Device Characteristics-based Differentiated Energy-efficient Adaptive Solution for Video Delivery over Heterogeneous Wireless Networks

Ruiqi Ding and Gabriel-Miro Muntean
Performance Engineering Laboratory, Network Innovations Centre, The Rince Institute, Dublin City University, Ireland
Email: ruiqi.ding2@mail.dcu.ie, munteang@eeng.dcu.ie

***Abstract*—The limited battery capacity of current mobile devices and increasing amount of rich media content delivered over wireless networks have driven the latest research on energy efficient content delivery over wireless networks. Many energy-aware research solutions have been proposed involving traffic shaping, content adaptation, content sharing, etc. The existing solutions focus on the delivery application without considering application running environment and device features that pose different energy constraints on the whole content delivery process. This paper presents *a Device characteristics-based differentiated Energy-efficient Adaptive Solution* (DEAS) for video delivery over heterogeneous wireless networks. DEAS constructs an energy-oriented system profile including power signatures of various device components for each running application. Based on this profile, an energy efficient content delivery adaptation is performed for the current application. The proposed solution is evaluated by simulation-based testing and compared with other state of the art approaches in terms of performance and energy efficiency. The results show how DEAS outperforms the other well-known solutions.**

I. Introduction

Mostly due to the increased number and use of smartphones, the number of mobile network-connected devices will exceed the world's population in 2012; handsets will generate in excess of 50 percent of mobile data traffic in 2014 [1]. Additionally the popularity of network delivered rich media content among users will determine that for example video content to account for more than two thirds of the globe networking traffic by 2016, according to Cisco [1]. The existing various technology-based wireless networks and hot spots provided by businesses, public institutions, etc. establish a perfect heterogeneous wireless network environment which can provide ubiquitous connectivity. Such an environment is illustrated in Fig. 1.

Since mobile device owners are most of the time on the move and use them anywhere and anytime, energy efficiency is of paramount importance when delivering data to mobile devices in general, and in particular rich media content. Moreover, the content providers provide content at different quality levels, and users always prefer higher quality content, which requires more energy to transmit and process. Consequently there is a trade off between quality and energy levels.

The authors would like to thank Irish Research Council and Citadel 100 Ltd for their support.

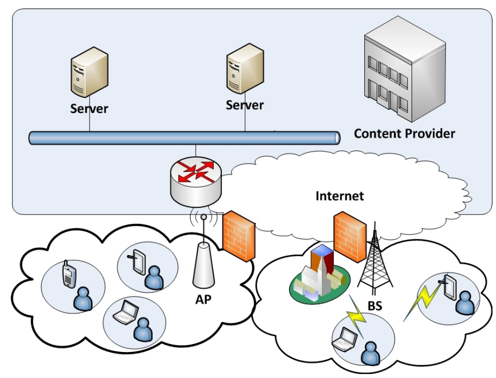

Fig. 1. Multimedia Content Delivery over Heterogeneous Wireless Networks

Existing mobile devices are highly heterogeneous in terms of hardware features and user settings. Given this fact, the optimal characteristics of the content can vary significantly between any two devices, between two applications on the same device, and even between two different time periods for the same application on the same device. This suggests that there is a need for an algorithm based on end system profiling to help decide what content characteristics are desirable when considering both energy efficiency and QoS.

Existing research effort on multimedia content exchange focuses mostly on supporting high quality of service (QoS) levels. However, even for energy-efficient content delivery solutions, the heterogeneity of mobile devices is not considered. In this context, a **Device characteristics-based differentiated Energy-efficient Adaptive Solution (DEAS)** for video delivery over heterogeneous wireless networks is proposed in this paper. DEAS constructs an energy-oriented system profile including power signatures of various device components for each running application. Based on this profile, an energy efficient content delivery adaptation is performed for the current application.

