International Journal of Engineering and Technical Research (IJETR)
ISSN: 2321-0869, Volume-3, Issue-6, June 2015

A Novel Mobile Video Streaming and Efficient Social Video Sharing in the Clouds

Satbige Ashavini, Prof. Ms. Asha. S

Abstract— **While requests on feature movement over versatile systems have been souring, the remote connection limit can't keep up with the movement request. The crevice between the movement interest and the connection limit, alongside time-differing connection conditions, brings about poor administration nature of feature spilling over portable systems, for example, long buffering time and irregular interruptions. Leveraging the cloud computing technology, we propose a new mobile video streaming framework, dubbed AMES-Cloud, which has two main parts: AMoV (adaptive mobile video streaming) and ESoV (efficient social video sharing). AMoV and ESoV construct a private agent to provide video streaming services efficiently for each mobile user. For a given client, AMoV gives her a chance to private operators adaptively alter her spilling stream with a versatile feature coding strategy taking into account the criticism of connection quality. Similarly, ESoV screens the social system cooperation's among versatile clients, and their private specialists attempt to prefetch feature content ahead of time.**

Index Terms— **Adaptive video streaming, social video sharing, Mobile networks and cloud computing tools.**

I. INTRODUCTION

Cloud computing is changing more and more services on Internet [1,2]. In the area of IaaS, Amazon is the most popular cloud provider, but more and more providers are coming into this area. The numbers of cloud providers will increase explosively in future. Netflix is a video streaming service provider and based on Amazon EC2. It has been proved that a video service based on cloud computing is feasible. But with more cloud providers, how to choose from the providers is becoming increasingly important.

Over the previous decade, progressively more movement is accounted by feature gushing and downloading. Specifically, feature gushing administrations over versatile systems have get to be pervasive in the course of recent years [1]. While the feature gushing is not all that testing in wired systems, versatile systems have been experiencing feature activity transmissions over rare transfer speed of remote connections. In spite of system administrators' frantic endeavors to improve the remote connection transfer speed (e.g., 3G and LTE), taking off feature activity requests from portable clients are quickly overpowering the remote connection limit.

For a video service system based on cloud, the cost of renting storage and virtual machines (VM) are the main part of the total cost. The cost is dynamically changing with the need of applications. Less VMs than needed will result in a high resource occupancy rate. More VMs than needed will cause a waste of cost. The standard of the needed number is based on QoS. An appropriate resource occupancy rate of VM can reduce the packet loss or decoding delay in the video

However the vast majority of the proposition looking to together use the feature adaptability and flexibility depend on the dynamic control on the server side. That is, each portable client needs to exclusively report the transmission status (e.g., parcel misfortune, postpone and signal quality) intermittently to the server, which predicts the accessible transmission capacity for each client. In this manner the issue is that the server ought to assume control over the considerable handling overhead, as the quantity of clients increments.

Distributed computing systems are ready to adaptably give adaptable assets to substance/administration suppliers and procedure offloading to portable clients [13] [14] [15] [16] [17] [18] [19]. Hence, cloud server farms can undoubtedly procurement for expansive scale ongoing feature benefits as researched in [9] [20] . A few studies on portable distributed computing advances have proposed to produce customized insightful operators for adjusting portable clients, e.g., Cloudlet [21] and Stratus [22]. This is on the grounds that, in the cloud, different operators cases (or strings) can be kept up powerfully and proficiently relying upon the time-changing client requests.

