International Journal on Recent and Innovation Trends in Computing and Communication
Volume: 6 Issue: 4
ISSN: 2321-8169
311 - 316

Adaptional Algorithmic Rule for Video Streaming

Miss. Suvidha Sonawane
Student, Computer Engineering
Government College of Engineering, Jalgaon
Jalgaon, Maharashtra, India
sonawane.suvidha@gmail.com

Prof.Pramod B. Gosavi
HOD &Associate Professor, Computer engineering
GF's College of Engineering, Jalgaon
Jalgaon, Maharashtra, India
gosavi.pramod@gmail.com

Abstract**:** Nowadays major wants of top quality videos while not reproduce interruption. Considering on various request at the same time, it was delaying videos of upper bitrates. During this paper, the load balancing video streaming method we tend to thought of as reinforcement learning task. We tend to outline a state to explain this state of affairs, as well as the index of the requested phase, this offered information measure and alternative system parameters for every streaming step. For this reinforcement learning task a finite state Andre Markov Decision Process (MDP) may be sculptured. To specialize in the problems connected with video streaming is that the psychological feature work of this analysis. On load equalization on server, less buffering, individuals have less time to buffering video and streaming depends. However, with the introduction of two internet server system performance is seems to be increased.

Keyword - *DASH, Markov decision process, video streaming, Load balancer.*

*****

I. INTRODUCTION

Recently, among the mobile users VIDEO streaming is nothing but a gaining quality. Considering that the mobile devices have restricted process capability and energy offer, so it's terribly difficult to produce prime quality video streaming services for mobile users systematically and therefore the wireless channels square measure extremely dynamic. With completely different wireless communication techniques for mobile devices, it's a promising trend to use multiple wireless network interfaces. For instance, good phones and tablets square measure typically equipped with cellular, local area network and Bluetooth interfaces. Utilizing load balancer at the same time will improve video streaming in many aspects: the aggregative higher information measure will support video of upper bit rate; once one wireless link suffers poor link quality or congestion, the opposite scan catch up on it.

While reducing the wireless service value, so as to keep up high video streaming quality, during this paper, like Markov Decision Process (MDP), the best video streaming method with multiple links is developed. The standard of service (QoS) necessities for video traffic, like the start-up latency, playback fluency, average playback quality, playback smoothness and wireless service value, for this thought the reward operate is meant. We have a tendency to propose associate adjustive, best-action search algorithmic rule to get a sub-optimal resolution for to resolve the MDP in real time. We have a tendency to enforce a testbed mistreatment the automaton movable and therefore the Scalable Video Coding (SVC) codec, for to judge the performance of the planned adaptation algorithmic rule. For mobile video streaming applications, Experiment results demonstrate the practicableness and effectiveness of the planned adaptation algorithmic rule that outperforms the present progressive adaptation algorithms

II. RELATED WORK

DASH has been a hot topic in recent years. There aremany commercial products which have implemented DASHin different ways, such as Apple HTTP Live Streaming andMicrosoft Smooth Streaming. Since the clients may havedifferent available bandwidth and display size, each videowill be encoded several times with different quality, bit rateand resolution. All the encoded videos will be chopped intosmall segments and stored on the server, which can be atypical web server. These small segments will be downloadedto the browsers' cache and played by the client (browser orbrowser plug-in). The video rate adaptation is performed atthe client side, which is also called the pull-based approach.The client will determine the quality version of the requestedvideo segment according to its current available bandwidth,resolution and the number of buffered unwatched segments.After the current segment is completely downloaded, therate adaptation algorithm will be invoked again for the nextsegment.

There is thorough work covering this subject [2]–[6]. The authors in [2] projected to estimate the knowledge live by a applied mathematics technique, which they took every the quality contribution and writing time of each section into thought. K.P. Moketal.presented a QoE aware DASH system [3]. Their formula estimates the gettable metric by looking out with the video info. Thus on keep the quality level as sleek as potential, their formula will switch the video quality version bit by bit and may commit to maintain the buffer level being stable. S. Akhshabi designed associate analysis technique in to check the performance of the many existing business DASH merchandise, like sleek Streaming, Netflix, and OSMF. In, T. Kupka projected to gauge the performance of live DASH below on/off traffic and tested four fully other ways to enhance the performance. In, the way to cut back inessential video quality variations using a looking out technique to identify the effective gettable metric was given. Within the authors designed the optimum rate adaptation formula for streaming.

