A Study on the Effect of Transmission Power Adaptation and Multi-hop Path Usage on Power Consumption and QoS in Adaptive Mobile Video Delivery

Martin Kennedy and Gabriel-Miro Muntean

Performance Engineering Laboratory

Dublin City University (DCU)

Dublin, Ireland

martin.kennedy@ieee.org, gabriel.muntean@dcu.ie

***Abstract*— Smart-phones have become a ubiquitous technology, replacing a large number of previously independent digital devices. While the functionality of these smart mobile devices is increasing exponentially, the devices are limited in terms of practical use because of their battery life. This paper introduces PowerHop, a novel algorithm which combines three linked approaches for reducing the power consumption of a smart-phone while it is transmitting video content. The first approach is to dynamically reduce the transmission power of the device's Wireless Network Interface Card (WNIC). The side-effect of a reduction in the transmission power is that the range of the transmissions is also reduced. The second approach of PowerHop compensates for this range reduction by transmitting data to an intermediary relay node. The final approach of the PowerHop algorithm is to adapt the video quality in response to loss on the transmission link. The effect of the transmission power adaptation, multi-hop paths and the video quality adaptation, on the power consumption of a device is measured in real-world tests on Android devices. In addition, the effect of the transmission strategy on the Quality of Service (QoS) of the video transmission application is also analyzed.**

Keywords- VoD, interactivity, datacasting; Field trials and test results; Mobile, portable, and handheld devices; Performance evaluation; Objective evaluation techniques; Propagation and coverage; Traffic and performance monitoring; Networking and QoS; Energy efficient multimedia;

I. INTRODUCTION

It is estimated that in 2013, the number of mobile-connected devices exceeded the number of people on Earth for the first time [1]. These devices include laptops, tablets and smart-phones, among others. The same report highlights that between 2012 and 2017, global mobile data traffic will have increased 13-fold and that video will account for two-thirds of this traffic. While the networks and mobile devices used to handle this traffic are constantly improving in terms of functionality and efficiency, one area that is not keeping pace is the improvement in battery technology [2]. Different applications running on smart-phones have different power consumption profiles, as has been illustrated in our previous research [3]. Some of the most energy intensive applications on a smart-phone involve video streaming or videoconferencing, which poses a problem for the projected dominance of this traffic type.

The solution to the issue of short battery life in smart mobile devices is not as simple as installing batteries with higher capacities in the devices. There is a strict limitation in the design of a smart-phone in terms of physical space that prohibits this "quick-fix" from being possible. The solution instead lies with the creation of an intelligent, dynamic mechanism for utilizing the hardware components on a device in an energy-efficient manner, while also meeting the QoS requirements of the applications running on the device.

This paper tackles the problem of high power consumption in a smart-phone while it is transmitting video content, e.g. videoconferencing or live video broadcasting apps. It proposes and presents **PowerHop, a novel algorithm for balancing energy saving and quality during mobile video delivery** in a wireless network environment. PowerHop, performs adaptations to the transmission power of a device, decides the number of hops to include in the communication route and dynamically scales the video quality. PowerHop assesses network conditions, neighboring node devices and QoS requirements in order to decide whether or not to adapt the transmission power, whether to use a direct or multi-hop route for communication and whether to increase or decrease the video quality level.

The remainder of this paper is organized as follows. In Section II different video transmission platforms are introduced and energy saving mechanisms for video transmission applications are discussed. Following that, the architecture of the system and the PowerHop algorithm are introduced in Section III. The test set-up and results are presented in Sections IV and V and then conclusions and future work are described in Section VI

II. RELATED WORK

With the exponential increase in the computational power of modern smart mobile devices, a whole new area has opened up in the video streaming space. The capture and live broadcasting of video online used only be possible with specialist equipment. However, most modern smart-phones now support high definition video capture and have the ability to perform H.264 video encoding/decoding. This functionality means that a smart-phone can be used as a video broadcaster.

In this section, the architectures of the most popular mobile video broadcasting applications are introduced. Following this, the state-of-the-art approaches used for reducing the power consumption while performing a mobile video broadcast are discussed in some detail.

A. Mobile Video Broadcasting Platform Architectures

There are number of different architectures used for live video broadcasting applications. The first is the direct streaming approach. IP Webcam [4], for instance, serves the video directly from an Android device, without using a video server platform. This approach functions perfectly when transmitting video to one or two people, video conferencing for instance, but one limitation is that the system is not very scalable. As the number of people watching the video broadcast increases, more strain is placed on the mobile broadcasting device.