In this paper, we design a adaptive video streaming and prefetching framework for mobile users with the above objectives in mind, dubbed AMES-Cloud. AMES-Cloud constructs a private agent for each mobile user in cloud computing environments, which is used by its two main parts: (i) AMoV (adaptive mobile video streaming), and ESoV (efficient social video sharing). AMoV offers the best conceivable spilling encounters by adaptively controlling the gushing bit rate depending on the change of the connection quality. AMoV conforms the bit rate for every client utilizing the adaptable feature coding. AMES-Cloud backings appropriating feature streams effectively by encouraging a 2-level structure: the first level is a substance conveyance system, and the second level is an information center. ESoV looks to furnish a client with moment playing of feature clasps by prefetching the feature cuts ahead of time from her private specialists to the neighborhood stockpiling of her gadget

II. ADAPTIVE AND EFFICIENT VIDEO STREAMING AND SHARING IN CLOUD

The figure 1 shows the architecture of the adaptive and efficient way of enhancing the video streaming and sharing of video to the mobile users. The architecture was constructed

Satbige Ashavini, Dept. of Computer Science And Engineering GNDEC Bidar, Karnataka, INDIA
Prof.Ms.Asha.S, Dept. of Computer Science and Engineering GNDEC, Bidar, Karnataka, INDIA

based on the video service provided in cloud called as ―AMES‖. The architecture contains A. Video service provider (VSP): the originated place of actual video data. It used the traditional video service provider. VSP can handle multiple request at the same time, while coming to the QoS with the mobile users, the VSP does not provide service up to the mark. B. Video cloud (VC): the cloud step up has been established with many components working together, virtually to get the original video data from the VSP and provide the reliable service to the mobile user and it also provides availability of video and makes the sharing of those videos among the users much easier. C. Video base (VB): Video base consists of the video data that are provided as the service to the mobile users in cloud.

D. Temp video base (TVB): it contains the most recently accessed video data and it also contains most frequently accessed video data. E. Vagent: it is an agent created for every mobile user who requests for the video service to the video cloud. F. Mobile users: the users who are mobile and providing the availability of the service to their location is difficult. The video cloud provides services under two main methodologies adaptive mobile video streaming and efficient mobile video sharing. The video streaming and video sharing plays the vital role in providing the reliable service to the customers. The rate in which frames of the videos are streams determines the quality and availability of the video service. Video data are most commonly shared among the users in the network. Mobile users are most commonly found to use social networking sites more offently [6,7]. The mobile device and mobile computing provides them space to be connected on the social network. Multimedia data such as images and videos are shared among the friend and users of the social media. The request of the video and sharing of video are two main actions requested from customer. Video cloud provides platform to provide these two services in better way.

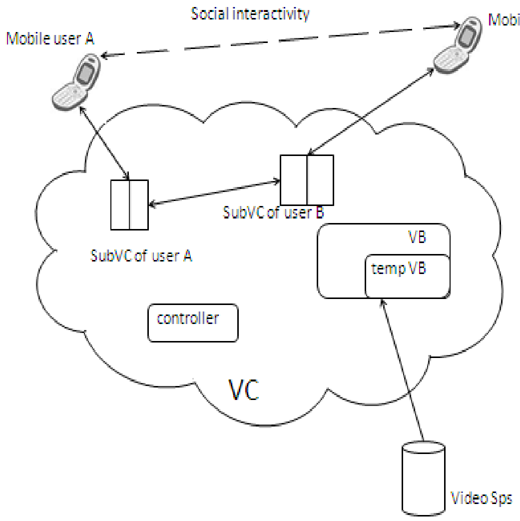


Fig1: Video Cloud Architecture

III. AMOV: ADAPTIVE MOBILE VIDEO STREAMING

As demonstrated in Fig. 2, conventional feature streams with settled bit rates can't adjust to the vacillation of the connection quality. For a specific bit rate, if the feasible connection data transfer capacity changes much, the feature gushing can be often ended because of the parcel misfortune. In SVC, a blend of the three least versatility is known as the Base Layer (BL) while the upgraded blends are called Enhancement Layers (ELs). To this respect, if BL is ensured to be conveyed, while more ELs can be likewise gotten when the connection can bear, a superior feature quality can be normal.

By utilizing SVC encoding systems, the server doesn't have to concern the customer side or the connection quality. Indeed a few parcels are lost, the customer still can translate the feature and presentation. However, this is still not data transmission productive due to the superfluous bundle misfortune. So it is important to control the SVC-based feature spilling at the server side with the rate adjustment system to proficiently use the data transmission.