Section II presents related works on energy efficient content delivery solutions. Section III describes DEAS architecture and principle, including energy-oriented system profiling, and energy efficient content delivery adaptation mechanism. Final sections present simulation testing results showing how DEAS outperforms other solutions, conclusions and future works.

II. Related Works

Since the improvements in battery capacity cannot keep up with the pace of the energy consumption demands of mobile devices, energy is a vital resource. Faster, yet more energy demanding hardware and increasingly complex applications make the energy issue even more important. Wireless multimedia content delivery is a typical example, as large data transmission and decoding processing are both very energy consuming tasks that involve the wireless network interface card and CPU [2]. Consequently, many research efforts have been put in devising energy efficient solutions for content delivery over wireless networks.

The network interface card of mobile devices consumes significant energy when transmitting. Therefore placing the interface card in low power or sleep modes as much as possible saves energy. To maintain high QoS level and energy efficiency, STELA [3] estimates future traffic pattern based on the historic data traffic arrival patterns, and accordingly adjusts the sleeping window of the mobile devices' wireless transceiver. The server often shapes the traffic strategically so that the client can sleep longer when it can, and receive data efficiently once it wakes up. Yan et al. [4] introduces a client-centred TCP compatible schema that tracks each TCP connection to determine the timing to transit wireless network interface card into sleep mode. It also shapes the traffic from the client side by requesting the server to send data bursts in order to prolong sleeping intervals. This work is limited to TCP-based applications, including web browsing and FTP downloads.

Buffering the data in the AP can also increase the device sleep time. Adams et al. [5] buffer the data traffic at AP in order to hide the traffic when the client is in sleep mode. Thus the frequency of client wake ups is reduced, conserving energy at client devices.

Reducing the amount of traffic to be transmitted is another way of conserving energy. Content sharing is a straightforward method to reduce the traffic from the server to the clients. GroupDL [6] allows the server to transmit only one content copy to a group of users requiring the same content and a part of the original content to each group member only. The content will be then shared with other group members. The server reduces the energy consumption to only one fraction of the group size. However, content is shared via P2P via ad-hoc wireless networks, which is not always feasible. Besides, GroupDL cannot guarantee any energy reduction on client side. In fact, the total energy consumption on client side is similar with that of traditional solutions. Chen et al. [7] introduce a schema where a device can request other device with more residual energy to download the content for it, then transfer it locally. It assumes the content is downloaded via a 3G interface and local transmission is performed via a WiFi interface. Tests performed on an iPad showed how energy was saved because less 3G interface usage was required for the device with less residual energy.

Quality adaptation during multimedia content delivery is also an effective approach for energy saving. SAMMy [8] is a dynamic video delivery solution that adjusts content quality based on estimated signal strength and monitored packet loss rate. These parameters are utilised to make more efficient use of the wireless network resources, increase user perceived quality and save energy. BaSe-AMy [9] monitors the device residual energy, packet loss rate and remaining video duration and performs accordingly energy-aware video quality adjustments.

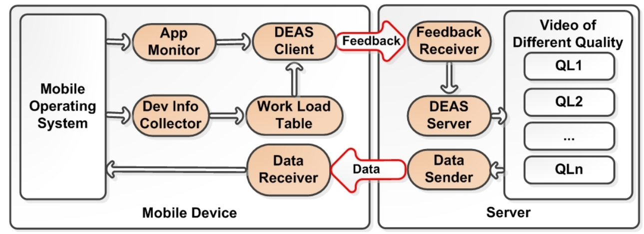

Fig. 2. Block-level architecture of DEAS

The above solutions consider devices in the same context, and use metrics related to content delivery QoS, network conditions and energy level to adjust the delivery strategy. This assumption obviously does not perfectly suit current wireless communications scenarios, especially smart devices dominated heterogeneous wireless networks. Along with the multimedia delivery application, multiple other applications could be running on the mobile device at the same time. A device performing streaming only is certainly less energy constraint than when it is running multiple applications. Different devices can also demand different amounts of energy to perform streaming at the same quality. Therefore a smart content delivery strategy can improve energy efficiency by differentiating individual devices in different application load scenarios, and by adjusting the quality level of the multimedia delivery accordingly.