A solution to this issue is to have a cloud-based distribution system. Two of the largest video broadcasting platforms, Ustream.tv [5] and Veetle.com [6], have apps for iOS and Android devices. These enable users to broadcast live video content from their phones and tablet devices through online portals, where other people can view the streams. Both of these applications rely on the backend cloud-based systems provided by Ustream and Veetle respectively. These systems receive the video content from the mobile broadcaster and are then used to handle viewer requests and the distribution of the content.

BitTorrent Live [7] is a system which has addressed the issue of scalable video delivery without the need for a cloud-based backend. Using the Peer-to-Peer style of network that forms the backbone of Torrent file downloads, the video stream can be distributed between viewers. Each viewer can then pass on sections of the stream data to other viewers, so that all nodes on the Peer-to-Peer network can view the whole stream. The benefits of this architecture are that the cloud-based video server is not required for scalability issues and the broadcast becomes more stable when more people attempt to view it. In addition, this peer-to-peer architecture lends itself nicely to the multi-hop architecture of the PowerHop algorithm, proposed in this paper.

B. Power-Saving Techniques for Mobile Video Transmissions

While transmitting video from a mobile device, battery-life quickly becomes a large stumbling block. There are a number of different techniques which can reduce the power consumption on a mobile device that will now be discussed in detail. These techniques are specifically geared to yield savings during transmissions. For other relevant power saving techniques see Kennedy *et al*. [8], Trestian *et al*. [9] and Moldovan *et al*. [10].

Transmission Power Adaptation: The Transmission Power setting of a wireless card is used to configure the gain applied to a device's radio antenna, during data transmission. This gain level dictates the Signal to Noise Ratio (SNR) of the transmission and thus the range of successful transmissions. While setting the device to the maximum transmission power level will result in the data being successfully transmitted over a larger distance, this is not always an ideal solution. A high transmission power directly results in a high power consumption level on the device's battery. Setting the transmission power to the lowest level can also result in increased power consumption. This can occur when the SNR becomes so low that loss on the network rises. This in turn can require a higher level of packet retransmission, depending on the application and transport layer protocols.

Lu *et al.* [11] proposed a mechanism for optimizing the process of encoding and transmitting H.263 video over wireless links. This mechanism set the INTRA frame frequency in the encoding process, dynamically managed the channel coder and also adapted the transmission power on the device in response to information about the video communication link. The authors were able to prove that for successful transmission of video across a wireless network, the transmission distance greatly effects the power consumption of the sender device. For larger distances, the transmission power must be increased to ensure delivery of the data. This in turn increases the power consumption on the device. While this paper provides very useful insights, it does not consider the use of multi-hop paths for saving power. Additionally, the tests were performed with low resolution video sequences and laptop computers, not smart-phones/tablet devices.

In [12], a novel cross-layer, state-machine based algorithm is presented for limiting the loss rate of important video frames in a H.264 stream while keeping the device power consumption stable. The algorithm aims to maintain a stable level of power consumption on the device while increasing the QoS of the stream received by the viewer. This is achieved by configuring a higher transmission power for the more important video frames (i.e. Intra frames) in the video sequence. In addition, a feedback mechanism is used to allow dynamic control of the transmission power so that it can react appropriately to packet loss on the network. The simulation results show that for a minor power-consumption overhead, the QoS of the received video can be increased slightly. These test scenarios have not been attempted on real-world devices however and do not include any provision for multi-hop routing either.

Multi-hop Paths: When transmitting data over large distances, the transmission power of the wireless interface needs to be set high, so that the data will be received. Multi-hop routing can be used to circumvent this issue by allowing the sender to transmit its data to a nearby node, which then forwards the data on to the destination node. The benefit of this approach is that the transmissions take place over shorter distances and as a result, do not require as high a transmission power level. By extending this idea further, with more intermediary hops in the transmission path, the range of the wireless network can be increased greatly too.

