Multimedia Tools and Applications (2020) 79:4185–4198
https://doi.org/10.1007/s11042-019-07935-0

Maximizing the quality of service in distributed multimedia streaming in heterogeneous wireless network

S. Duraimurugan[1] · P. Jesu Jayarin[2]

Received: 1 November 2018 / Revised: 22 May 2019 / Accepted: 24 June 2019 /
Published online: 12 July 2019

Abstract
In a distributed multimedia streaming, QoS plays an important role while transmitting videos from multiple servers to the client. A good QoS is required to provide videos at high resolution and to achieve better video quality, packet loss and delay have to be reduced. In this paper, combined throughput for multiple flows is proposed to increase the quality of service in distributed multimedia streaming (DMSHN). This scheme is used when the user receives the video packets from multiple servers where each connection forms a different flow. Through this scheme, the location where two or more flows combine is identified and the bandwidth in the channel is divided equally between the different flows, thereby increasing the network performance and dynamically adjusting the transmission rates of all the combined flows.

Keywords Combined throughput · Distributed multimedia streaming in heterogeneous wireless network (DMSHN) · Quality of service (QoS) · Congestion level determination (CLD) · Fuzzy logic congestion controller (FLCC)

1 Introduction

1.1 Multimedia streaming in distributed environment

Any known system, either hardware or software that consists of multiple processing capabilities and storage is known as a distributed system. In these type of systems, the program is

✉ S. Duraimurugan
duraimurugans@stjosephs.ac.in

P. Jesu Jayarin
jjayarin@gmail.com

[1] Department of Information Technology, St. Joseph's College of Engineering, OMR, Chennai, Tamil Nadu 119, India

[2] Department of Computer Science and Engineering, Jeppiaar Engineering College, OMR, Chennai, Tamil Nadu 119, India

generally divided into sub parts which are executed simultaneously over multiple internet terminals. Distributed computing is a variant of Parallel Computing, where tasks are handled simultaneously. Figure 1 shows the overall structure of distributed environment.

Here, multimedia streaming applications runs on the distributed systems. Multimedia content like audio, video or image will be transmitted from more than one sender to the receiver in order to ensure faster transfer and also better performance. The different applications of multimedia streaming has been shown in Fig. 2.

The "streaming stored audio or video" content is already stored in one of the available sources, which is streamed to the client whenever the client provides the request. The "Streaming live audio or video" content is based on real-time transmission of audio or video information and in the meantime storing it in local storage as well. Then, the content will be streamed to client. If it happens in both direction (Sender and Receiver) simultaneously, it is called as "interactive audio or video" streaming. In multimedia applications, delay is sensitive, and at the same time the system should ensure minimum packet loss. Congestion Control mechanisms play a vital role, which ensures a good QoS to the client.

The main contribution of the paper is the quality during streaming services has been maximized by combined throughput in distributed multimedia streaming. The rest of the paper includes a survey of related contributions to our proposed work under Section 2, proposed work is discussed under Section 3, Experimental result and performance analysis are discussed in Section 4 and finally the paper is concluded in Section. 5.

2 Works related to DMSHN

To tackle the enormous amount of data being generated by RFIDs in IOT devices, the authors have proposed a distributed model that is built on top of an unstructured P2P overlay. Further approaches are proposed to prevent overheads as well [20]. An algorithm called SVC-QA has been proposed which can optimize the quality of streamed data by adapting the streaming quality at both the system level and client level. It works on the basis of buffering capacities and users' quality preferences [12].