In [6], T. Kupka proposed to evaluate the performance of liveDASH under on/off traffic and tested four different methods toimprove the performance.Following Table shows the summary of literature survey Related to video streaming.

In [4], [8], the authorsdesigned the optimal rate adaptation algorithm for streamingscalable video coding (SVC) over HTTP using MDP. WithSVC, each video frame is encoded into a base layer and severalenhancement layers. Higher video quality can be achievedwhen more layers are received. These works only consideredthe single-link case

Table 2.1 Summary of Literature Survey

Sr. No	Paper Title	Work done	Author Name	Publisher and Year
1	Performance of on-off traffic stemming from live adaptive segmented HTTP video streaming [6]	Authoranalyze the TCP performance of such live on-off sources, and we investigate possible improvements in order to increase the resource utilization on the server side.	T. Kupka, P. Halvorsen, and C. Griwodz	In IEEE LCN'12, 2012, pp. 401–409.
2	Flicker effects in adaptive video streaming to handheld devices[1]	TCP streaming guarantees this and provides lossless streaming as a side-effect. Adaptation by packet drop does not occur in the network, and excessive startup latency and stalling must be prevented by adapting the bandwidth consumption of the video itself	P. Ni, R. Eg, A. Eichhorn, C. Griwodz, and P. Halvorsen	In ACM MM'11, 2011, pp. 463–472.
3	Robust and efficient stream delivery for application layer multicasting in heterogeneous networks.[7]	However, if a node resides in the tree leaves, it cannot deliver the stream to its descendant nodes. In this case,Quality of Service (QoS) will be compromised dramatically. To overcome this problem, Topology-aware Hierarchical Arrange-ment Graph (THAG) was proposed	M. Kobayashi, H. Nakayama, N. Ansari, and N. Kato	IEEE Trans. Multimedia, vol. 11, no. 1, pp. 166–176, 2009
4	Transmission control for compressive sensing video over wireless channel	The wireless node needs to send the data out in a timely and energy efficient way. This transmission control problem is challenging in that we need to jointly consider perceived video quality, quality variation, power consumption and transmission delay requirements, and the wireless channel uncertainty	Xiang and L. Cai,	IEEE Trans. Wireless Commun., vol. 12, no. 3, pp. 1429–37, 2013.
5	Scalable Video Coding and Packet Scheduling for Multiuser Video Transmission Over Wireless Networks	Scalable video coding offers a flexible bit stream that can be dynamically adapted to fit the prevailing channel conditions. Advances in scalable video compression techniques, such as the newly adopted scalable extension of H.264/AVC, as well as recent advances in wireless access technologies offer possibilities for tackling this challenge.	EhsanMaani a , Peshala V. Pahalawatta b , Randall Berry a , and Aggelos K. Katsaggelos a a EECS Department, Northwestern University	Evanston, IL; b Image Technology Group, Dolby Labs, Burbank, CA.

III.MOTIVATION

To specialize in the problems connected with video streaming is that the psychological feature work of this analysis. On load equalization on server, less buffering, individuals have less time to buffering video and streaming depends. However, with the introduction of two internet server.

IV. PROPOSED WORK

Over multiple links as associate MDP disadvantage we have a tendency to formulate the video streaming technique. We've got a bent to stipulate several actions and reward functions for each state to appreciate swish and high quality video streaming. Second, we've got a bent to propose a depth-first fundamental measure search rule. The projected adaptation rule will take several future steps into thought to avoid playback interruption and reach higher smoothness and quality. Last, we've got a bent to implement a sensible testbed using associate robot phone and Scalable Video Coding (SVC) encoded videos to gauge the performance

The experiment results show that the projected adaptation rule is feasible for video streaming over multiple wireless access networks, and it outperforms the prevailing state-of-threat algorithms

In this paper we tend to projected dynamic adaptational streaming over HTTP has been projected. During a DASH system, multiple copies of pre-compressed videos with completely different resolution and quality area unit keep in segments. We tend to handle multiple request for the same video via load balancer streaming method as a reinforcement learning task. For every streaming step, we tend to outline a state to explain the present scenario, as well as the index of the requested section, the present out there information measure and different system parameters. A finitestate Andre Markov Decision Process (MDP) is sculpturesque for this reinforcement learning task. The reward operate is rigorously designed to contemplate the video QoS necessities, like the interruption rate, average playback quality, and playback smoothness, also because the service prices.

