Wireless Video Traffic Bottleneck Coordination with a DASH SAND Framework

Zhu Li, Shuai Zhao, Deep Medhi
Computer Science & Electrical Engineering Department
University of Missouri – Kansas City, USA
{ lizhu, Shuai.Zhao,dmedhi@umkc.edu}

Imed Bouazizi
Samsung Research America, USA
{i.bouazizi@samsung.com}

***Abstract*—MPEG DASH is a client driven, pull based, adaptive streaming technology that allows for quality and bandwidth trade offs at streaming times. This is especially useful for mobile and Over The Top (OTT) video streaming applications, as throughput variations are inevitable. However, when multiple DASH clients are sharing a common wireless bottleneck with under-provisioned resources, e.g., multiple devices sharing the same home cable connection, or multiple UEs sharing the radio resource served by the same base station, simple adaptation within the same traffic session is not enough and may lead to sub-optimal solutions. In this work, we introduce new Quality of Experience (QoE) metrics for DASH sub-representations, and propose a marginal utility maximization-based resource pricing scheme to coordinate multiple video traffic sharing the bottleneck resource. The solution is based on the DASH SAND (Server And Network assisted DASH) messaging framework. Simulation results demonstrate the QoE multiplexing gains from this solution, and the pricing control scheme is adopted in SAND messages.**

***Index Terms*—QoE; DASH Video Streaming; Resource Management; Network Optimization**

I. Introduction

The advances in 3G/4G mobile communication technology and explosive growth of mobile devices are fueling an unprecedented rapid growth of video traffic over mobile networks. The wireless capacity is lagging far behind this rapid growth, and how to deal with it is the main issue for mobile network operators. According to some marketing studies [1], the mobile video growth rate will be 90% per year from 2013 through 2017, which far outpaces the growth of the mobile network capacity.

In addition to improving video coding and wireless spectrum efficiency, more efficiency can be extracted from the optimization of the multimedia transport and mobile network operation optimization. MPEG developed an adaptive streaming of video over HTTP protocol, called DASH [2], that allows for the adaptation of variable bit rate streaming of videos over a throughput varying channel. This is well suited for the mobile video application, where instead of streaming at a constant rate that may cause a freeze in playback when the network bandwidth is low, DASH allows for switching to a lower rate to avoid freezes and continues the streaming with a lower quality version. How to best adapt the DASH rate to the network constraints and achieve the best quality of experience (QoE) possible has been well studied, as in [3], [4], within a single session. To deal with bottlenecks where multiple video sessions are sharing the link, DASH is lacking the information and mechanisms. Recently, a Server And Network Assisted DASH (SAND) amendment work was created within the MPEG DASH ad hoc group, and introduce a new set of messages and mechanisms that address a variety of efficiency issues, including resource allocation, congestion, and QoE signalling. SAND [5] allows for more flexibility in deploying the DASH solution by CDN operators that which collects a lot of network congestion, client side buffers, and QoE information that should be utilized to make the end-to-end DASH solution optimal.

In this work, we address the aforementioned problems by developing a bottleneck coordination solution that can be implemented as a set of DASH SAND messages. The coordination problem is formulated as a resource constrained total utility maximization problem. New QoE metrics [6] are introduced to characterize a spatio-temporal quality of DASH sub-representation and offering much finer granular streaming operating points. A resource pricing solution is derived to compute the optimal streaming operating points for clients to achieve total QoE gains over non-coordinated adaptations. Simulation results demonstrate the effectiveness of the proposed solution, and video demo clips are also available for subjective evaluation.

In the rest of this paper, we explain DASH bottleneck coordination problem in Section II, then introduce our QoE metrics and DASH utility function in Section III. Section IV shows our simulation results. We conclude the paper in Section V.

II. DASH Bottleneck Coordination Problem

The growth of video data traffic is far out pacing the network capacity growth, and it dictates the reality that a large portion of mobile video sessions will be operating at a quality of service (QoS) deficit over a bottleneck. The bottleneck can happen over both the eNodeB wireless channels, over the home broadband gateway, and the links in the mobile core networks. The original DASH framework offers an effective rate adaptation scheme within a single streaming session, but lacks the coordination mechanisms among users sharing a common bottleneck. In recent work, DASH SAND [5] is addressing this problem by introducing new messaging and control schemes among DASH servers, clients and middle boxes (known as DANE: Dash Aware Network Elements). The overall SAND architecture is illustrated in Fig. 1 [7].

