International Journal of Engineering & Technology, 7 (3.2) (2018) 713-717

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Minimizing Energy Consumption on Mobile Phone by Rearranging Transport Protocol Load

S Suherman[1]*, Naemah Mubarakah[1], Marwan Al-Akaidi[2]

[1]*Electrical Engineering Department, Universitas Sumatera Utara, Medan, Indonesia*
[2]*Arab Open University, Kuwait*
**Corresponding author E-mail: suherman@usu.ac.id*

Abstract

There are two transport layer protocols that have been used in the internet protocol (IP) networks: Transmission Control Protocol (TCP) and User Datagram Protocol (UDP). Both protocols have been utilized for video streaming applications. This paper examines energy consumed by a mobile device when TCP or UDP employed by the application within it for streaming a video file. A transport protocol load management is proposed to reduce the mobile device energy consumptions. The experiments were conducted in the 802.11 environment. The results show that the proposed method is able to minimize mobile device energy consumptions up to 10.7% and 3.34% for both TCP and UDP protocols.

Keywords: *Mobile device energy consumption, transport load management, TCP, UDP*

1. Introduction

One of mobile communication development is size reduction towards miniaturization with compact, real-time, connected and interactive applications. One requirement to enable this is energy source technology. The smaller the size, the more limited the battery capacity. The more limited the energy source, the more important the energy efficiency [1], [2]. Various techniques have been proposed for minimizing energy consumption in mobile device, including energy-aware hardware optimization [3], middleware or by designing green application software [4].

The energy consumption reduction can be achieved in various layers of TCP/IP architecture [5]. Physical layer covers frequency selection, modulation techniques, multiplexing and channel coding. Data link layer ranges from framing, error control methods, to even security scheme. Network layer consists of addressing, routing methods, and mobility management. Transport layer uses reliable connection oriented or unreliable services. Application layer focuses on data representation or user interfacing.

One of research on physical (PHY) and data link/medium access (MAC) layers is performed by Han et.al [6] revealing that energy consumptions in mobile communication infrastructure are dominated by both base station and mobile station. As the number of mobile station users increases over 695 million in 2016 [7], the energy efficiency step in mobile devices is very urgent to perform. Tsao [8] in his survey on energy efficiency for MAC layer classified power saving solutions into MAC layer and cross layer. The first solution is also referred to as active mode including contention energy reduction, transmitting and receiving at associated speed, and protocol overhead reduction. Power save mode (PSM) includes sleeping during inactive period, beacon scheduling, PSM and non-PSM channel differentiation.

Research on network layer is dominated by routing protocol optimization, where routing overhead reduction and routing updates in finding the shortest path become the primary issues [9]. Transport layer protocol energy-efficient methods is achieved by using service differentiation between wired and wireless path as well as traffic differentiation of transmission control protocol (TCP), by using splitting [10], multipath methods [11], or scheduling [12].

Agrawal et al. [13] proposed a model of energy consumption TCP/IP traffic over 802.11 networks. The acknowledgement within TCP and MAC layers were considered as a waste of energy. Pang et al. [14] avoid duplicated acknowledgement by generating ACK on behalf of the mobile node in access point. Further, Tan et al. [3] proposed PSM-throttling by reshaping traffic into periodic bursts so that the channel is maximized and energy consumption is minimized.

Energy efficiency in application layer has been surveyed in [1] where hidden background process consumed energy the most. This hidden background process can relate to application process or network activities.

The multimedia traffics such as video are the main traffic on mobile networks replacing text messages. It urges energy efficiency on the installed application. Therefore, energy consumption on application or transport layers that are directly connect to installed application is very important.

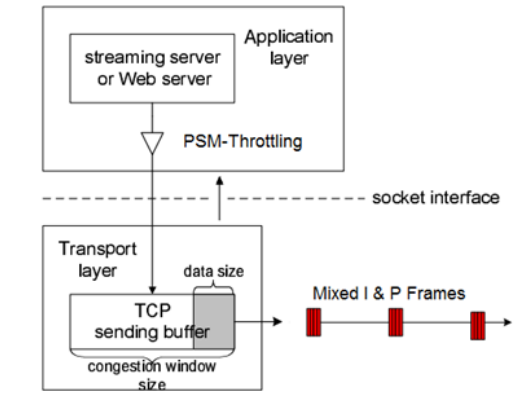
This paper compares energy consumed by a mobile node when transmitting video by using TCP and UDP. A transport layer load management to reduce energy consumption is proposed and compared to existing PSM throttling. The energy consumption measurement is evaluated by using real devices over an 802.11 network.