The planned RTRA are able to do similar perceived video quality as East Chadic. With the low-layer startup, the buffer is crammed terribly quickly. Once the buffer level reaches a definite level, forceful buffer level increment can bring a negative reward, and therefore the video quality are going to be upgraded to avoid such a penalty. During this approach, prime quality video streaming is warranted. As East Chadic is kind of greedy, it invariably tries to request the best potential layer, and East Chadic are able to do the similar quality as RTRA. Since layer amendment might bring a negative reward, the planned RTRA tends to keep up this video quality. Thus, the typical range of layer variations for RTRA is kind of small.

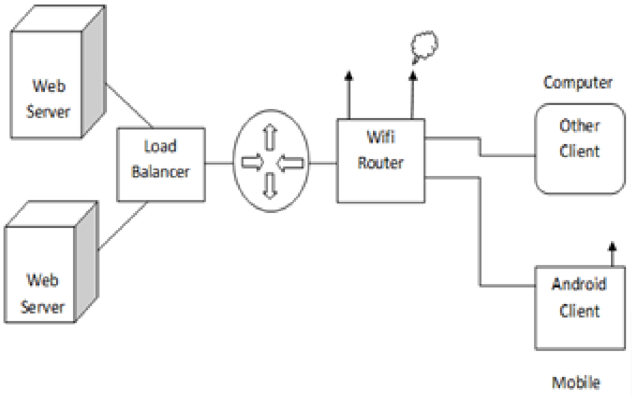

Fig1: Propose System Architecture

As mentioned before, East Chadic could be a greedy rule. One amongst the drawbacks of the greedy rule is that the smoothness is unnoticed because it solely considers the fast video quality. Thus, there's a big range of layer variations for East Chadic. With one WiFi link, as WiFi cannot sustain the primary improvement layer, typically it's to modify from the primary improvement layer to base layer so as to avoid playback freeze, which ends up in additional layer variations for RTRA SthanRTRA.

A. Adaptive Search Depth

Search depth is an important issue in our work. The searchdepth can determine how good the search result is, and alarger value of depth will achieve a better result. Meanwhile,with the increment of the search depth, the search time toobtain the action for a segment will be increased exponentially.Therefore, the search depth can be viewed as a trade-offbetween the video quality and the search time.Based on several preliminary experiment results, when thesearch depth D is larger than three, it will take more than twoseconds to obtain a decision on the test Android smart phone.Thus, the maximum search depth D_{max} is set to three. As theperceived video streaming fluency is generally considered asone of the most important QoS for the user, the search depthD is determined by the current queue length in our work.We divide the buffer queue into three regions, [0, q1), [q1, q2), and [q2, qL]. For each state, the search depth D is determinedaccording to its queue length q as follows:

$$D = \begin{cases} 1 \;\; if\ q\ \epsilon\ [0, q1) \\ 2 \;\; if\ q\ \epsilon\ [0, q2) \\ 3 \;\; if\ q\ \epsilon\ [0, qL) \end{cases}$$

When the queue length is low, there is a high probabilitythat a playback interruption may occur soon, and thus a shortsearch time and depth is preferred. When the queue length ishigh, there is sufficient time to search a deeper depth to obtaina better result.

Mathematical model of proposed work:

$$V^{\pi}(s) = E_{\pi}\left\{\sum_{n=0}^{N_T} r_n | s_n = s\right\}$$
$$= \sum_{s'} P_{ss'}^{a} \left[R^{a}(s) + \gamma V^{\pi}(s')\right],$$

R is the next reward of taking action an atstate s.
γ is the discount rate and $0 \le \gamma \le 1$.
The parameter γ makes a tradeoff between myopic video quality and future interruptions and variations

Real-time Search Algorithm

$$Q^{*}(s, a) = R(s) + \gamma \sum_{s'} P_{ss'}^{a}\ V*(s')$$

The calculation of the best long-term reward at state s0with action a.
Where,V∗(s') is the best long-term reward for state s