Multi-hop routing creates some additional challenges however. For example, why would a mobile device agree to relay data for another device? The most common solutions to

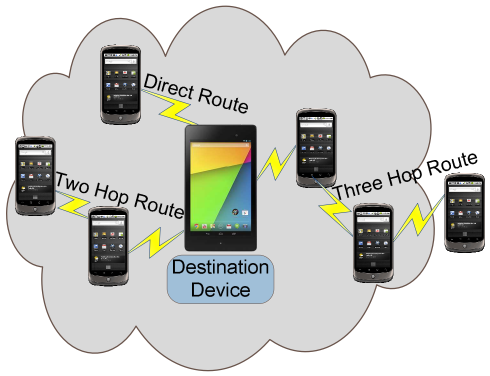

Figure 1 - Network Topologies

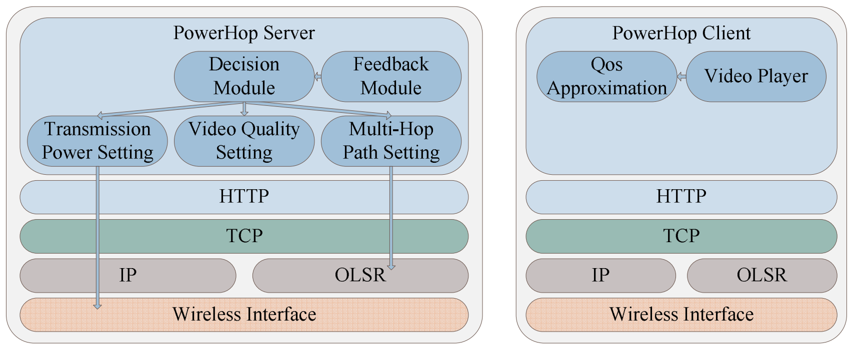

Figure 2 - PowerHop Block Diagram

this issue are to provide a reputation-based [13] or credit-based [14] incentive to the relay device, so that they benefit from helping other devices with their transmissions. Some incentive schemes propose a combination of the reputation and credit-based approaches as seen in [15]. These incentivisation issues are not addressed directly in this paper but it is assumed that they would be in place on the multi-hop network already.

Another issue that is inherent in using multi-hop paths is the selection of the specific path and number of hops to use. In [16], Banerjee *et al.* prove that any energy-aware routing policy should consider the error rate and the probability of retransmissions in order to calculate the true cost of a multi-hop path. By considering the transmission error rate of all hops in their transmission path, the authors were able to achieve a 70% power reduction over other minimum-energy routing protocols. The proposed routing protocol does not consider any specific traffic type or delivery constraints, so does not assess how network latency would affect the playback of a video stream, for instance. This information is crucial for the selection of an appropriate energy-aware path.

Video Quality Adaptation: The power consumption of the sender devices is directly proportional to the amount of data that it is sending. For video streaming applications, the bit-rate of the stream can be altered dynamically. This technique is included in some of the papers mentioned above to augment their power savings, [11] [12].

Kaddar *et al.* [17] proposed an energy-aware video delivery model for transmitting video content to wireless mobile devices in an *ad hoc* network. The proposed model assesses the device characteristics and the battery life of the mobile device and then adapts up or down enhancement layers in an MPEG4 SVC video in order to send an appropriate stream. This changing of video quality changes the traffic on the network. In simulations, the lifetime of the *ad hoc* network can be increased by up to 200%, when compared to streaming the maximum quality of video. While the proposed model is compatible with energy-aware routing protocols and multi-hop paths, these have not been incorporated into the model. Similarly, the residual power of a device is the only characteristic that is monitored periodically. This is a bit of a simplistic implementation.

III. ARCHITECTURE

The PowerHop algorithm is implemented within a custom video streaming application. This application has both client and server components. **Error! Reference source not found.** depicts the network and three different transmission paths: a direct connection, a two hop path and a three hop path. In each case, the smart-phone at the end of the link is serving the video to the tablet in the center of the diagram. While the client device in this case is a tablet computer, this could be implemented with any device capable of handling the video stream. Similarly the client device could be on another network, provided there was a gateway to that network.

A block diagram of the individual components of the PowerHop system, and where they are located, can be seen in Figure 2. For both the client and the server devices, the base system is identical. Each device is running OLSR for *ad hoc* routing purposes and the video data is transmitted using HTTP. On the client device, there are just two additional modules in the PowerHop architecture. The Video Player module handles the playback of the video stream on the device. The Qos Approximation module records when an error occurs during the video playback. This could come in the form of the buffer being empty or a frame being decoded too late to be displayed. Additionally, using the ping command, the latency on the network is approximated and recorded here too. This information is then be used to assess the loss on the network and thus approximate the QoS of the video stream. QoS estimation is performed with the formula as shown in Equation 1 [18]. The data from this module is sent back to the Feedback module on the PowerHop Server.

$$PSNR = 20\log_{10}\left(\frac{Max\_Bitrate}{\sqrt{(Exp\_Thr - Crt\_Thr)^2}}\right) \qquad (1)$$

In Equation 1, *Max_Bitrate* is the maximum data rate of the transmitted stream, *Exp_Thr* is the expected throughput and *Crt_Thr* is the actual average throughput.