We plan the versatile customer and the subVC with the structure as demonstrated in Fig. 3. The connection quality screen at portable customer continues following on measurements including sign quality, parcel round-trek time (RTT), jitter and bundle misfortune with a certain obligation cycle. Also, the customer will occasionally answer to the subVC. Therefore we characterize the cycle period for the reporting as the "time window", indicated by Twin, Note that the feature is likewise part by transient division by interim Twin.

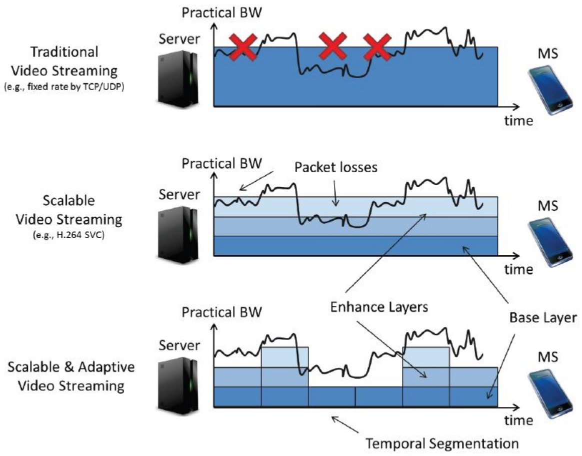


Fig2: A comparison of Traditional video streaming

Once the subVC gets the data of the connection quality, it will perform a count and anticipate the potential transmission capacity in whenever window. Note that we will utilize "anticipated data transfer capacity" and "anticipated goodput" reciprocally in taking after parts.

IV. ESOV: EFFICIENT SOCIAL VIDEO SHARING

In SNSs, clients subscribe to known companions, celebrated individuals, and specific intrigued substance distributers too; additionally there are different sorts of social exercises among clients in SNSs, for example, direct message and open posting. For spreading features in SNSs, one can post a feature in the general population, and his/her supporters can rapidly see it; one can additionally specifically prescribe a feature to indicated friend(s); moreover one can occasionally get saw by subscribed content distributer for new or well known features.

Like studies in [23] [24], we characterize distinctive quality levels for those social exercises to show the likelihood that the feature shared by one client may be observed by the recipients of the one's sharing exercises, which is known as a "hitting likelihood", so that subVCs can complete viable foundation prefetching at subVB and even localVB. Since after a feature sharing action, there may be a sure postpone that the beneficiary becomes acquainted with the sharing, and starts to watch [38]. Hence the prefetching in earlier won't affect the clients at most cases.

Rather, a client can snap to see immediately as the starting part or even the entire feature is as of now perfecthed at the localVB. The measure of prefetched portions is principally controlled by the quality of the social exercises. What's more, the prefetching from VC to subVC just alludes to the "connecting" activity, so there is just document finding what's more, connecting operations with minor defers; the prefetching from subVC to localVB additionally relies on upon the quality of the social exercises, however will likewise consider the remote connection status.

Algorithm 1: Matching Algorithm between BW and Segments:

```
i = 0
$BW_o$ = RBL
Transmit $BL_o$
Monitor $BW_0^{ipractical}$
Repeat
    Sleep for $T_{win}$
    Obtain $p_i$, $RTT_i$, $SINR_i$ etc., from client's report
    Predict $BW_{i+1}^{estimate}$ (or $BW_{i+1}^{estimate} = BW_i^{practical}$)
    k=0
    $BW_{EL}$=0
    Repeat
        K++
        if k >= j break
        $BW_{EL}=BW_{EL} + R_{EL}^k$
    Until $BW_{EL} >= BW_{i+1}^{estimate} - R_{BL}$
    Transmit $BL_{i+1}$ and $EL^1_{i+1}$, $EL^2_{i+1}$... $EL_{i+1}^{k-1}$
    Monitor $BW_{i+1}^{practical}$
    i++
Until all video segments are transmitted
```

We order the social exercises in current mainstream SNSs into three sorts, with respect to the effect of the exercises also, the potential responding need from the perspective of the beneficiary:

Subscription: Like the well known RSS administrations, a client can subscribe to a specific feature distributer or an extraordinary feature gathering administration in light of his/her advantage. This hobby driven network between the endorser and the feature distributer is considered as "middle", in light of the fact that the supporter may not generally observe all subscribed features.