III. Device characteristics-based differentiated Energy-efficient Adaptive Solution - DEAS

A. Architecture

The architecture of the proposed DEAS, the device-differentiated adaptive video delivery solution is presented in this section. Fig.2 illustrates the major components of DEAS and the information sharing mechanism among them for energy efficiency in heterogeneous network environments.

In Fig. 2 *App Monitor* identifies the current running application each time a new application is launched. *Dev Info Collector* is responsible for fetching the status of the device from the *Mobile Operating System* by taking samples of multiple readings, including work load on CPU, load of wireless network card, cellular interface, resolution of the display unit, battery characteristics, battery energy level. Based on the above information, *DEAS Client* stores device features, maintains application profiles, evaluates the delivery quality of service, calculates "expected battery life", and makes video quality level adjustment requests according to the DEAS's video quality adaptation algorithm. On the server side, *Feedback Receiver* listens for adaptation requests from the mobile device, and *DEAS Server* handles the requests by dispatching video at certain quality level (QL). Data packets are transmitted by the server's *Data Sender* to the *Data Receiver* at the mobile device.

B. Background and Contributions

Data transmission and decoding are two most energy consuming tasks of video delivery. Video applications can easily reduce energy consumption by handling lower bit rate video sequences that require less data transmission and decoding effort. This is because the energy budget for video delivery at different quality levels vary significantly [10]. Therefore adjusting content to lower quality levels is necessary when energy constraints become tight. Very high quality video content is not even necessary for some small hand-held devices, nor appreciated by certain user groups. Importantly, current mobile operating systems support multi-tasking, allowing multiple applications run on the machine simultaneously. Apart from the currently active video delivery application, the energy consumption of the other applications on hold has to be considered. A typical use case illustrating this is when one uses a browser to send a post on Facebook, and then starts a Skype multimedia session, putting the browser on the hold, without quitting. In energy constraints situations, content adaptation during the multimedia session may be needed.

DEAS maintains an energy-oriented system profile in the device, and passively monitors all running applications. It also monitors the active video application in terms of received data packets and calculates "battery expected lifetime" and "perceived quality", as estimated using PSNR. The data collected in the system profile is used by an energy-efficient content adaptation algorithm to request the server to perform adjustments to the content delivery. This distributed approach reduces the computation load at the server. The overhead introduced by system profiling can be small by implementing the online monitoring system carefully [11].

The major contributions of DEAS are novel algorithms for:

- energy-oriented system profiling and
- energy-efficient quality-oriented content adaptation.

C. Energy-oriented System Profiling

The Energy-oriented System Profiling is introduced by DEAS to characterize both device component and application energy consumption and is performed by DEAS in three phases: setup, monitoring and update. DEAS focuses on those components which are the major energy consumers among the hardware components of the latest mobile devices [12]. They are the screen (SCR), graphics processor (GRA), WLAN interface card (WLAN), such as WiFi for example, cellular network interface module (CELL), such as GSM for instance and processing chip set (CPU) only.

The Energy-oriented System Profiling requires the construction of two tables: the *Component Workload Profile Table* which maintains the relationship between workload on the major device hardware components and their corresponding power consumptions and the *Application Profile Table* that relates the workload on the hardware components and applications. The principle of system profiling was first introduced in the application-aware energy model of the AWERA routing solution [13][14]. This paper develops and improves the profiling techniques with multi-task support and incremental update as described in the following subsections.