On the PowerHop Server, additional modules provide functionality to read the feedback from the client application. This feeds directly into the decision module, where the PowerHop algorithm runs and decides what settings to use for the outputs. The output settings include changing the transmission power of the WiFi interface, the quality of the video stream and the number of hops used in the transmission route.

The PowerHop algorithm considers the estimated PSNR which is made available to the device through the feedback module. There are two metrics that can affect the PSNR level. These are the loss and the latency in the communication link. If either of those metrics rises, then the *Crt_Thr* drops. This in turn lowers the PSNR value. Loss and latency on the network can be assumed to have been caused by either a *low SNR* on the communication link or by *network congestion*. The PowerHop algorithm attempts to tackle both of these root issues. Equation 2 shows the core formula of the PowerHop algorithm. α and β are normalization factors so that the value of U will always be between '0' and '1'. *U* is a utility function and is used in PowerHop's decision making.

$$U = \beta * \ln(PSNR^{\alpha}) \quad (2)$$

When the value of *U* goes below threshold γ^3 (e.g. γ^3 = 0.7), then PowerHop boosts the transmission power up a level. If *U* drops to below threshold γ^2 (e.g. γ^2 = 0.65), then PowerHop switches down one video quality level. If *U* continues to drop and goes below threshold γ^1 (e.g. γ^1 = 0.5), PowerHop looks to switch to using a direct route. If the value of *U* increases above γ^3 again, then PowerHop switches up one video quality level. If *U* increases above threshold γ^4 (e.g. γ^4 = 0.75), then PowerHop lowers the transmission power by one level. If *U* increases above threshold γ^5 (e.g. γ^5 = 0.8), then PowerHop looks for a neighboring node to add as a hop in the transmission path.

IV. Testing Setup and Scenarios

Testing is performed using three HTC Nexus One devices, running Android 4.2.2, as the server and relay nodes. The power consumption of the Nexus One devices is measured externally in real-time and logged to SD cards. This is achieved by using an Arduino microcontroller as seen in Figure 3. An Asus Nexus 7, running Android 4.3, is used as the client device. The full test-bed setup can be seen in Figure 4. The adaptive transmission power control and the selection of the number of hops is implemented through the MANET Manager app on all of the devices [19]. This application performs all the *ad hoc* routing operations using the OLSR protocol and the PowerHop algorithm functions on top of that.

The testing scenarios involve a HTTP Live Streaming (HLS) video stream to be transmitted from the server device to the client device. This streaming is repeated for different distances, for both single and two-hop routes, for different video quality levels and for different transmission power levels. For each scenario, the loss, latency and power consumption of the stream are recorded. The received video on the client device and the power consumption required for the whole transmission can then be analyzed in order to assess

Table 1 - Testing Parameters

	Server Device			Relay Device		
Single Hop	32m	16m	1m			
Two Hop	32m	16m	1m	32m	16m	1m
Tx. Power	32dB	16dB	1dB	32dB	16dB	1dB
Video Rate (Quality Level)	0.3 Mbps	0.9 Mbps	1.5 Mbps	H.264, Baseline		

Figure 3 - Power Measurement Setup

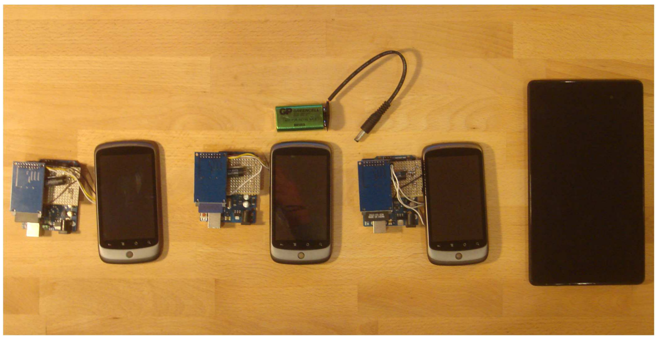

Figure 4 – Test-bed Devices

PowerHop's performance. In total, 270 streaming tests were performed, with all the possible permutations of each of the items in Table 1. The tests were performed in an open space with a line-of-sight path between each of the devices. Additionally, the wireless spectrum was scanned and the WiFi channel selected specifically to prevent unwanted interference with other networks.