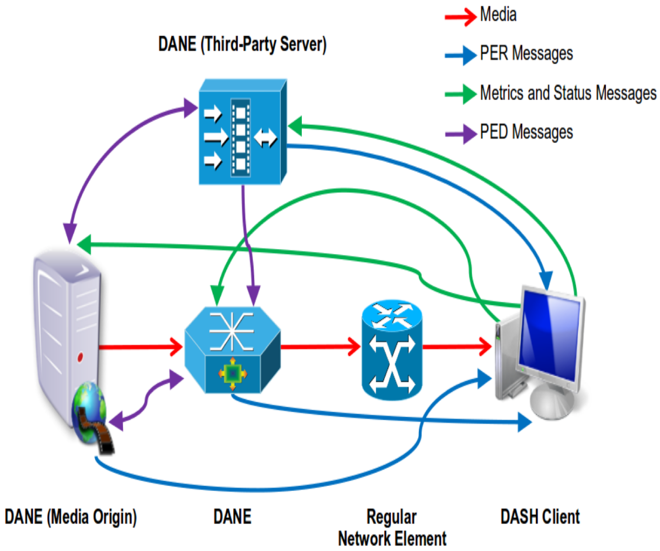


Fig. 1: DASH SAND Architecture

The media data plane is illustrated in red, and the measurement and request messages from the clients to the server and DANE are in green. The coordination message from DANE and the server is in blue. In this work we will introduce a set of messages and distributed computing solutions for bottleneck coordination [8], [9] based on the DASH SAND framework. Without losing the generality, the bottleneck coordination problem can be formulated as:

$$\max_{x_1,x_2,\ldots x_n} \sum_{k=1}^{n} U_k(x_k), s.t., \sum_{k=1}^{n} x_k \le C \qquad (1)$$

where x_k are the bottleneck resources, or throughput allocated to the video session k, and the utility function, which typically reflects the rate distortion or QoS-QoE tradeoffs, are represented as U_k. The total resource constraint is C, for the scheduling period with duration T. When $\sum_{k=1}^{n} x_{kt} > C$, the video sessions sharing the link are considered a bottlenecked link for the duration.

For the constrained optimization problem in Eq. (1), it is well known from the Kuhn-Tucker [10] condition, that when the link is in the bottleneck mode, the optimal solution is tight on the resource constraint, and the minimizer to the Lagrangian relaxed Eq. (1) has the same gradient on the utility function plane. This is verified by taking the Lagrangian of Eq. (1),

$$L(X,\lambda) = \max_{x_1,x_2,\ldots x_n} \sum_{k=1}^{n} U_k(x_k) - \lambda(\sum_{k=1}^{n} x_k - C)$$

and sets its first order differentiation to zero; this gives us the optimal resource allocation condition:

$$\frac{\partial L}{\partial x_k^*} = 0 \implies U'(x_k^*) = -\lambda, \forall k \qquad (2)$$

that decouples the original constrained problem into an individual resource operating points x_k^* search problem. This indicates that at the optimal allocation with a tight constraint, the utility gradient functions have the same value at the total/average utility maximizer x_k^*.

This is intuitively true. In case there is a certain session that can derive more utility gains by shifting resources from other sessions, then the allocation is not optimal. This gradient $-\lambda$ is also interpreted as the resource price [11]. A distributed solution can indeed be developed with this formulation and decomposition. A bottleneck coordinator can be identified from some DASH Aware Network Elements (DANE), an eNodeB or eNodeB controller, or a home gateway, for example, that measures and estimates the bottleneck throughput, issuing resource/throughput prices in iteration, then each DASH client can perform a local optimization by,

$$x(i)_k^* = arg \max_x U_k(x) - \lambda x^i \qquad (3)$$

basically requesting an optimal resource allocation in price iteration i, as the one that maximizes the surplus function in Eq. (3). The price iteration will converge if the resource allocation is tight enough towards the resource constraint.

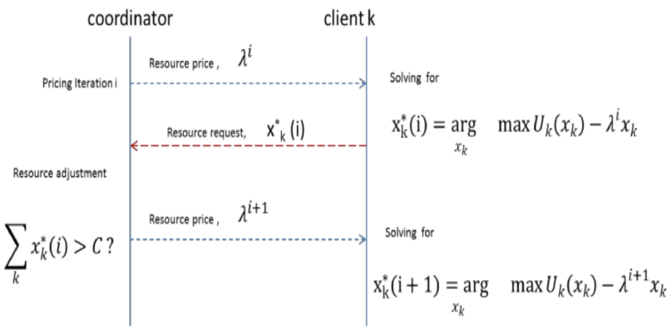

coordinator
client k
Resource price , λ^i
Pricing Iteration i
Solving for
$x_k^*(i) = \arg \max_{x_k} U_k(x_k) - \lambda^i x_k$
Resource request, $x^*_k(i)$
Resource adjustment
$\sum_k x_k^*(i) > C$?
Resource price , λ^{i+1}
Solving for
$x_k^*(i+1) = \arg \max_{x_k} U_k(x_k) - \lambda^{i+1} x_k$