Direct proposal: In SNSs, a client straightforwardly prescribe a feature to specific friend(s) with a short message. The beneficiaries of the message may watch it with high likelihood. This is considered as "solid". Public sharing: Each client in SNSs has a timetable based of action stream, which demonstrates his/her later exercises. The movement of a client watching or sharing a feature can be seen by his/her companions (or adherents)

Diverse qualities of the social exercises show distinctive levels of likelihood that a feature will be soon observed by the beneficiary. Correspondingly we likewise characterize three prefetching levels in regards to the social exercises of versatile clients: "Parts": Because the features that distributed by memberships may be observed by the supporters with a not high likelihood, we propose to just push a piece of BL and ELs portions, for instance, the initial 10% fragments. "All": The feature shared by the immediate suggestions will be viewed with a high likelihood, so we propose to prefetch the BL and all ELs, with a specific end goal to let the recipient(s) straightforwardly watch the feature with a decent quality, with no buffering. "Little": people in general sharing have a feeble integration among clients, so the likelihood that a client's companions (supporters) watch the feature that the client has watched or shared is low. We propose to just prefetch the BL fragment of the first run through window to start with to the individuals who have seen his/her action in the stream.

V. VIDEO STORAGE AND STREAMING FLOW BY AMOV AND EMOS

The two sections, AMoV and EMoS, in AMES-Cloud system have tight associations and will together administration the feature spilling and sharing: they both depend on the distributed computing stage and are done by the private organizations of clients; while prefetching in EMoS, the AMoV will in any case screen and enhance the transmission considering the connection status; with a certain measure of prefetched fragments by EMoS, AMoV can offer better feature quality.

With the endeavors of AMoV and EMoS, we delineate the stream outline of how a feature will be spilled in Fig. 3. Note that keeping in mind the end goal to trade the features among the localVBs, subVBs, tempVB and the VB, a feature map (VMap) is utilized to demonstrate the obliged sections.

When a portable client begins to watch a feature by a connection, the localVB will first be checked whether there is any prefetched portions of the feature so it can straightforwardly begin. In the event that there is none or simply a few sections, the customer will report a comparing VMap to its subVC. On the off chance that the subVC has prefetched parts in subVB, the subVC will start the fragment transmission. In any case, if there is additionally none in the subVB, the tempVB and VB in the inside VC will be checked. For a non-existing feature in AMES-Cloud, the authority in VC will promptly bring it from outer feature suppliers by means of the connection; after re-encoding the feature into SVC organization, taking somewhat more defer, the subVC will exchange to the versatile client.

Likewise in AMES-Cloud, if a feature is shared among the subVCs at a certain recurrence edge (e.g., 10 times every day), it will be transferred to the tempVB of the VC; and on the off chance that it is further shared at a much higher recurrence (e.g., 100 times each day), it will be put away with a more drawn out lifetime in the VB. In such a way, which is truly comparable to the leveled CPU store, the subVB and VB can simply store crisp and well known features to expand the likelihood of re-use.