V. Test Result Analysis

In this paper, a subset of the results are presented. The results have been arranged in such a way as to illustrate three specific aspects of the tests: the effect of transmission power control, the effect of video quality adaptation and the performance of the PowerHop algorithm. For each of these three arrangements, a transmission distance of 32m was selected. This distance was measured to be the maximum reliable transmission range for the devices in our tests in a direct route.

A. Effect of Transmission Power Adaptation

This first set of results can be seen in Table 2. For one level of video quality, the same stream was sent across the network 9 times. For each of these repetitions, the transmission power of the server and relay device were changed and the transmission path was modified between two different two-hop routes and a direct route. The single-hop route is just a direct wireless connection between the server and client over a 32m wireless link. In the first of the two-hop paths, the server transmits the video to a neighboring node

Table 2 - Effect of Transmission Power Adaptation

	Server Tx. Distance	Relay Tx. Distance	Video Rate	Tx. Power = 32dB					Tx. Power = 16dB					Tx. Power = 1 dB				
				Loss (%)	PSNR (dB)	Delay (ms)	Server Power (W)	Relay Power (W)	Loss (%)	PSNR (dB)	Delay (ms)	Server Power (W)	Relay Power (W)	Loss (%)	PSNR (dB)	Delay (ms)	Server Power (W)	Relay Power (W)
Single Hop	32m	N/A	0.9 Mbps	0	100	6.1	0.336 (0.509)		0	100	11.5	0.356 (0.529)		0	100	3.9	0.356 (0.529)	
Two Hop	1m	32m	0.9 Mbps	78	12.6	74.3	0.337	0.550	6.25	34.54	41.8	0.377	0.522	31.8	20.4	139.3	0.354	0.635
Tx. Power	16dB	16dB	0.9 Mbps	6.25	34.5	20.8	0.389	0.373	9.1	31.3	32.6	0.407	0.362	6.25	34.5	35.8	0.361	0.336

Table 3 - Effect of Video Quality Adaptation

	Server Tx. Distance	Relay Tx. Distance	Tx. Power	Video Quality = High (1.5 Mbps, 720p)					Video Quality = Medium (0.9 Mbps, 480p)					Video Quality = Low (0.3 Mbps, 426x240px)				
				Loss (%)	PSNR (dB)	Delay (ms)	Server Power (W)	Relay Power (W)	Loss (%)	PSNR (dB)	Delay (ms)	Server Power (W)	Relay Power (W)	Loss (%)	PSNR (dB)	Delay (ms)	Server Power (W)	Relay Power (W)
Single Hop	32m	N/A	16dB	59.5	10.5	101.4	0.618 (0.791)		0	100	11.5	0.356 (0.529)		0	100	11.0	0.347 (0.520)	
Two Hop	1m	32m	16dB	18.9	20.5	52.2	0.383	0.604	6.25	34.54	41.8	0.377	0.522	0	100	7.8	0.310	0.382
Tx. Power	16dB	16dB	16dB	18.9	20.5	145.6	0.466	0.580	9.1	31.3	32.6	0.407	0.362	77.6	22.2	178.9	0.412	0.329

which is 1m away. This node then relays the data over the remainder of the link to the client device. In the second two-hop route, the relay device is exactly halfway between the server and the client devices. For simplicity of presentation, when changing the transmission power in the two-hop route, the server and relay devices are configured with identical transmission power levels. There is no reason that the two devices cannot be configured with different power levels for a real-life application.

Before looking at the results for these tests in-depth, there are a couple of important items to take note of. For the single-hop route, the tests were performed with no other nodes in the network. This is possible in an experimental setting, but is not practical in a deployment setting. Additional nodes on the network mean that more energy is spent on the device processing routing information. The "Server Power" column shows the power consumption of the server device for each of the test scenarios. For the single-hop routes, there is an additional number in this cell in brackets. This number refers to the power consumption of the device for the transmission if there is another node on the network. The other node does not have to be involved in the communication, but drives up the power consumption on the server device anyway. In order to compare like-with-like, comparisons between the power consumption of the single-hop and multi-hop routes will use the updated number which accounts for the overhead of other nodes on the network. Another aspect to note is that the PSNR has been capped at 100dB for identical streams with no distortions (i.e. using eq. (1) will result in an infinite value).