Fig. 2: Resource pricing for DASH bottleneck coordination

This process is illustrated in Fig. 2, where a bottleneck issues resource prices λ_i in iterations, the clients will compute the surplus maximization according to Eq. 3) and report back the resource requests $x(i)_k$. If the total requests exceed the resource constraint in Eq. (1), then the price is raised for the next iteration. A bi-section search can usually converge quickly in 3∼5 iterations to a practically tight allocation of the resources.

III. QOE METRICS AND DASH UTILITY FUNCTION

For the modern streaming solutions like MPEG DASH [2], the distortion does not result from packet losses. A common fallacy assumes that the receiver will somehow cope with packet losses and employ error-resilient solutions to mitigate that. Instead, in reality, most of the distortions are not due to the loss but rather the delay, this is especially true for the DASH case, where the underlying TCP transport makes sure that the video streams are received in a byte stream in order and without error. A proper utility modeling of different streaming operation options that can be estimated at the content preparation time and hopefully reflected either in the DASH MPD or SAND messages, is desirable. The resulting solution is a bottleneck coordination and DASH transmission, pacing/congestion avoidance solution.

To have a useful solution in practical applications for DASH video coordination, additional challenges need to be overcome, especially how to characterize the QoE of the received DASH segments. DASH achieves rate adaptation by having multiple rate representations of the same video content in segments. The encoding SNR quality of the segment can be computed at the content encoding time. Indeed, the MPEG ISOBMFF file formats allow for the carriage of such information [6]. However, the number of operating points are usually rather limited at each coordination interval, as typically only 3∼5 rate representations are prepared and signalled in the DASH MPD. This leads to a suboptimal allocation of the bottleneck throughputs as the operating points are lacking fine granularity in the rates.

In this work, we introduce a new temporal quality metric and DASH sub-representation packing solution, to offer finer granular streaming operating points in rates and graceful spatio-temporal quality degradation when streaming rates need to be reduced. For a pre-coded DASH segment at a certain PSNR quality level, we can further derive temporal quality operating points by packing the I, P, and B frames according to their if-loss distortion-induced characteristics. This is achieved by a frame significance function based frame loss distortion metric [6]. The frame significance function V_k characterizes the relative importance of the frame k in the visual sequence, computed as the visual difference it introduces w.r.t the previous frame,

$$V_k = d(f_k, fk - 1)$$

while the frame loss induced distortion is a weighted summation of lost frame visual significance:

$$D(L) = \sum_{k=1}^{m} l_k V_k e^{-h(k-p(k)-1)} \quad (4)$$

where the frame loss index L=[$l_1, l_2, ..., l_n$] indicates the frame loss (1) is present in the sequence (0), and $p(k)$ is the previous frame received, an exponentially decaying weight function where the heat kernel h is used. More details are in [6]. This gives us a new distortion on frame loss as in Eq. (4), that captures any temporal layering of the frame samples, and resulting distortion if certain layers are not received. This in combination with DASH's segment indexing scheme, as illustrated in Fig. 3 (from Iraj's DASH tutorial slides), allow us further fine granular streaming options at a sub-representation level, to achieve a certain temporal and PSNR quality level.

Now a DASH video segment can be represented as a combination of different PSNR quality layers as well as the sub-representation within each PSNR layer that reflects a different temporal quality. The utility of this sub-representation level video segment is therefore computed as the linear combination of the PSNR and temporal quality:

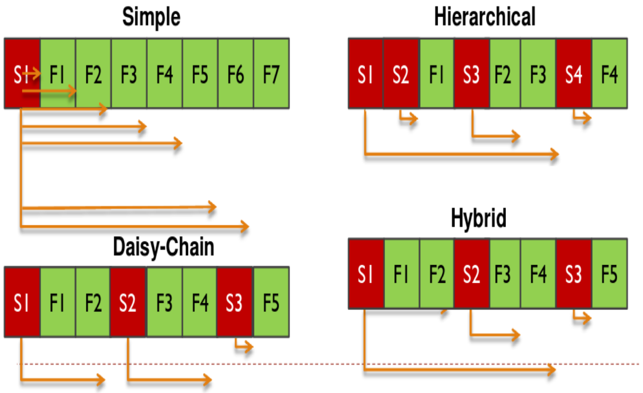


Fig. 3: DASH Sub-Segment Access

$$U(s_k) = PSNR(s_k) - wD(L(s_k)) \quad (5)$$

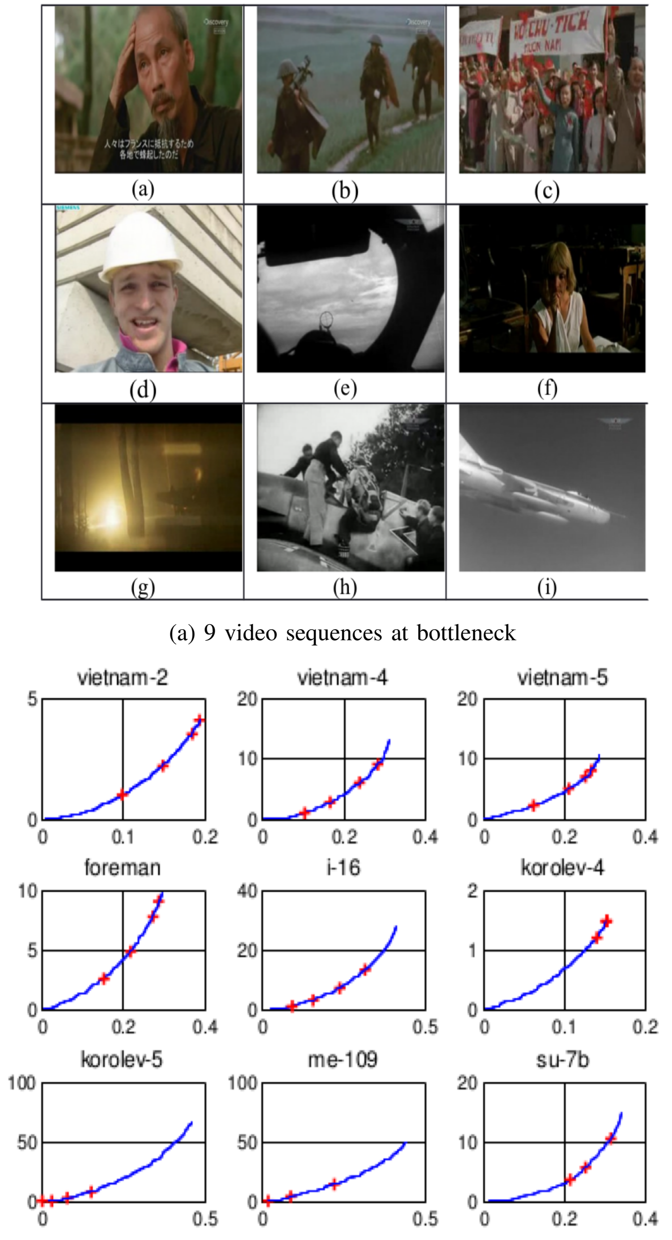
The $L(s_k)$ is the frame loss index of the sub-representation s_k. The utility is better for a larger PSNR value, and worse off for a larger temporal loss, and the weight w reflects this tradeoff between the PSNR and temporal distortions. The implementation can be very flexible in this case. Indeed in MMT we introduced a cache manifest [12] that signals this spatio-temporal quality operating points. Notice that this utility modeling provides much finer granular streaming time rate operating points, as well as allows for more graceful QoE degradation when resources are under provisioned.

IV. SIMULATION RESULTS

To test the effectiveness of the proposed algorithm, we set up a wireless bottleneck that needed a throughput reduction of x=[10%, 15%, 20%, 25%], and the bottleneck was shared by 9 video sessions consisting of a variety of activity levels and rate- distortion characteristics. The 9 test sequences are all from Youtube, and a thumbnail is illustrated in Fig. 6 as, “vietnam-2”, “vietnam-4”, “vietnam-5”, “foreman”, “i-16”, “korolev-4”, “korolev-5”, “me-109” and “su-7b”.

As indicated in section II, a DANE node will coordinate the operating points with these 9 DASH clients via resource pricing. To further demonstrate the effects of the new frame loss induced distortion metric, we further limited the adaptation along the temporal quality layer, by setting the utility function temporal quality weight in Eq. (5) as w=1600 and basically over-emphasizing the temporal loss as a way of showcasing the effectiveness of this new temporal metric.