TABLE I
COMPONENT WORKLOAD PROFILE TABLE

Workload	P_{CPU}	P_{GRA}	P_{CELL}	P_{WLAN}	P_{SCR}
5%	$P_{CPU}5$	$P_{GRA}5$	$P_{CELL}5$	$P_{WLAN}5$	$P_{SCR}5$
20%	$P_{CPU}20$	$P_{GRA}20$	$P_{CELL}20$	$P_{WLAN}20$	$P_{SCR}20$
...	...	...	...	...	...
100%	$P_{CPU}100$	$P_{GRA}100$	$P_{CELL}100$	$P_{WLAN}100$	$P_{SCR}100$

1) Setup Phase: The setup phase constructs the *Component Workload Profile Table* to make DEAS device independent. Due to the difference among devices of different specifications, even running the same application results in different loads on different device components. Besides, the same workload results in different battery depletion on different devices.

In this phase, DEAS runs a set of predefined tasks to put various loads on the different device components and monitors the corresponding power consumption. For example, DEAS runs a dedicated floating point addition program to load CPU to different degrees. Perrucci et al. [12] applied a similar approach to measure energy consumption of each hardware component of a smart phone with external circuitry. DEAS takes readings from the operating system directly and with better granularity. Although this phase introduces overhead, it is applied to each individual device once only.

The *Component Workload Profile Table* is shown in Table I, where $P_{CPU}x$, $P_{GRA}x$, $P_{CELL}x$, $P_{WLAN}x$ and $P_{SCR}x$ (x = 5, 20, 40, 60, 80, 100) are power consumptions of the indicated device component at the workload of x percent for CPU, graphics processor, cellular module, WLAN interface and screen, respectively.

DEAS measures a small set of values to reduce the overhead. In Table I, the power level between the values measured is determined by linear interpolation as in (1). This approach trades off some accuracy for the amount of resources required. In (1), $P_{comp}(WL_{comp}(x))$ represents the power of the component at workload of x% and $WL_{comp}(x)$ represents the observed workload value x on the hardware component. The component *comp* can be in turn CPU, GRA, CELL, WLAN, or SCR.

$$\frac{P_{comp}(WL_{comp}(x)) - P_{comp}(WL_{comp}(a))}{WL_{comp}(x) - WL_{comp}(a)} = \frac{P_{comp}(WL_{comp}(b)) - P_{comp}(WL_{comp}(a))}{WL_{comp}(b) - WL_{comp}(a)} \quad (1)$$

2) Monitoring Phase: This phase constructs the *Application Profile Table*. Once a new application with no previous record is launched, DEAS records the extra workload on the mentioned hardware components on top of the existing figure caused by the applications on hold. Once the application shuts down, the average value is calculated and recorded as a new entry in the *Application Profile Table*. Adaptive applications exchanging significantly different data rates are associated multiple records, so that the energy constraints imposed by different data rates are calculated separately. Each application is assigned a grade of its power consumption share of the device. The whole device is assigned a grade of its power consumption as well. The following paragraph explains how the grades are calculated based on the application profile.

In equation (2), G^k_{app} represents the utility function corresponding to the energy constraints imposed by the application *k* on all the major device hardware components considered: CPU, screen, graphics, WLAN card and cellular module, respectively. $G^k_{comp_i}$ represents the utility grade corresponding to the energy constraint imposed by the application *k* on the *i*-th device component. Normalized weights are used to balance the contribution of different hardware components on the overall utility function. Weight values W_{comp_i} are obtained by dividing the maximum power consumption of each of the components to the maximum system power consumption, as indicated in equation (3), where $MaxP_{comp_i}$ represents the maximum power of the hardware component *i*. For each application *k*, $G^k_{comp_i}$ of each individual component is obtained from the ratio between the power used by the application *k* ($P^k_{comp_i}$) and the maximum power of that component ($MaxP_{comp_i}$), as described by equation (4). Importantly, $P^k_{comp_i}$ is obtained according to the observed workload of application *k* ($WL^k_{comp_i}$) with reference to the *Component Workload Profile Table* and equation (1).