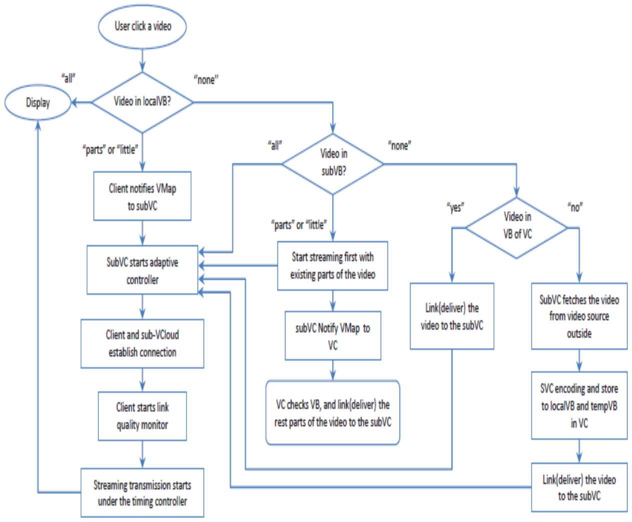


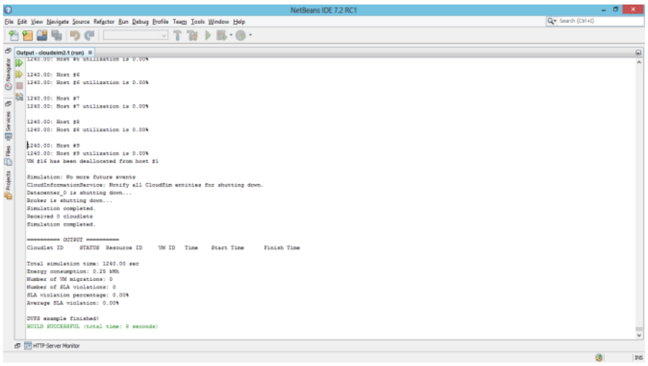
Fig 3: subVC and VC of AMES cloud framework

VI. IMPLEMENTATION AND EVOLUTION

We present tests and evaluation that we undertook in order to quantify the efficiency of CloudSim in modeling and simulating Cloud computing environment. The tests were conducted on a Celeron machine having configuration: 1.86GHz with 1MB of L2 cache and 1 GB of RAM running a standard Windows version 7 and JDK 1.7. To evaluate the overhead in building a simulated Cloud computing environment that consists of a single data center, a broker and a user, we performed series of experiments. The number of hosts in the data center in each experiment was varied from 100 to 100000. As the goal of these tests were to evaluate the computing power requirement to instantiate the Cloud simulation infrastructure, no attention was given to the user workload.

For the memory test, we profile the total physical memory used by the hosting computer (Celeron machine) in order to fully instantiate and load the CloudSim environment. The total delay in instantiating the simulation environment is the time difference between the following events: (i) the time at which the runtime environment (java virtual machine) is directed to load the CloudSim program; and (ii) the instance at which CloudSim's entities and components are fully initialized and are ready to process events.

We have organized this experiment to evaluate the performance of Bandwidth, energy consumption and delay ratio. We organized this experiment in cloudsim and cloud analyst software.



```
1240.00: Host #5 utilization is 0.00%

1240.00: Host #6
1240.00: Host #6 utilization is 0.00%

1240.00: Host #7
1240.00: Host #7 utilization is 0.00%

1240.00: Host #8
1240.00: Host #8 utilization is 0.00%

1240.00: Host #9
1240.00: Host #9 utilization is 0.00%
VM #16 has been deallocated from host #1

Simulation: No more future events
CloudInformationService: Notify all CloudSim entities for shutting down.
Datacenter_0 is shutting down...
Broker is shutting down...
Simulation completed.
Received 0 cloudlets
Simulation completed.

========== OUTPUT ==========
Cloudlet ID     STATUS  Resource ID     VM ID   Time    Start Time      Finish Time

Total simulation time: 1240.00 sec
Energy consumption: 0.25 kWh
Number of VM migrations: 0
Number of SLA violations: 0
SLA violation percentage: 0.00%
Average SLA violation: 0.00%

DVFS example finished!
BUILD SUCCESSFUL (total time: 8 seconds)
```