The most energy efficient option for this group of test scenarios is to transmit the video content directly, in a single hop. As we can see in Table 2, when there are no other devices in the network, this approach results in the lowest power consumption of all the scenarios, while also providing the highest PSNR value. Unfortunately, this option does not perform quite so well when there are other devices in the network. In these cases a power savings of up to 33% can be achieved on the sender by switching to using a multi-hop route and offloading the long data transmissions onto a neighboring device. The negative aspects of using the two-hop route are that the delay and loss on the network are likely to increase. This can be combated by tuning the transmission power of the devices.

When setting the transmission power it is important to consider the loss on the network. Configuring a high transmission power for devices that are very close to each other can introduce network congestion and loss. Setting the power too low on the other hand means that the SNR of the communication may not be high enough to transmit over the distance required. Both of these issues result in the need for more retransmissions for successful delivery of the data (when using TCP at least), which lowers the PSNR and increases the power consumption. In the testing data in this table, it can be inferred that for the first of the two-hop routes, the combination of the transmission power and the distance between the server and the relay (1m) becomes problematic. For similar rates of power consumption on the server device, an increase in the PSNR of up to 170% can be achieved by switching to the second of the two-hop paths. Additional power savings can be achieved by switching down to the lowest transmission power level for this route.

B. *Effect of Transmission Power Adaptation*

Table 3 shows a slightly different subset of the testing data. For the tests shown in this table, the transmission power is kept constant while, for each set of distances, different video