$$G^k_{app} = \frac{\sum_{i=1}^{n}(W_{comp_i} \cdot G^k_{comp_i})}{\sum_{i=1}^{n} W_{comp_i}} \quad (2)$$

$$W_{comp_i} = \frac{MaxP_{comp_i}}{\sum_{i=1}^{n} MaxP_{comp_i}} \quad (3)$$

$$G^k_{comp_i} = \frac{P^k_{comp_i}}{MaxP_{comp_i}} \quad (4)$$

In equation (5), the grade of the system load G_{sys} is given by the sum of the grade values of all the applications G^k_{app}.

$$G_{sys} = \sum_{k=1}^{n}(G^k_{app}) \quad (5)$$

3) Update Phase: This phase incrementally updates the *Application Profile Table*. DEAS regularly samples the average power on each component for applications which already have records in the *Application Profile Table*. According to equation (6), the updated workload WL^k_{update} for an application *k* is calculated from the old value WL^k_{old} taken from the application profile entry and the new value WL^k_{new} which has just been measured. The weight values W^k_{old} and W^k_{new}, which distribute the effect of the two workload values, are proportional with the duration the application was running.

$$WL^k_{update} = W^k_{old} \cdot WL^k_{old} + W^k_{new} \cdot WL^k_{new} \quad (6)$$

$$\frac{W^k_{old}}{W^k_{new}} = \frac{Runtime^k_{old}}{Runtime^k_{new}}$$

D. Energy-efficient Quality Adaptation Algorithm

The Energy-efficient Quality Adaptation Algorithm requests the server to perform adjustments to the content delivery according to two metrics: ”battery expected lifetime”, calculated based on the data collected from the energy-oriented system profile and ”perceived quality”, estimated using PSNR of the monitored active video application.

As revealed by extensive tests [10], the variation of traffic volume and decoding effort caused by different video at quality levels has a major impact on system energy efficiency when compared with that of link quality, network load and transport protocol. Consequently the proposed Energy-efficient Quality Adaptation Algorithm focuses on the above two metrics. The network load and link quality are also important factors, but this paper assumes homogeneity in network aspects. However, detailed discussion of network related parameters will be fully investigated in a future publication.

It is assumed that video content encoded at different rates and consequently at different quality levels (QL) is already stored at the server, or the server has means to change the video encoding rate on the fly. The multimedia data is transmitted via RTP, and any DEAS enabled device transmits feedback via RTCP to the server. Based on the feedback, the Energy-efficient Quality Adaptation Algorithm involves dynamic switching of the video content quality source and video delivery of an energy-aware adapted stream. This mechanism provides a user-centric energy-aware solution to address the challenges introduced by the heterogeneity of mobile devices in terms of software and hardware.

Expected battery lifetime is used to determine the maximum quality level upon which the adaptation is performed. Equation (7) presents the utility function to calculate the ”expected battery life” based on the currently running applications and individual device features. α_B is the compensation factor that reflects the depletion curve of the battery *B*. $Residual_B$ is the current residual energy of the battery, $Voltage_B$ is the voltage value of the battery used. They are both obtained from the mobile operating system. P_{sys} is the full load power of the system obtained in Setup phase. G_{sys} reflects the current energy constraints imposed by both the active application and applications on hold.

$$Exp\_Life = \frac{Residual_B * Voltage_B}{P_{sys} \cdot G_{sys} \cdot \alpha_B} \quad (7)$$

PSNR is used to estimate user perceived quality in DEAS. Based on PSNR, DEAS client makes adaptation requests to best balance quality of experience and energy saving. Equation (8) estimates PSNR [15], where *MAX_BitRate* is the maximum bit rate of the video, *Exp_Thru* is the expected throughput, and

Thru is the actual throughput. When the PSNR value is below certain threshold, video quality level degradation is needed.

$$PSNR = 20 \cdot \log_{10} \left(\frac{MAX\_BitRate}{\sqrt{(Exp\_Thru - Thru)^2}} \right) \quad (8)$$