Fig 4: Cloudsim output

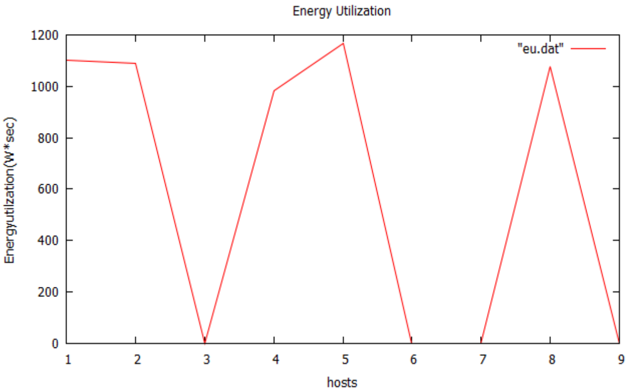


Fig 5: Energy Utilization

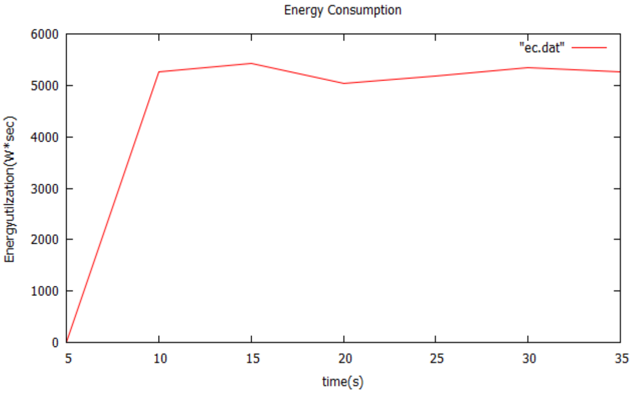


Fig 6: Energy Consumpoin

VII. CONCLUSION

In this paper, we discussed our proposal of an adaptive mobile video streaming and sharing framework, called AMES-Cloud, which efficiently stores videos in the clouds (VC), and utilizes cloud computing to construct private agent (subVC) for each mobile user to try to offer "non-terminating" video streaming adapting to the fluctuation of link quality based on the Scalable Video Coding technique. Also AMES-Cloud can further seek to provide "nonbuffering" experience of video streaming by background pushing functions among the VB, subVBs and localVB of mobile users. We evaluated the AMES-Cloud by prototype implementation and shows that the cloud computing technique brings significant improvement on the adaptivity of the mobile streaming. The focus of this paper is to verify how cloud computing can improve the transmission adaptability and prefetching for mobile users. We ignored the cost of encoding workload in the cloud while implementing the prototype. As one important future work, we will carry out large-scale implementation and with serious consideration on energy and price cost. In the future, we will also try to improve the SNS-based prefetching, and security issues in the AMES-Cloud.

REFERENCES

[1] CISCO, "Cisco Visual Networking Index : Global Mobile Data Traffic Forecast Update , 2011-2016," Tech. Rep., 2012.

[2] Y. Li, Y. Zhang, and R. Yuan, "Measurement and Analysis of a Large Scale Commercial Mobile Internet TV System," in *ACM IMC*, pp. 209–224, 2011.

[3] T. Taleb and K. Hashimoto, "MS2: A Novel Multi-Source Mobile-Streaming Architecture," in *IEEE Transaction on Broadcasting*, vol. 57, no. 3, pp. 662–673, 2011.

[4] X. Wang, S. Kim, T. Kwon, H. Kim, Y. Choi, "Unveiling the BitTorrent Performance in Mobile WiMAX Networks," in *Passive and Active Measurement Conference*, 2011.

[5] A. Nafaa, T. Taleb, and L. Murphy, "Forward Error Correction Adaptation Strategies for Media Streaming over Wireless Networks," in *IEEE Communications Magazine*, vol. 46, no. 1, pp. 72–79, 2008.

[6] J. Fernandez, T. Taleb, M. Guizani, and N. Kato, "Bandwidth Aggregation-aware Dynamic QoS Negotiation for Real-Time Video Applications in Next-Generation Wireless Networks," in *IEEE Transaction on Multimedia*, vol. 11, no. 6, pp. 1082–1093, 2009.