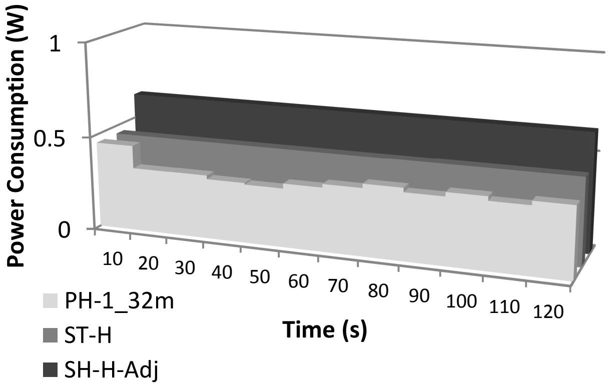

Figure 5 - Power v's Time - 1m - 32m Hop

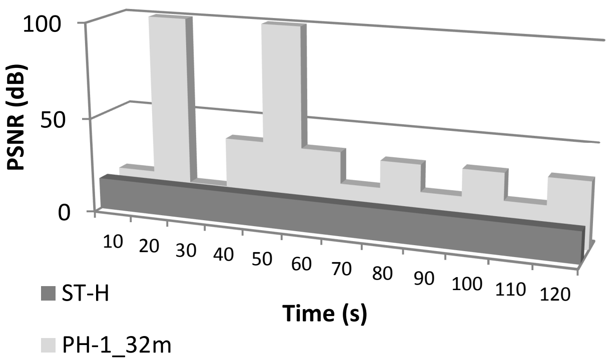

Figure 6 - PSNR v's Time - 1m - 32m Hop

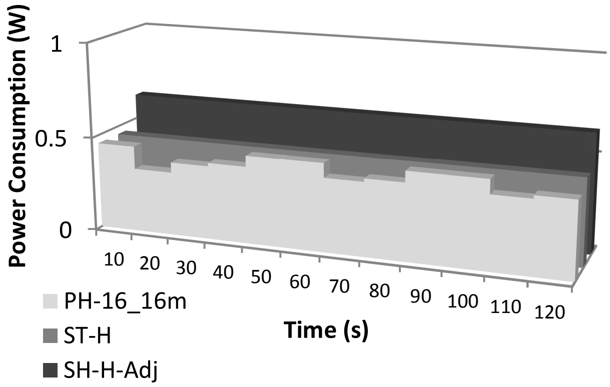

Figure 7 - Power v's Time - 16m - 16m Hop

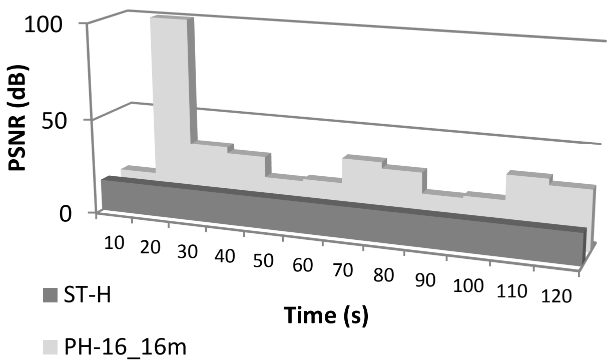

Figure 8 - PSNR v's Time - 16m - 16m Hop

quality levels, i.e. bit-rates, are exploited for reducing the power consumption during video transmission. The pattern here is clear, for each transmission path, the lower the number of bits being sent across the network, the lower the power consumption on both the server and relay device. The last result in the table bucks this trend slightly, but this is an anomaly most likely caused by external interference on the network. Interestingly though, this spurious result also indicates how the network behaves when the network is suffering from high levels of loss due to congestion or interference. In this situation, the low quality video can still make it through the network with an acceptable PSNR level. A high quality video stream would only compound the network issues in this situation and because of the loss and required retransmissions, would have a significantly lower PSNR level upon delivery, while also consuming more power on the server and relay devices. This highlights how crucial it is to consider the bit-rate of the communication as well as the device transmission power and the communication route, when targeting system-wide power savings.

In Table 3, the most efficient option in terms of power consumption is to use the first of the two-hop routes with the lowest quality video. This achieves a power saving of approximately 19%, against streaming the high quality video across the same link.

C. *Performance of PowerHop*

The final set of results presented here shows the performance of the PowerHop algorithm and can be seen in Figures 5 – 8. PowerHop was tested in two scenarios. For both scenarios, a video stream is sent wirelessly over a 32m distance for 2 minutes. Every 10 seconds the PowerHop algorithm repeats to decide whether or not to change the streaming parameters.

In the first scenario, the algorithm has a choice of using the direct path or following a multi-hop route where it transmits the data to a relay node that is 1m away. This relay node then sends the data the remainder of the distance. Figure 5 plots the power consumption over time of the system using PowerHop. Additionally, the power consumption of a static stream of the high quality video over the direct path is shown in the background. This consumption rate is shown for when there are only two nodes on the network and also in an adjusted format to include the overhead of other nodes on the network, as described above. In the graph we can see that the power consumption is decreased by the operation of the PowerHop algorithm. A power savings of 20% is achieved using the PowerHop algorithm or 58% for the adjusted power consumption rate. Figure 6 shows the PSNR of the video over time for both the PowerHop system and the static stream. The

PowerHop algorithm achieves an increase in the PSNR of 138% over the static stream. There is an additional overhead involved in the multi-hop path due to an increase in the power consumption of the relay device. For this scenario, the average power consumption on the relay device was 0.55W. As a result, the power consumption of the whole network increases by approximately 50%, but power can be saved on the server device.

In the second scenario, the distances are slightly different. The relay is exactly half way between the sender and receiver in this case. The rest of the parameters remain the same from the first scenario. Figure 7 plots the power consumption of the server device over time. As noted before, a clear reduction in the average power consumption is visible. The PowerHop algorithm saves 8% power on the server device over the static route. With the adjusted power consumption of the static route, this increases to 33% power savings. In addition to the power savings, the PSNR of the video stream is increased by 102% in comparison with the static stream case. A plot of the PSNR over time is illustrated in Figure 8. In this scenario, the relay device is used to achieve the savings on the server device. As a result the average power consumption of the relay device increases to 0.44W. The overall network power consumption increases by approximately 50% in this scenario, too.

VI. Conclusion

This paper proposes PowerHop as an algorithm for increasing the energy efficiency of mobile video transmission applications. These energy savings are achieved by dynamically configuring the transmission power of the device's WNIC, selecting whether or not to use a multi-hop communication route and adapting the quality of the video stream. Real world tests are performed on smart-phones to assess the effect of the algorithm. The results show that power savings of up to 20% can be achieved by using the PowerHop algorithm. In fact a 58% savings can be achieved if there are more than two devices in the network. For this power saving, the PSNR of the transmitted stream has also increased by 138% in comparison with the static delivery of the same video stream.

This paper demonstrates how power savings can be achieved for the video transmission device when using PowerHop. It is important to note that while using the PowerHop algorithm, although the power savings can be achieved on the video server device, the power consumption for the whole network increases. This occurs because additional burden is spread out onto other network devices.

Future work includes considering a higher number of hops in the tests, in order to investigate how the range of the network can be increased. Additionally, the PowerHop algorithm will be embedded into our existing platform, EASE [20]. Finally, going forward, it would be important to compare the PowerHop algorithm against other adaptive transmission algorithms in order to assess its benefit.

References

[1] Cisco Systems, Inc., 'Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2012–2017', White Paper, Feb. 2013.

[2] Eric Eason, 'Smartphone Battery Inadequacy', 17-Dec-2010. [Online]. Available: http://large.stanford.edu/courses/2010/ph240/eason1/. [Accessed: 07-Jan-2014].