DEAS performs the energy-efficient video quality adaptation as described in Algorithm 1. Since the mobile device receives the first data packet, it starts sending feedback regularly to the server. The proposed adaptation algorithm first calculates the expected battery life *Exp_Life*, RTT and PSNR for the current session. *Exp_Life* is compared with the threshold *Thre_Life*, indicating the desired session duration (e.g. video clip time length). If *Exp_Life* is shorter than the threshold *Thre_Life*, the maximum quality level *Max_Level* is decreased to a lower level if any. Otherwise, the maximum quality level is set to a higher level if any. Adaptation is caused by any energy-relevant event, for example, starting or ending an application, battery depletion in time, etc. If the computed PSNR value is less than the threshold *Thre_PSNR*, the device will also request a video quality level decrease via feedback. Otherwise, it will request a quality level increase from the server. The device waits for an RTT before sending another feedback. The server performs quality adjustments following feedback requests. If no feedback is received by the server, it gradually reduces the video quality level to the minimum.

Algorithm 1 Energy-efficient Quality Adaptation Algorithm

```
Since the Smart Device Received the First Data Packet:
while True do
    Calculate(RTT, PSNR, Exp_Life);
    if (Exp_Life < Thre_Life) then
        Decrease(Max_Level);
    else
        Increase(Max_Level);
    end if
    if (PSNR < Thre_PSNR) ∧ (QL > Lowest) then
        Request(Degrade, QL);
    else if (PSNR > Thre_PSNR) ∧ (QL < Max_Level) then
        Request(Upgrade, QL);
    end if
    Wait(RTT); ▷ Wait for RTT before sending feedback
end while
```

IV. Performance Evaluation

A. Testing Setup

DEAS was deployed in Network Simulator 3 [16], version 3.15. The simulation used the IEEE 802.11g wireless standard and the wired-cum-wireless topology illustrated in Fig.3 as a simplified case of heterogeneous wireless network. The wired links had 100 Mbps bandwidth and 2 ms latency. Users A and B are moving towards West with 0.5 m/s through the coverage area of the AP. The distance between the users and the AP

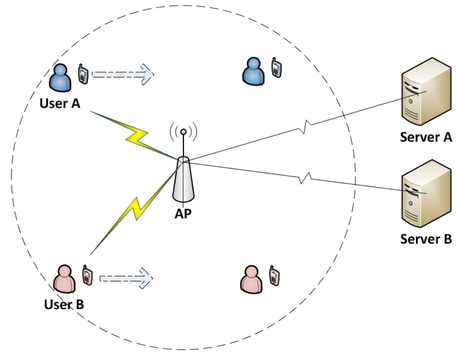

Fig. 3. Simulation Topology

ranges from 50m to 70m. Server A stores video content at five quality levels and transmits data packets to user A. Server B transmits UDP or TCP background traffic to user B. The background traffic differs in the six scenarios considered in the simulations as follows: A) 1 Mbps UDP traffic; B) 5 Mbps UDP traffic; C) 10 Mbps UDP traffic; D) 1 Mbps TCP traffic; E) 5 Mbps TCP traffic; F) 10 Mbps TCP traffic.

The compensation factor α_B in equation (7) is set to 1 to adhere to the linear depletion curve of the basic energy source model provided by NS-3. The battery voltage is 3.7 V, and the battery capacity is 186.48 J, 1% of the original value in order to reduce the simulation time until full depletion. *Thre_Life* and *Thre_PSNR* from Algorithm 1 are set to 100 s and 30 dB, respectively. The video content is of H.264/MPEG-4 format and stored in 5 quality levels with the average bit rates of 1920 kbps, 960 kbps, 480 kbps, 240 kbps and 120 kbps, from the highest to the lowest quality levels.