2. Proposed Model

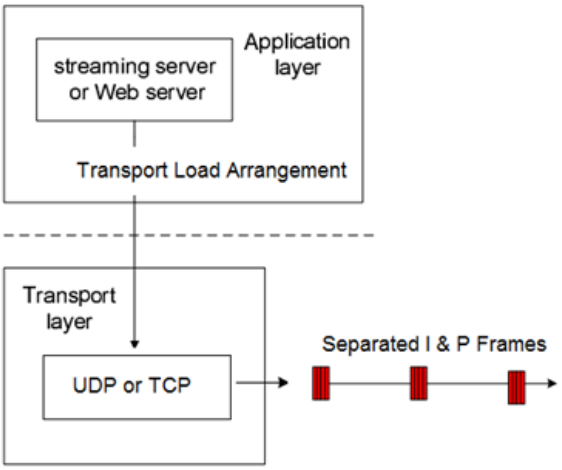
A video file is constructed by a sequence of images or frames. These frames are generated in term of frame per second (fps). Video compression technique is encoding these frames to mini-

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

mize the overall size. Frames are classified into several types: frame I, frame P and frame B. Frame I is the original image that has high capacity as each pixel is coded to bytes. Frame P and frame B are constructed based on image differences. Frame I can be decoded directly into an image, while frame P and frame B require an image (frame I) to be decoded. This shows that frame I is the most important frame compared to frame P and B. Frame P and B cannot be decoded correctly if the previous frame I is missing. Some MAC and transport layer techniques [15] protects frame I by prioritizing the frame I. Therefore, in the proposed transport load management, frame I bytes should not be separated; it should be transmitted at once in a single burst, so that the integrity of the received frame I is protected.



a. PSM-Throttling



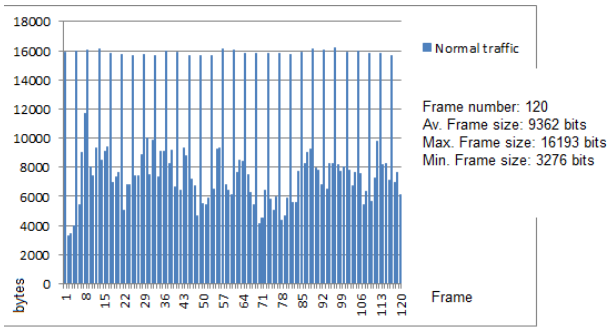
b. Transport Load Arrangement

Fig.1: Transport layer load management

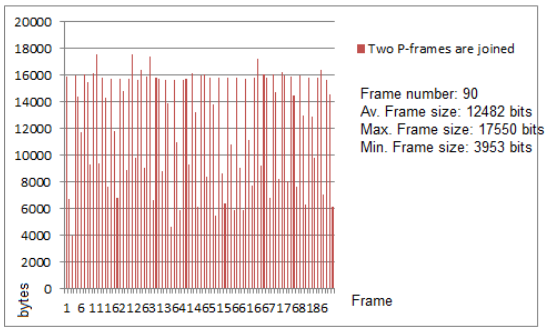
The proposed method applies traffic arrangement as in PSM-throttling [3]. However, instead of buffering data and dividing them into the same length of periodic burst, the proposed method transmits frame I as a whole, separated from P or B frames and joins the two or three consecutive P or B frames.

In PSM-Throttling, dividing traffics may cause frame I be chopped into several parts and send them separately. Frame I may be received imperfectly. The loss of one or two frame I part (s) results all frames be imperfectly decoded and the image quality be poor. Furthermore, since the P or B frame sizes are small enough, buffering multiple frames may result high packet delay. In addition, PSM-throttling applied only for TCP only.