Algorithm 1 Real-Time Best-Action Algorithm
Procedure GETBESTACTION(s)
Initialize action ← -1, *Qmax* ← -∞
Generate all possible actions A(*s*) for state *s*
For all Action α € *A(s)* **do**
q ← REWARDSEARCH(s, α, 0)
if q >*Qmax***then**
*Qmax*q← action α←
end if
end for
return action
end procedure

procedure REWARDSEARCH (*s*, α, *d*)
q ← reward of (*s*, α)
if d ≥ D **then**
return *q*
end if
Generate all possible next states *S'* of (*s*, α)
for all *s'* from *S'***do**
Qmax ←∞
Generate all possible actions A'(s) for state s'
for all Action α' € *A'(s)***do**
Q_t ← REWARDSEARCH (*s'*, α', *d*+1)
ifQ_t>*Qmax***then**
QmaxQ_t←
end if
end for
q ← *q*+ *γPss'Qmax*
end for
return *q*
end procedure

B. Load Balancer algorithm

Load balancing is the practice of distributing a workload across multiple computers for improved performance. Load balancing distributes work among resources in such a way that no one resource should be overloaded and each resource can have improved performance, depending on the load balancing algorithm. Items such as network traffic, SSL requests, database queries, or even hardware resources such as memory can be load balanced.

Web traffic is carried out in HTTP (Hypertext Transfer Protocol), which also carries through to communication between servers working together to fulfill a request. HTTP is a plain-text format where actual words make up the messages running between computers. This create an overhead of traffic. AJP converts these messages into a binary code, reducing the amount of space taken by each message.

Speed between collaborating servers is a distinctive feature of the protocol. The process maintains permanent connections between servers to reduce the time taken to establish contact. The link is dedicated to a particular request and then maintained for the next request to be tunneled down, rather than each session causing a session to be created and broken.At present, there are 4 load balancer scheduler algorithms available for use:

a. Request Counting

(mod_lbmethod_byrequests)

**lbfactor* – Load factor

The idea behind this scheduler is that we distribute the requests among the various workers to ensure that each gets their configured share of the number of requests. It works as follows:

lbfactor is how much we expect this worker to work, or the workers' work quota. This is a normalized value representing their "share" of the amount of work to be done.

lbstatus is how urgent this worker has to work to fulfill its quota of work.

The worker is a member of the load balancer, usually a remote host serving one of the supported protocols.

We distribute each worker's work quota to the worker, and then look which of them needs to work most urgently (biggest lbstatus). This worker is then selected for work, and its lbstatus reduced by the total work quota we distributed to all workers. Thus the sum of all lbstatus does not change(*) and we distribute the requests as desired.

If some workers are disabled, the others will still be scheduled correctly.

for each worker in workers

ISSN: 2321-8169
311 - 316

workerlbstatus += worker lbfactor
total factor += worker lbfactor
if worker lbstatus> candidate lbstatus
candidate = worker
candidatelbstatus -= total factor

b. Weighted Traffic Counting

(mod_lbmethod_bytraffic**)**

The idea behind this scheduler is very similar to the Request Counting method, with the following changes: lbfactor is how much traffic, in bytes, we want this worker to handle. This is also a normalized value representing their "share" of the amount of work to be done, but instead of simply counting the number of requests, we take into account the amount of traffic this worker has either seen or produced.If a balancer is configured as follows:

c. Pending Request Counting

(mod_lbmethod_bybusyness)

This scheduler keeps track of how many requests each worker is currently assigned at present. A new request is automatically assigned to the worker with the lowest number of active requests. This is useful in the case of workers that queue incoming requests independently of Apache, to ensure that queue length stays even and a request is always given to the worker most likely to service it the fastest and reduce latency.

In the case of multiple least-busy workers, the statistics (and weightings) used by the Request Counting method are used to break the tie. Over time, the distribution of work will come to resemble that characteristic of byrequests.

d. Heartbeat Traffic Counting

(mod_lbmethod_heartbeat)

This modules load balancing algorithm favors servers with more ready (idle) capacity over time, but does not select the server with the most ready capacity every time. Servers that have 0 active clients are penalized, with the assumption that they are not fully initialized.

V. RESULTS

The graph shown is Fig. 2 describes the system for existing and propose system projected within the net state of affairs and system for video transfer and modified video transfer algorithms for time parameter.

So the proposed system consumes less Buffer time and offers higher results that is been shown within the graph.

The graph shown is Fig. 3the system for existing and propose system projected within the internet situation and within the standalone system for video transfer and transfer algorithms for time parameter. Graph shows difference of existing and proposed system in performance.

So the internet system consumes less performance and offers higher results that is been shown within the graph.