In order to perform realistic modeling and simulations, the power settings are configured according to *Trestian et. al* [10], which performed comprehensive measurements with a Google Nexus One mobile device in a real test bed. They collected power readings when playing video content at different quality levels. Notably, despite the tests on a particular device, due to the setup phase, DEAS algorithm is still device independent. These power settings are 1445 mW (1920 kpbs), 1022 mW (960 kbps), 841 mW (480 kbps), 764 mW (240 kbps), 699 mW (120 kbps) for the five quality levels considered. The simulation duration is 100 s. After 80 s, a new application is launched which adds 100 mW power consumption to the system. This is to test system's reaction to both change of device features (battery residual level change) and applications.

DEAS is compared against a Non Adaptive video delivery solution (NonAd) and SAMMy [8]. SAMMy is a fair choice for comparison as it also performs video content quality adaptation, which is mainly how DEAS achieves energy saving.

The three delivery schemes are compared in terms of average values of packet loss rate, device energy consumption, and PSNR in order to illustrate their levels of quality of service and estimated user perceived quality, respectively.

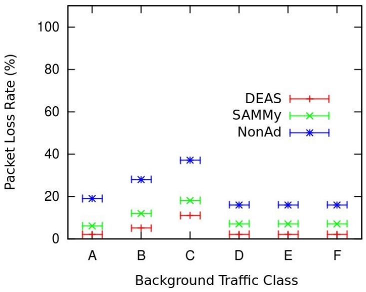

Fig. 4. Packet Loss Rate (%)

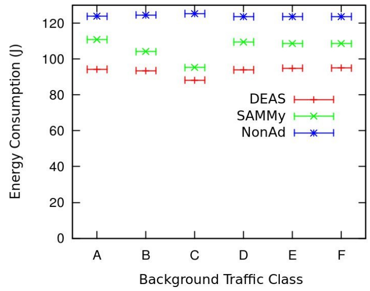

Fig. 5. Energy Consumption (J)

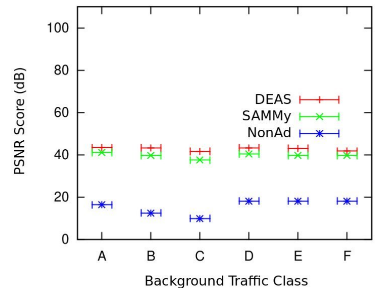

Fig. 6. Quality - PSNR (dB)

B. Simulation Results and Analysis

Fig. 4 shows how the packet loss ratios of the adaptive schemes are much lower than those of the NonAd solution. DEAS's loss ratio is below 10% in all scenarios, roughly 5% less than that of SAMMy's. SAMMy suffers more from loss (∼20%) in heavy UDP background traffic conditions. NonAd's loss rate is nearly 20% worse than DEAS's average value.

DEAS demonstrates significant benefits in terms of energy efficiency as illustrated in Fig. 5. DEAS outperforms SAMMy by 10% and NonAd by nearly 30% in all TCP scenarios and lightly loaded UDP scenarios. DEAS's energy saving performance is even more evident in heavy UDP traffic conditions, when it records up to 30% benefit over SAMMy and 40% benefit over NonAd.

In terms of estimated user perceived quality DEAS is better than both SAMMy (by 5% on average) and NonAd (by more than 20 dB), as shown in Fig. 6.

V. Conclusion

This paper presents *a Device characteristics-based differentiated Energy-efficient Adaptive Solution* (DEAS) for video delivery over heterogeneous wireless networks that performs content delivery adjustments with respect to the applications and device features in order to save device battery energy. DEAS novel energy-oriented system profiling addresses challenges introduced by the heterogeneity of mobile devices. DEAS new energy-efficient quality adaptation algorithm targets not only the active video application, but also other applications running on the same mobile device. Simulation results show how DEAS outperforms a traditional solution and another state of the art adaptive solution in terms of quality of service metrics and energy efficiency.