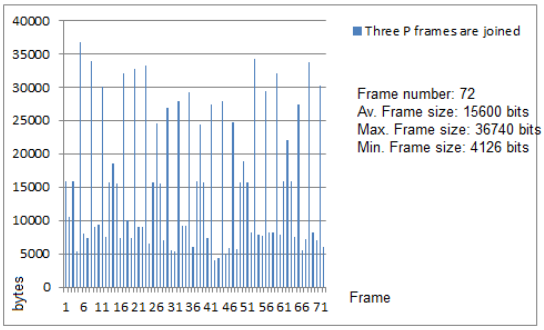
Figure 1 illustrates the proposed method. The main differences of the proposed method and the existing one are that the proposed method considers the frame types in sending video content. Further, it is applicable for both TCP and UDP.



(a)



(b)



(c)

Fig.2: Transport layer load management

Figure 2a shows the trace of a video traffic taken from akiyo_cif.yuv [16], Figure 2b and 2c are the re-shaping traffics by using the proposed method in two and three joined P frames. The original traffic contains 120 frames with average size 9362 bits. By assuming that the network bandwidth is enough, this traffic may produce 120 time transmissions. When the proposed method is applied, two or three frames are concatenated; the numbers of transmissions reduce to 90 and 72 times with denser traffics of 12482 and 15600 bits in averages. By using the existing PSM-throttling method, traffic from frame I, P and B may experience separation, producing various bit level with certain number of frames depending on transmission rate.

3. Research Method

In order to evaluate the effectiveness of the method, streamer j2me software is build and an experimental apparatus was developed by using a Symbian operating system based mobile phone (a Nokia E6x series) as the sender, a laptop with a java based downloader as the receiver. An arduino microcontroller circuit is devised to measure voltage and current level on the mobile phone battery whis is BL-4PC series. The adhoc 802.11 network is used as the wireless link between the mobile phone and the laptop. The apparatus is shown in Figure 3.

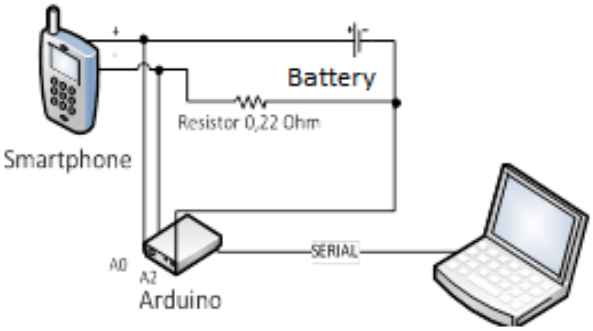


Fig.3: The measurement setup

The socket programming tool, Java Mobile Edition (J2ME) with wireless toolkits is employed to create TCP and UDP client and server programs. The program with an embedded video source is installed in Nokia E63 phone. The server java based software is programmed and installed in the laptop as the video receiver. The arduino is used to read voltages across the battery and the resistor 0.22 Ohm, to calculate the electrical energy consumed by the mobile phone. The flow current through the mobile phone is measured by the help of 0.22 Ohm resistor. The scratch program is made and installed in the arduino. The energy calculation result is sent to laptop to monitor the energy level. The time of energy recording and packet reception is synchronized.

The proposed method is compared to the existing PSM-Throttling. The bandwidth throttling detection within the compared method is replaced by the predicted optimum bandwidth as 902.11 fully controlled without adjacent network.

Initially, the experiments explore the impact of TCP window to energy consumption in normal traffic, followed by the impact of the proposed method in TCP and UDP energy consumption.

No additional transport layer method is employed during TCP and UDP characterizations. The standard TCP and UDP were employed.

4. Experiment Results

4.1 TCP Window versus Energy Consumption

In order to determine what TCP window size is effective to stream video packets over mobile 802.11, the j2me programming is explored. However, the socket programming in j2me does not bind to fixed window size; instead, the TCP flow control that decides what optimum TCP window size the streaming needs. So the energy usage and TCP window relation is extracted from the overall stream. Figure 4 shows the relation between window size and energy consumption.