[3] M. Kennedy, A. Ksentini, Y. Hadjadj-Aoul, and G.-M. Muntean, 'Adaptive Energy Optimization in Multimedia-Centric Wireless Devices: A Survey', *IEEE Communications Surveys & Tutorials*, vol. 15, no. 2, pp. 768–786, 2013.

[4] 'IP Camera Adapter'. [Online]. Available: http://ip-webcam.appspot.com/. [Accessed: 05-May-2014].

[5] 'Ustream — The leading HD streaming video platform'. [Online]. Available: http://www.ustream.tv/. [Accessed: 05-May-2014].

[6] 'Veetle - Live Video Stream on Veetle'. [Online]. Available: http://veetle.com/. [Accessed: 05-May-2014].

[7] 'Coming Soon: Smartphone Streaming, Powered by BitTorrent Live | The Official BitTorrent Blog'. [Online]. Available: http://blog.bittorrent.com/2014/02/14/coming-soon-smartphone-streaming-powered-by-bittorrent-live/. [Accessed: 05-May-2014].

[8] M. Kennedy, H. Venkataraman, and G.-M. Muntean, 'Dynamic Stream Control for Energy Efficient Video Streaming', *IEEE International Symposium on Broadband Multimedia Systems and Broadcasting* (BMSB), Nuremberg, Germany, 2011.

[9] R. Trestian, O. Ormond, and G.-M. Muntean, 'Energy-Quality-Cost Tradeoff in a Multimedia-Based Heterogeneous Wireless Network Environment', *IEEE Transactions on Broadcasting*, vol. 59, no. 2, pp. 340–357, Jun. 2013

[10] A. N. Moldovan and C. H. Muntean, 'Subjective Assessment of BitDetect – A Mechanism for Energy-Aware Adaptive Multimedia', *IEEE Transactions on Broadcasting*, Vol. 58, No. 3, pp. 480 - 492.

[11] X. Lu, E. Erkip, Y. Wang, and D. Goodman, 'Power efficient multimedia communication over wireless channels', *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 10, pp. 1738–1751, Dec. 2003.

[12] C. Bouras, S. Charalambides, K. Stamos, S. Stroumpis, and G. Zaoudis, 'Power Management Mechanism Exploiting Network and Video Information over Wireless Links', *Journal of Networks*, vol. 8, no. 1, Jan. 2013.

[13] M. T. Refaei, L. A. DaSilva, M. Eltoweissy, and T. Nadeem, 'Adaptation of Reputation Management Systems to Dynamic Network Conditions in Ad Hoc Networks', *IEEE Transactions on Computers*, vol. 59, no. 5, pp. 707–719, May 2010.

[14] B. B. Chen and M. C. Chan, 'Mobicent: a credit-based incentive system for disruption tolerant network', in *INFOCOM, 2010 Proceedings IEEE*, 2010, pp. 1–9.

[15] Z. Li and H. Shen, 'Game-Theoretic Analysis of Cooperation Incentive Strategies in Mobile Ad Hoc Networks', *IEEE Transactions on Mobile Computing*, vol. 11, no. 8, pp. 1287–1303, Aug. 2012.

[16] S. Banerjee and A. Misra, 'Minimum energy paths for reliable communication in multi-hop wireless networks', in *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing*, 2002, pp. 146–156.

[17] L. Kaddar, Y. Hadjadj-Aoul, and A. Mehaoua, 'EVAN: Energy-Aware SVC Video Streaming over Wireless Ad Hoc Networks', in *2011 IEEE 73rd Vehicular Technology Conference (VTC Spring)*, Budapest, Hungary, 2011, pp. 1–5.

[18] S.-B. Lee, G. Muntean, and A. F. Smeaton, 'Performance-aware replication of distributed pre-recorded IPTV content', *IEEE Transactions on Broadcasting*, vol. 55, no. 2, pp. 516–526, 2009.

[19] 'MANET Manager - Android Apps on Google Play'. [Online]. Available: https://play.google.com/store/apps/details?id=org.span. [Accessed: 07-Jan-2014].

[20] M. Kennedy, H. Venkataraman, G.-M. Muntean, 'Battery and Stream-Aware Adaptive Multimedia Delivery for Wireless Devices', IEEE International Workshop on Performance and Management of Wireless and Mobile Networks (P2MNET), *IEEE Local Computer Networks* (LCN), Denver, USA, October 2010, pp. 843 - 846.