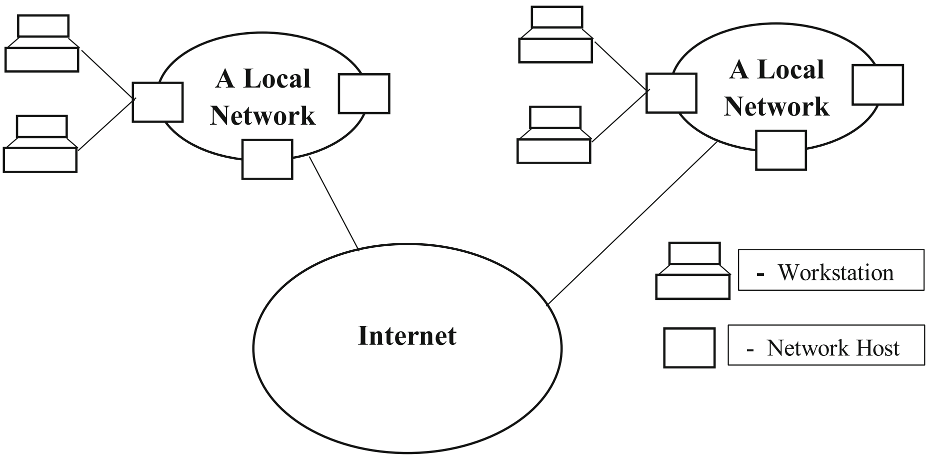

Fig. 1 Distributed Environment

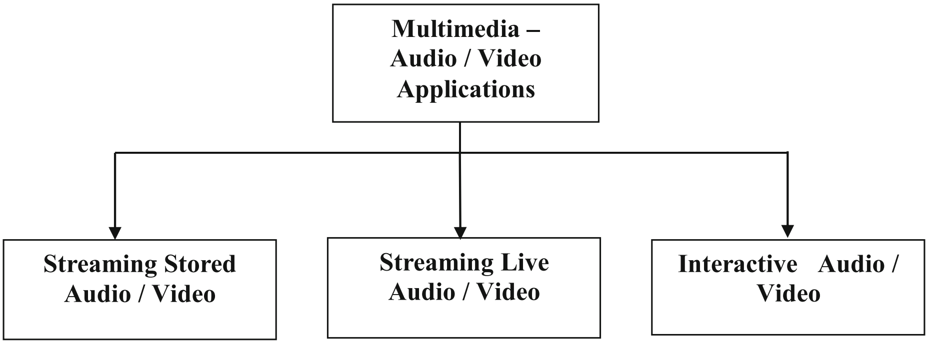

Fig. 2 Multimedia Audio / Video Applications

A Bayesian instantaneous end-to-end bandwidth change prediction model has been proposed that uses the inter arrival time of packets to identify congestion. The empirical data shows that it can detect congestion in less amount time (<200 ms) which is much superior to the existing techniques [10]. The author takes advantage of the dynamic adaptive nature of HAS services and adopts it in managed networks which hasn't been done before. This helps the networks to improve management policies which in turn can better coordinate the quality selection process [1].

An elastic stream processing method has been proposed for allowing distributed environments to cope up with large amount of real time streaming in this IOT big data era. This method drastically improves the scalability and maintainability of distributed environments [9]. A solution called EPOC is presented to enable better energy efficiency and streaming quality for multimedia content over heterogeneous networks. It properly uses energy-goodput tradeoff and multipath diversity to optimize the transmission process [21].

A TCP friendly method for multimedia streaming has been proposed. It is a rate control algorithm that changes the transmission rate and also tracks the media rate in a parallel manner which can provide a better throughput even in the long run [15]. A HAS system has been proposed where the congestion level is used by the client side nodes to optimize the requests that they make for multimedia data. A controller is then used to maximize the QoS and QoE, which uses the shared bandwidth from the client [3].

A method to properly calculate the quality variation in adjacent information for video streaming services using statistical techniques has been presented. It can adopt the standard quality metrics and existing models for QoE when performing its calculations [14]. A caching strategy that is variable length in nature for RTP streaming which can adapt itself based on the video segment's access patterns is discussed. This allows us to properly use the available storage conditions and improve the cache hit rate [22].

An optimization model along with a heuristic technique that can optimize the Multi-Video Stream bundles and increase playback quality and reduce the amount of playback interruptions has been presented. The buffer management system is shown to provide a better playback switching system [2]. The network errors in LTE services is focused upon and a method to effectively synchronize the broadcasting server and other players which reduces the latency present in the sever has been proposed. This is done by properly analyzing the buffer limits and modifying them accordingly [11].

A framework that utilizes elastic cloud computing services for enabling live streaming among heterogeneous devices which is based on crowd sourced streaming has been proposed.

The solution is also extended to conform themselves to regional constraints [8]. A streaming model that uses a controller that is based on feedback control and queuing model. This can improve the elasticity of virtual machines to improve the performance of virtual machines has been discussed [16].

A new flow and congestion control scheme called PLUS is developed for distributed multimedia systems. The scheme uses an adjustment mechanism to reduce the data loss in multimedia systems. This scheme is designed to increase number of PLUS-based streaming traffic and to be in agreement with TCP-based traffic. It predicts the future congestion in the bandwidth based on the current network situation [13, 17]. Topology aggregation and link summarization methods are proposed to acquire network topology and state information, optimization framework for flow based end to end QoS provision over multi domain networks, two distributed control plane designs by addressing the message between controllers for scalable and secure inter-domain QoS routing. The results show proposed distributed solution approach as a global optimum and works for larger networks [7, 18].