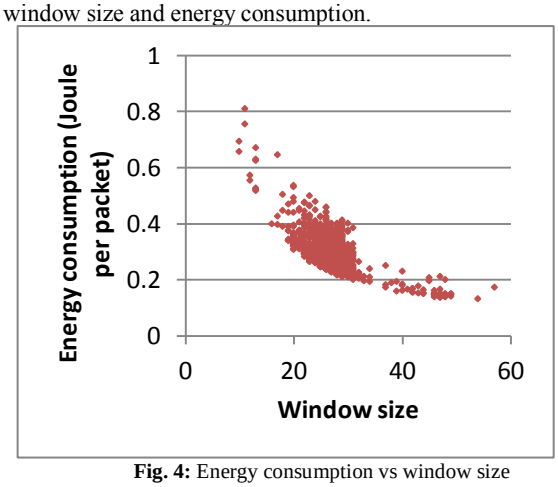


Fig. 4: Energy consumption vs window size

The data is collected from a number of times experiment and the results are classified based on its window size. The scattered energy consumption data over various window sizes show a clear pattern. The experiments found that the mobile phone energy consumptions when transmitting video using TCP are influenced by the size of TCP window. The higher the window size, the more packets are sent, the more efficient the energy usage. The energy consumption is represented by energy consumption per packet (Joule per packet).

Energy consumption decreases as window size increases. The average energy absorbed by the mobile device reaches up to 0.8 Joule per packet for TCP window smaller than 20. The energy consumption decreases to about 0.25-0.5 Joule per packet when TCP window is in between 20 to 30. Energy decrement continuous up to only 0.13 Joule per packet foR window size higher than 30. The energy consumption on mobile phone decreases exponentially to window size. Window size of 20-30 dominates the streaming.

4.2 Load Arrangement Impact on TCP Energy Consumption

Window size 20 to 30 packets dominate TCP packet transmission without any method. By using the proposed method with traffic arrangement of 2 joined P frames shifts window size to higher values to area with lower energy consumption as depicted in Figure 5. The window sizes of 22 to 37 dominate the transmission. The average TCP energy consumption by using the proposed method is 0.2544 Joule per packet. It is about 10.7% lower than without traffic arrangement.

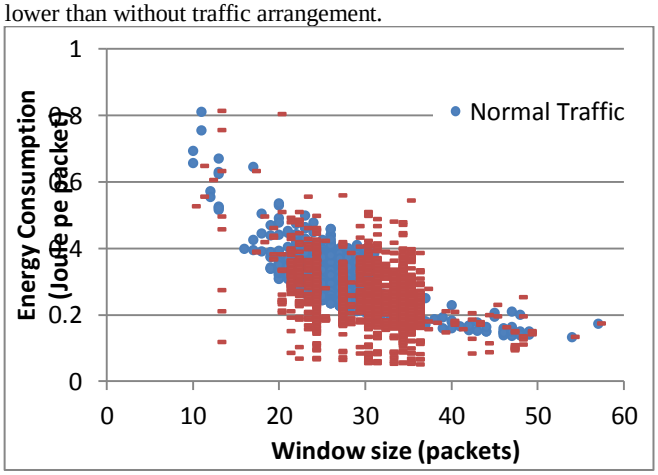


Fig. 5: TCP Energy consumption comparison

However, packet delay increases from 82.6 ms to 88.8 ms. The proposed method causes about in 7.51% average delay increment. However, delay is still acceptable for real-time video transmission.

Other performance parameters are shown in Table 1. The proposed method causes packet loss percentage increasing from 0.17% to 0.21%. However, image quality (peak signal to noise ratio, PSNR) can be kept as close as the previous condition. The normal traffic and the proposed method have almost similar average PSNR value, 34.6 dB and 34.2 dB. These facts show an initial proof that the proposed method is able to reduce energy consumption in mobile device without experiencing significant delay increment and packet loss addition.

Table.1: Performance comparison of traffic load arrangement

Method Parameter (unit)	Normal Traffic	Proposed Method		PSM-Throttling Bulk Size (bytes)		
		2 Joined P frames	3 Joined P Frames	10000	13000	16500
Energy consumption (Joule/packet)	0.285	0.2544	0.2123	0.2019	0.1857	0.1567
Average Delay (ms)	82.6	88.8	93.5	212.5	254.2	288.2

Packet Loss (%)	0.17	0.21	0.25	0.42	0.67	1.25
PSNR (dB)	34.6	34.2	32.5	25.2	24.7	32.7

PSM-Throttling on the other hand, is able to reduce lower the proposed method. PSM-throttling results energy consumption at least 0.1567 Joule per packet. However, packet delay increases tremendously, more than 200 ms, which is not acceptable for real-time multimedia application. The high delay occurs as traffic re-shaping delaying the received frame from application layer, especially small size P frames. This delay is a must as system should fulfil the expected bulk size. For instance, Frame P with size 4000 bytes experience additional delay at least (1/fps)* ceiling of (bulk_size/4000). For frame rate 25 fps and bulk size 10000 bytes inject at least 120 ms additional delay.
Furthermore, frame I defragmentation is happened when the bulk size is smaller than the frame I size. As a result, PSNR decrease. The lowest PSNR is 24.7 dB. Generally, PSM-Throttling causes high average delay, which is bad for real time video. PSM-throttling reduce energy significantly but worsen delay.