For each segment, different combinations of B-frames based sub-representations induced different distortion and rate reduction, if dropped. This is illustrated in Fig. 4a, for the aforementioned 9 sequences. Notice that the gradient reflects the distortion induced over rate reduction; the smaller the



(a) 9 video sequences at bottleneck

(b) Bottleneck Coordination R-D Operating Points

Fig. 4: DASH SAND Bottleneck Coordination Operating Points

gradient, the more profitable it was to drop.

It is obvious that these 9 sequences are different in their ease of dropping. Some active sequences like me-109 and su-7b, consist of very active scenes and multiple scene cuts, and therefore, are the most difficult to prune; whereas the korolev-2, foreman is easier to prune. For the bottleneck throughput reduction operating points of x=[10%, 15%, 20%, 25%], the resulting R-D operating points for each individual sequences are plotted in Fig. 4b.

For a lower reduction rate of x=10%, a more difficult sequence like "su-7b" and "me-109" do not lose any frames/sub-representations. This is indeed what is desired, to let the easy sequence help out the busy sequences at the bottleneck. The pruned test sequences from these bottleneck coordination operations are available for subjective evaluation as well. The resulting sequences exhibit fairly graceful QoE degradation and are much more desirable than the uncoordinated solutions.

V. Conclusion and Future Work

In this work, we introduced a set of DASH SAND messages that support bottleneck coordination across multiple video traffic that share the bottleneck. The core idea is a resource pricing control that can be adapted to any DASH content with a combination of spatio-temporal quality layering schemea. The resource pricing reflects the rate reduction gradients on the spatio-temporal distortion plane, the steepest gradient is the most desirable for the rate reduction at bottleneck for individual video sessions, whereas the pricing control ensures that a fairness in the sense of socially optimal utility, i.e, no re-allocation of resource can improve the totally utility received. Simulation results demonstrated the effectiveness of the proposed solution.

In the future, we will conduct more use cases to find the perceptually optimal steepest rate gradient on the spatio-temporal distortion plane, i.e, find the tradeoff between spatial vs temporal quality degradation, and deploy the framework on the GENI [13] test bed and conduct more extensive studies in real world situations with various background traffic loads.

References

[1] V. N. I. Cisco, "Forecast and methodology, 2012–2017," *White Paper*, 2013.

[2] I. Sodagar, "The mpeg-dash standard for multimedia streaming over the internet," 2011.

[3] C. Liu, I. Bouazizi, and M. Gabbouj, "Rate adaptation for adaptive http streaming," in *Proceedings of the second annual ACM conference on Multimedia systems*. ACM, 2011, pp. 169–174.

[4] A. El Essaili, D. Schroeder, D. Staehle, M. Shehada, W. Kellerer, and E. Steinbach, "Quality-of-experience driven adaptive http media delivery," in *2013 IEEE International Conference on Communications (ICC)*. IEEE, 2013, pp. 2480–2485.

[5] I. D. 23009-5, *Information technology – Dynamic adaptive streaming over HTTP (DASH) – Part 5: Server and network assisted DASH (SAND)*, Std.

[6] Z. Li and I.Bouazizi, *FF: Temporal Quality Signalling in ISO Based Media File Format*, ISO/IEC/JTC1/MPEG2014/m33239 Std.

[7] C. T. Rmi Houdaille, *DASH/CESAND: cooperative parameters*, ISO/IEC JTC1/SC29/WG11 MPEG 111/m36033 Std.

[8] Z. Li, J. Huang, and A. K. Katsaggelos, "Pricing based collaborative multi-user video streamming over power constrained wireless downlink," in *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, vol. 5. IEEE, 2006, pp. V–V.

[9] Y. Li, Z. Li, M. Chiang, and A. R. Calderbank, "Content-aware distortion-fair video streaming in congested networks," *IEEE Transactions on Multimedia*, vol. 11, no. 6, pp. 1182–1193, 2009.

[10] H.-T. KUHN, "Aw (1951) nonlinear programming," in *2nd Berkeley Symposium. Berkeley, University of California Press*.

[11] J. Huang, Z. Li, M. Chiang, and A. K. Katsaggelos, "Joint source adaptation and resource allocation for multi-user wireless video streaming," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 5, pp. 582–595, 2008.

[12] Z. Li and I. Bouazizi, *MMT Amd1: Multiple QoE Operating Points Signalling in MMT ADC*, ISO/IEC JTC1/SC29/WG11/MPEG2013/m33237 Std.

[13] C. Elliott, "Geni-global environment for network innovations." in *LCN*, 2008, p. 8.