client B is currently participating in another call. As such, we have implemented a status field so as to notify the CAAS of the statuses of communicating clients. This scenario can be seen in Figure 4:

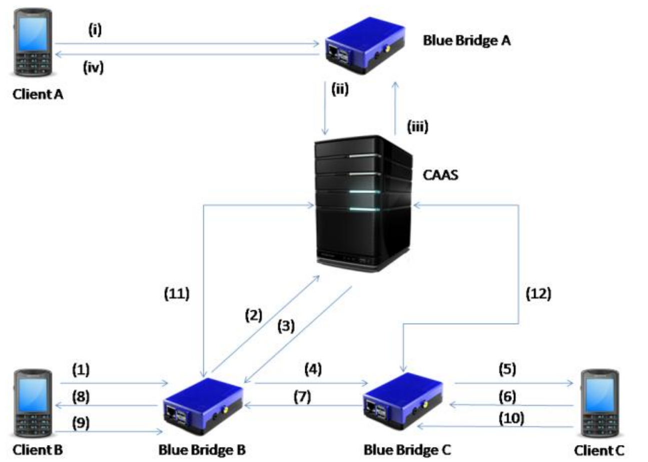

Figure 4. Typical communication session

From Figure 4, Client B and Client C are in a communication session with one another. Client A attempts to call Client B by querying Client B's information from Blue Bridge A (shown as (i) above). Since Blue Bridge A does not have any information pertaining to Client B, it then queries the CAAS for this information (show as (ii) above). The CAAS retrieves Client B's status from the database, and determines that Client B's status is "calling". The CAAS then alters the status field of the *Linkage* packet to "calling" and notifies Blue Bridge A (shown as (iii) above). Blue Bridge A then notifies Client A that Client B is engaged and terminates the connection (shown as (iv) above).

A typical communication session between Client B and Client C can be seen in Figure 4. The communication process begins with Client B querying Blue Bridge B for information pertaining to Client C. The *Linkage* packet sent to Blue Bridge B contains the first media packet to be transmitted to Client C. Since Blue Bridge B does not contain any routing information pertaining to Client C, it strips and stores the voice data temporarily while it attains this routing information from the CAAS (steps 2 and 3). If the CAAS determines that the status of Client C is "available" it updates Client C's status to "calling", which will prevent unnecessary call setup when other clients try to call Client B or C while in the "calling" state.

Throughout the attainment of Client C's routing information, Client B continues to transmit voice data to Blue Bridge B. Blue Bridge B records packet information in a local MYSQL database which references the stored Adaptive Multi-Rate (AMR) media files. Once Blue Bridge B has determined the necessary routing information of Client C, it processes the stored queue of media data and transmits the packets to Blue Bridge C (as seen in step 4). Since Blue Bridge C has a local record of Client C being associated with it, packets can be sent directly to Client C without the need to obtain routing information from the CAAS (as seen in step 5). Blue Bridge C's local database is then also updated with the routing information of Client B. In order to eliminate the storage of non-current client routing information, the local database of Blue Bridges is purged every 10 minutes. Client

C is then able to respond to Client B by sending the voice data to Blue Bridge C (as seen in step 6) which forwards the packet to Blue Bridge B (step 7). Blue Bridge B in turn forwards the data to Client B (step 8). Since both Blue Bridge B and C contain the necessary routing information, communication between Clients B and C then takes place via Blue Bridges B and C (independent of CAAS queries and updates).

When Client B is done communicating with Client C, a status packet (stripped of all voice data) is sent to Blue Bridge B (step 9), which then updates the local database of the status of Client B. Blue Bridge B then notifies the CAAS of the status update (the two way arrow in the diagram indicates a TCP connection, which ensures delivery of the packet - step 11). Similarly, Client C, Blue Bridge C, and the CAAS perform the necessary updates and termination of the communication session as seen in steps 10 and 12.

B. Lost Packets

Since *Linkage* is a best effort based protocol operating across interference prone wireless protocols, the likelihood of lost packets is quite high. In order to minimize the effect of lost packets on the functioning of the CTN we transmit the state the client is currently in; the total number of packets to be transmitted; and the current progress of the transmission from each client transmitting data to its associated *Blue Bridge*. The last packet transmitted from either client contains the "available" state, which notifies the concerned *Blue Bridge* that the client is not participating in a communication session.

In the event where the last packet is lost, the associated Blue Bridge is able to determine when the communication session should have ended based on the total number of packets and the last noted progress. The concerned Blue Bridge then terminates the connection and changes the status of the client to "available

VII. Future work

Although channeling all communication through *Blue Bridges* may result in increased delays throughout the CTN, this type of architecture allows for the system to be extended by providing group calling functionality. This is made possible by Blue Bridges forwarding voice packets to everyone in a particular group. By enabling group broadcasts on Blue Bridges, processing requirements of constrained client handsets is reduced. The ability to provide instant messaging functionality, and create community polling and support systems are other possible extensions. Community polling is particularly useful in areas where there is a shortage of staff. This allows the community to inform the municipality of areas which require attention. These polling systems could also provide benefits for the security and emergency industries, providing a means for people to alert authorities in the event of an emergency.

Our CTN network implementation currently only provides node status updates from *Blue Bridges* to the CAAS when calls between clients are complete; when new clients join the CTN, or when Blue Bridges terminate communication sessions in the event of suspected packet loss. The disadvantage of status updates only in these instances is that client statuses on communicating *Blue Bridges* are updated, but the CAAS does not have the most recent client statuses (except where clients have not communicated since their last session). A possible extension could be to enable client status updates upon *Blue Bridges* querying routing information for destination clients from the CAAS.