4.3 UDP Datagram Size Impact on Energy Consumption

The UDP experiment is conducted by examining various datagram size separately. As a result, each datagram size results various total consumed energy in delivering the video traffic. Therefore, the energy consumption is measured as the total energy required for sending video traffic from the beginning to the end, regardless the packet loss and delay.
Compared to TCP energy consumption, UDP absorbs more energy than TCP. This is caused as UDP stream is uncontrollable that the flow causes high collision in medium access. Figure 6 shows the total energy pattern of UDP for increasing datagram size.

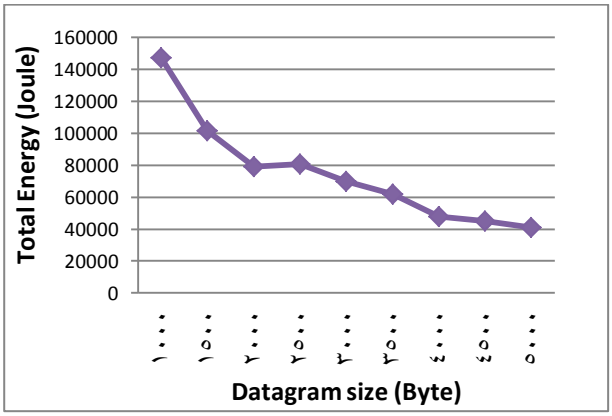


Fig. 6: Energy consumption vs packet size

As in TCP, Energy consumption is reduced when datagram size is enlarged. Both decrements are exponential. In UDP, only datagram size of 2500 shows inconsistent result. Small inconsistency could be influenced by wireless channel variations, battery conditions, internal phone process, as well as measurement faults. However, as the inconsistency is insignificant. Generally, energy consumption on mobile device for video streaming by using UDP decreases exponentially to datagram size.

4.4 Traffic Arrangement Impact on UDP Energy Consumption

Since PSM-Throttling is to reduce TCP energy consumption, the method is not suitable for UDP-based application. So data for PSM-Trothing is not provided. On the other hand, the proposed method by using traffic arrangement is applicable for UDP. Therefore, the experiment involves only traffic arrangement with the proposed method.

UDP datagram transmission with non-arranged traffic consumes the total energy about 75000 Joule. The proposed method with 2 joined P frames successfully reduces energy consumption about 3.34% lower than without traffic arrangement (Figure 7).
The method causes delay increases in average 11%, from 82.9 ms to 92.1 ms. However, this delay is acceptable for real-time video transmission. The proposed method makes UDP suffers packet loss of 6% higher than the original traffic. UDP experience about 35.2% in average. Both delay and packet loss characteristics are not shown.
Although delay and loss are slightly increase, the proposed method is successfully reducing energy consumption in mobile device.