VI. Future Work

A real life testbed with different mobile devices deployed in heterogeneous wireless networks is needed to quantify the benefit and overhead introduced. Video characteristics (format, bit rate, compression rate, etc) will be considered in the Quality Adaptation Algorithm. Extensive real life testing will be conducted on various mobile devices with vast range of multimedia content and applications.

References

[1] "Cisco visual networking index: Global mobile data traffic forecast update, 2011 2016," Cisco System, Inc, Tech. Rep., Feb. 2012. [Online]. Available: http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.html

[2] Z. He, Y. Liang, L. Chen, I. Ahmad, and D. Wu, "Power-rate-distortion analysis for wireless video communication under energy constraints," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 15, no. 5, pp. 645 – 658, May. 2005.

[3] Y. Song, B. Ciubotaru, and G.-M. Muntean, "Application-aware adaptive duty cycle-based medium access control for energy efficient wireless data transmissions," in *Local Computer Networks, 2012. LCN 2012. IEEE 37th Conference on*, Oct. 2012.

[4] H. Yan, S. Watterson, D. Lowenthal, K. Li, R. Krishnan, and L. Peterson, "Client-centered, energy-efficient wireless communication on IEEE 802.11b networks," *Mobile Computing, IEEE Transactions on*, vol. 5, no. 11, pp. 1575 –1590, Nov. 2006.

[5] J. Adams and G.-M. Muntean, "Adaptive-buffer power save mechanism for mobile multimedia streaming," in *Communications, 2007. ICC '07. IEEE International Conference on*, Jun. 2007, pp. 4548 –4553.

[6] S. Saha, M. Hoque, and A. Lukyanenko, "Analyzing energy efficiency of a cooperative content distribution technique," in *Global Telecommunications Conference, 2011 IEEE*, Dec. 2011, pp. 1 –6.

[7] M.-H. Chen, C.-Y. Yang, C.-Y. Chang, M.-Y. Hsu, K.-H. Lee, and C.-F. Chou, "Towards energy-efficient streaming system for mobile hotspots," in *Proceedings of the ACM SIGCOMM 2011 conference*, ser. SIGCOMM '11. New York, NY, USA: ACM, 2011, pp. 450 –451.

[8] R. Trestian, O. Ormond, and G.-M. Muntean, "Signal strength-based adaptive multimedia delivery mechanism," in *Local Computer Networks, 2009. LCN 2009. IEEE 34th Conference on*, Oct. 2009, pp. 297 –300.

[9] M. Kennedy, H. Venkataraman, and G.-M. Muntean, "Battery and stream-aware adaptive multimedia delivery for wireless devices," in *Local Computer Networks, 2010. LCN 2010. IEEE 35th Conference on*, Oct. 2010, pp. 843 –846.

[10] R. Trestian, A.-N. Moldovan, O. Ormond, and G.-M. Muntean, "Energy consumption analysis of video streaming to android mobile devices," in *Network Operations and Management Symposium (NOMS), 2012 IEEE*, Apr. 2012, pp. 444 –452.

[11] M. Dong and L. Zhong, "Self-constructive high-rate system energy modeling for battery-powered mobile systems," in *Proceedings of the 9th international conference on Mobile systems, applications, and services*, ser. MobiSys '11. New York, NY, USA: ACM, 2011, pp. 335–348.

[12] G. Perrucci, F. Fitzek, and J. Widmer, "Survey on energy consumption entities on the smartphone platform," in *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd*, May 2011, pp. 1 –6.

[13] R. Ding and G.-M. Muntean, "A context-aware cross-layer energy-efficient adaptive routing algorithm for WLAN communications," in *Local Computer Networks, 2012. LCN 2012. IEEE 37th Conference on*, Oct. 2012, pp. 176 –179.

[14] ——, "A novel device and application-aware energy efficient routing algorithm for WLANs," in *GLOBECOM Workshops (GC Wkshps), 2012 IEEE*, Dec. 2012.

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

[16] "The Network Simulator - NS3." [Online]. Available: http://www.nsnam.org/