[7] T. Taleb, K. Kashibuchi, A. Leonardi, S. Palazzo, K. Hashimoto, N. Kato, and Y. Nemoto, "A Cross-layer Approach for An Efficient Delivery of TCP/RTP-based Multimedia Applications in Heterogeneous Wireless Networks," in *IEEE Transaction on Vehicular Technology*, vol. 57, no. 6, pp. 3801–3814, 2008.

[8] K. Zhang, J. Kong, M. Qiu, and G.L Song, "Multimedia Layout Adaptation Through Grammatical Specifications," in *ACM/Springer Multimedia Systems*, vol. 10, no. 3, pp.245–260, 2005.

[9] M. Wien, R. Cazoulat, A. Graffunder, A. Hutter, and P. Amon, "Real-Time System for Adaptive Video Streaming Based on SVC," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1227–1237, Sep. 2007.

[10] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.

[11] H. Schwarz and M. Wien, "The Scalable Video Coding Extension of The H. 264/AVC Standard," in *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp.135–141, 2008.

[12] P. McDonagh, C. Vallati, A. Pande, and P. Mohapatra, "Quality-Oriented Scalable Video Delivery Using H. 264 SVC on An LTE Network," in *WPMC*, 2011.

[13] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud Computing: State-of-the-art and Research Challenges," in *Journal of Internet Services and Applications*, vol. 1, no. 1, pp. 7–18, Apr. 2010.

[14] D. Niu, H. Xu, B. Li, and S. Zhao, "Quality-Assured Cloud Bandwidth Auto-Scaling for Video-on-Demand Applications," in *IEEE INFOCOM*, 2012.

[15] Y.G. Wen, W.W. Zhang, K. Guan, D. Kilper, and H. Y. Luo, "Energy-Optimal Execution Policy for A Cloud-Assisted Mobile Application Platform," Tech. Rep., September 2011

[16] W.W. Zhang, Y.G. and D.P. Wu, "Energy-Efficient Scheduling Policy for Collaborative Execution in Mobile Cloud Computing," in *INFOCOM, Mini Conf.*, 2013.

[17] W.W. Zhang, Y.G. Wen, Z.Z. Chen and A. Khisti, "QoE-Driven Cache Management for HTTP Adaptive Bit Rate Streaming over Wireless Networks," in *IEEE Transactions on Multimedia*, November 2012.

[18] J. Li, M. Qiu, Z. Ming, G. Quan, X. Qin, and Z. Gu, "Online Optimization for Scheduling Preemptable tasks on IaaS Cloud systems," in *Journal of Parallel and Distributed Computing (JPDC)*, vol.72, no.5, pp.666-677, 2012.

[19] P. Calyam, M. Sridharan, Y. Xu , K. Zhu , A. Berryman, R. Patali, and A. Venkataraman, "Enabling Performance Intelligence for Application Adaptation in the Future Internet," in *Journal of Communication and Networks*, vol. 13, no. 6, pp. 591–601, 2011.

[20] Z. Huang, C. Mei, L. E. Li, and T. Woo, "CloudStream : Delivering High-Quality Streaming Videos through A Cloud-based SVC Proxy," in *IEEE INFOCOM*, 2011.

[21] N. Davies, "The Case for VM-Based Cloudlets in Mobile Computing," in *IEEE Pervasive Computing*, vol. 8, no. 4, pp. 14–23, 2009.

[22] B. Aggarwal, N. Spring, and A. Schulman, "Stratus : Energy-Efficient Mobile Communication using Cloud Support," in *ACM SIGCOMM DEMO*, 2010.

[23] Y. Zhang, W. Gao, G. Cao, T. L. Porta, B. Krishnamachari, and A. Iyengar, "Social-Aware Data Diffusion in Delay Tolerant MANET," *Handbook of Optimization in Complex Networks: Communication and Social Networks*, 2010.

[24] Z. Wang, L. Sun, C. Wu, and S. Yang, "Guiding Internet-Scale VIdeo Service Deployment Using Microblog-Based Prediction," in *IEEE INFOCOM*, 2012.