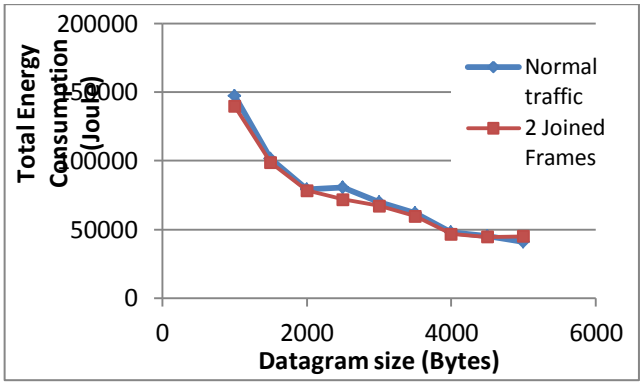


Fig. 7: Energy consumption vs window size

5. Conclusion

The work has examined energy consumption in a mobile device when transmitting video experimentally. By using a single 802.11 wireless link, mobile device connection has lower energy consumption if the transport layer protocol (TCP and UDP) has high window or datagram size. TCP energy consumption is about 0.8 Joule per packet for window size less than 20, consuming 0.25 to 0.5 Joule per packet for 20<window<30, and absorbing 0.13 Joule per packet for window>30. Meanwhile, UDP consumes total 150000 Joule for datagram size 1000 bytes, decreases to 40000 Joule for datagram size 5000 bytes.
The proposed method that arranges traffic is successful in reducing energy consumption for both TCP and UDP. In TCP, the proposed method shifts windows to a bigger size and reduces 10.7% energy consumption. Likewise, the method also successfully suppresses total energy consumption in UDP about 3.34%. However, the method adds delay and packet loss. Future work may concern with this matter.

References

[1] C. Wilke, "Energy Consumption and Efficiency in Mobile Applications: A User Feedback Study," in *Proc. IEEE Conf. on Green Computing and Communications (GreenCom)*, 2013, pp. 134–141.
[2] D. Subramaniam *et al.*, "A Stacked Planar Antenna with Switchable Small Grid Pixel Structure for Directive High Beam Steering Broadside Radiation," *Int. J. Eng. Technol.*, vol. 7, no. 2.5, pp. 122–127, Mar. 2018.
[3] S. H. W. Jin, A. Lee, SC. Tan, "Low-Power Multichannel Wireless Transmitter," *IEEE Trans. Power Electron.*, vol. 33, no. 6, p. 5016–5028., 2018.
[4] Z. Gai, K., Qiu, M., Zhao, H., Tao, L., & Zong, "Dynamic energy-aware cloudlet-based mobile cloud computing model for green computing," *J. Netw. Comput. Appl.*, vol. 59, p. 46–54., 2016.
[5] J. E. Christine, "A Survey of Energy Efficient Network Protocols for Wireless," *Wirel. Networks*, vol. 7, pp. 343–358, 2001.
[6] and S. A. C. Han, T. Harrold, "Green Radio: Radio Techniques to Enable Energy-Efficient Wireless Networks," *Energy Effic. Commun.*, vol. May, 2011.
[7] W. M. Cheung, M. F., & To, "The influence of the propensity to

trust on mobile users' attitudes toward in-app advertisements: An extension of the theory of planned behavior," *Comput. Human Behav.*, vol. 76, p. 102–111., 2017.

[8] and C.-H. H. Tsao, Shiao-Li, "A survey of energy efficient MAC protocols for IEEE 802.11 WLAN," *Comput. Commun.*, vol. 34, no. 1, p. 54–67., 2011.

[9] J. Z. and X. Wang, "Model and Protocol for Energy Efficient Routing over Mobile Ad Hoc Networks," *IEEE*, 2007.

[10] T. Khalifa, "Split- and Aggregated-Transmission Control Protocol (SA-TCP) for Smart Power Grid," *IEEE Trans. Smart Grid*, vol. 5, no. 1, pp. 381 – 391.

[11] S. Chen, "An energy-aware multipath-TCP-based content delivery scheme in heterogeneous wireless networks," in *EEE Conf. on Wireless Communications and Networking (WCNC)*, 2013, pp. 1291 – 1296.

[12] N. Mubarakah, "Reducing mobile device energy consumption in transmitting multimedia content by arranging transport protocol load," in *Technology of Information and Communication (ISemantic), International Seminar on Application for*, 2016.

[13] V. P. P. Agrawal, A. Kumar, J. Kuri, M.K. Panda, V. Navda, R. Ramjee, "Analytical models for energy consumption in infrastructure WLAN STAs carrying TCP traffic," in *International Conference on Communication Systems and Networks (COMSNETS)*, 2010.

[14] V. C. M. L. Qixiang Pang, S.C. Liew, "Performance improvement of 802.11 wireless network with TCP ACK agent and auto-zoom backoff algorithm," in *IEEE Vehicular Technology Conference*, 2005.

[15] M. Fakhrizal, S. R., & Al-Akaidi, "A subjective scheduler for subjective dedicated networks," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 237, no. 1, p. 012019, 2017.

[16] -, "Traffic sample." [Online]. Available: http://media.xiph.org. [Accessed: 16-Mar-2016].