VIII. Conclusion

This paper provided an overview of WMNs, CTNs and the Bluetooth protocol, as well as instances of how these technologies and systems are implemented in related work. This paper demonstrated a need for cost effective communication methods utilizing existing technologies and provided an architectural overview of our proposed CTN. This paper explained the benefits and potential constraints of the system, as well as possible solutions to these constraints. This paper identified the problems associated with voice communication in South Africa; provided an overview of existing systems and where they can be improved; and proposed a solution to these problems by means of bridging Bluetooth and IEEE 802.11 wireless connections on a series of Raspberry Pi computers (*Blue Bridge*). This paper also demonstrated how a CTN can be created through the inter-connection of these *Blue Bridges* by means of the proposed protocol - *Linkage*. Finally, this paper suggested possible extensions to the current infrastructure and how these extensions could be implemented.

Acknowledgment

We would like to thank our sponsors, the Telkom Centre of Excellence at Rhodes University, funded by Telkom SA, Business Connexion, Verso Technologies, THRIP, Stortech, Tellabs and the National Research Foundation.

References

[1] F. Ian Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," Computer Networks., 2005, vol. 47, no. 4, pp. 445-487.

[2] At&t., "Learn more about Push to Talk," 2012. Available at: http://www.wireless.att.com/learn/popups/push-talk-faq.jsp. [Accessed 04-10-2013].

[3] C. Bisdikian, "An overview of the Bluetooth wireless technology," Communications Magazine IEEE., 2001, vol. 39, no. 12, pp. 86-94.

[4] R. Bruno, M. Conti, and E. Gregori, "Bluetooth: Architecture, protocols and scheduling algorithms," Cluster Computing., 2002, vol. 5, no. 2, pp. 117-131.

[5] M.C. Castro, P. Dely, J. Karlson, and A. Kassler, "Capacity Increase for Voice over IP Traffic through Packet Aggregation in Wireless Multihop Mesh Networks," IEE Journal., 2007, vol. 2, pp. 350-355.

[6] S. Ganguly, V. Navda, K. Kim, A. Kashyap, D. Niculescu, R. Izmailov, S. Hong, and S.R. Das, "Performance Optimizations for delpoying VoIP Services in Mesh Networks," IEEE Journal, November 2006, vol. 24, no. 11, pp. 2147-2158.

[7] R. Huang, "Heavy-handed network throttling 'bad for business'," 2012. Available at: http://www.zdnet.com/heavy-

handed-network-throttling-bad-for-business-7000001953/. [Accessed 03-21-2013].

[8] J. Ishmael, S. Bury, D. Pezaros, and N.J.P Race, "Deploying rural community wireless mesh networks," IEEE Journal, 2008, vol. 12, no 4, pp. 22-29.

[9] P. Johansson, R. Kapoor, A. Kazantzidis, and M. Gerla, "Rendezvous scheduling in Bluetooth scatternets," IEEE International Conference., 2002, vol. 1, pp. 318-324.

[10] Push to Talk LTD., "Benefits of PTT," 2011. Available at: http://www.pushtotalkuk.com/push_to_talk_benefits.aspx. [Accessed 04-09-2013].

[11] Raspberry Pi., "An ARM GNU/Linux box for $25," 2012. Available at: http://www.raspberrypi.org. [Accessed 04-08-2013].

[12] B. Tubb, "Operators deny VOIP throttling," 2013. Available at: http://www.itweb.co.za/index.php?option=com_content&view=article&id=61979. [Accessed 03-21-2013].

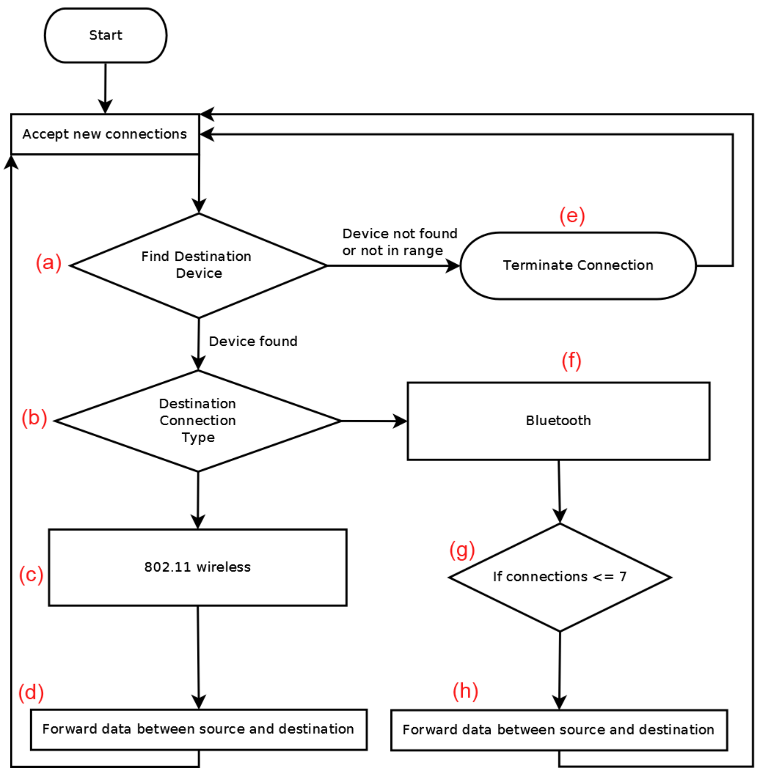

Figure 5. Process diagram for proposed CTN

Copyright (c) IARIA, 2013. ISBN: 978-1-61208-299-8