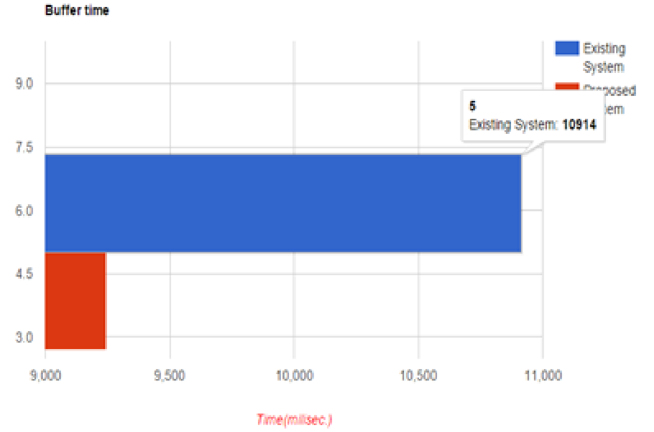

Fig 2.Existing System vs Propose System with bufferingtime

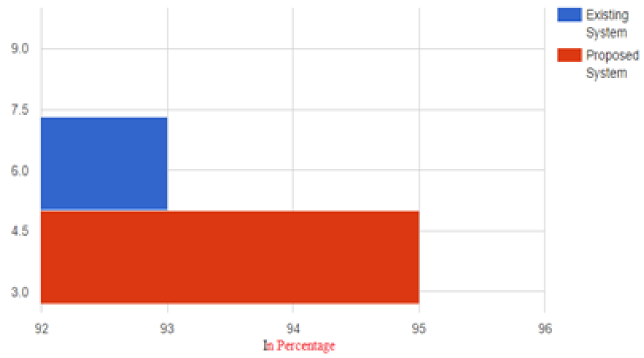

Fig 3.Existing System vs Propose System with Performance

VI. CONCLUSION

For video streaming over multiple wireless access networks we tend to project a fundamental measure adaptive best-action search formula. First, we have a tendency to tend to develop the video streaming method as Associate in Nursigned MDP. We have a tendency to tend to painstakingly design the reward functions to achieve smooth video streaming with prime quality.

Second, with the projected rate adaptation formula, we are able to solve the MDP to induce a sub-optimal answer in real time. Last,to gauge its performance and compare it with the progressive algorithms we have a tendency to tend to enforce the projected formula and conducted realistic experiments. The experiment results showed that the projected answer will do a lower startup latency, higher video quality and better smoothness. As a future analysis this experimental study will be done on another real time application like YouTube, through that speedily sharing and accessing of all video to word.

ACKNOWLEDGEMENT

The authors thank anonymous reviewers for their valuable. Authors would like to thank to all those who have helped in the construction of this study work.

REFERENCES

[1] P. Ni, R. Eg, A. Eichhorn, C. Griwodz, and P. Halvorsen, “Flicker effectsin adaptive video streaming to handheld devices,” *inACM MM’11*,pp. 463–472, 2011.

[2] K. Tappayuthpijarn, T. Stockhammer, and E. Steinbach, “HTTP-basedscalable video streaming over mobile networks,” in*IEEE ICIP’11*,pp. 2193–2196, 2011.

[3] R. Mok, X. Luo, E. Chan, and R. Chang, “QDASH: a QoE-aware DASHsystem,” in*ACM MMSys’12*, pp. 11–22, 2012.

[4] S. Xiang, L. Cai, and J. Pan, “Adaptive scalable video streaming inwireless networks,” *inACM MMSys’12*, pp. 167–172, 2012.

[5] C. Mueller, S. Lederer, and C. Timmerer, “A proxy effect analyis and fairadatpation algorithm for multiple competing dynamic adaptive streamingover HTTP clients,” in*IEEE VCIP’12*, pp. 1–6, 2012.

[6] T. Kupka, P. Halvorsen, and C. Griwodz, “Performance of on-off trafficstemming from live adaptive segmented HTTP video streaming,” in *IEEE LCN’12*, pp. 401–409, 2012.

[7] M. Kobayashi, H. Nakayama, N. Ansari, and N. Kato, “Robust and efficient stream delivery for application layer multicasting in heterogeneous networks” , *IEEE Trans. Multimedia, vol. 11, no. 1*, pp. 166–176, 2009

[8] Xiang and L. Cai,, “Transmission control for compressive sensing video over wireless channel”, *IEEE Trans. Wireless Commun., vol. 12, no. 3*, pp. 1429–37,2011