An analytical framework for optimal rate allocation based on observed available bit rate and round trip time for each network and distortion-rate characteristics has been proposed. The results when compared with heuristic AIMD-based schemes, media-aware allocation and optimal control benefits from the proactive congestion avoidance and reduces average packet loss rate [23]. A Priority Manager is used to dynamically compute the SDW and transmit data effectively over MANETs. It uses SCTP to overcome the issues caused by compulsory data ordering in TCP. The results show the improvement in transmission quality for data of all variants [19].

A modified version of Cat Swarm Optimization Server Selection (MCSO-SS) has been proposed to improve the quality of the transmitted multimedia. The method has been tested in heterogeneous environments and it shows better QoS and QoE [4]. A priority based proactive buffer management system has been proposed to overcome the irregularity present in the nodes of wireless networks. The buffer can vary itself based on the level of congestion in the given route and can also prioritize the information (packets) to ensure faster transmission of data [6]. A comparative analysis of multimedia streaming with respect to the quality of service in a heterogeneous network is provided based on the network conditions that are present in the last decade [5].

3 Proposed work

3.1 Overview of DMSHN

The proposed work deals with distributed multimedia streaming in heterogeneous wireless network (DMSHN). Here we consider three sender namely S0, S1 and S2 which is streaming the multimedia content to a single receiver R. The receiver R is connected with all the senders which means that there are multiple flows such as f0, f1 and f2. Multiple flows occur within the session and as a consequence senders are streaming the content to a receiver simultaneously. If there is a congestion, the system needs to regulate these flows. The forthcoming chapters deal with regulating these flows. Fig. 3 shows the topology of the proposed system.

This paper proposes a receiver driven protocol, so that the client can pull multimedia content from different senders simultaneously. The different senders stream the multimedia content to receiver based on the size of the buffer in the client side. So the streaming rate is

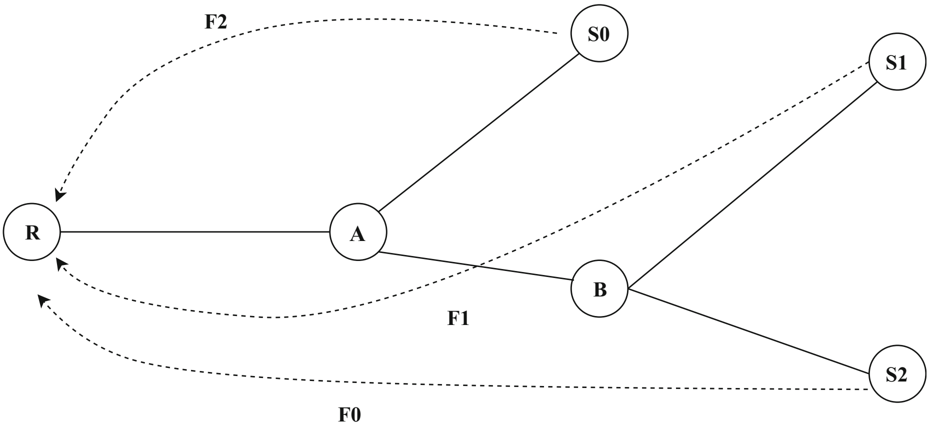

Fig. 3 Topology of DMSHN

decided by receiver which is intimated to the server. Therefore, it is more appropriate to implement the congestion control mechanism at the receiver side (as shown in Fig. 4).

Initially, the transmission rate is decided by network condition and according to the network condition, the packets transmitted from sender to receiver. At the client side, the incoming packets are monitored by the Congestion Level Determination (CLD) unit. Depending upon incoming packet dispersion, the congestion level and rate of change of congestion has been calculated. If high congestion is detected, the transmission rate of the various flows has to be changed. In this context, main challenge is to find the link which is responsible for congestion. Identify the link which is shared by more than one flows. So the system need to calculate new transmission rate for all the flows which is sharing the link. The transmission rate is calculated using Fuzzy Logic Congestion Controller. The input of FLC is the client side buffer threshold, congestion level and rate of change of congestion level which outputs a new streaming rate for the flows.

The main module of the proposed system is given in Fig. 5 and initially the system needs to identify bottlenecks in the network topology. If there is packet loss in the bottleneck then it identifies the flows that are sharing the bottleneck. The flows that are sharing the bottleneck are regulated by using a fuzzy logic congestion controller, due to which the system can achieve combined throughput. Finally the client side buffer is optimized using the Greedy Streaming to Threshold and Stop (GSTS) algorithm.

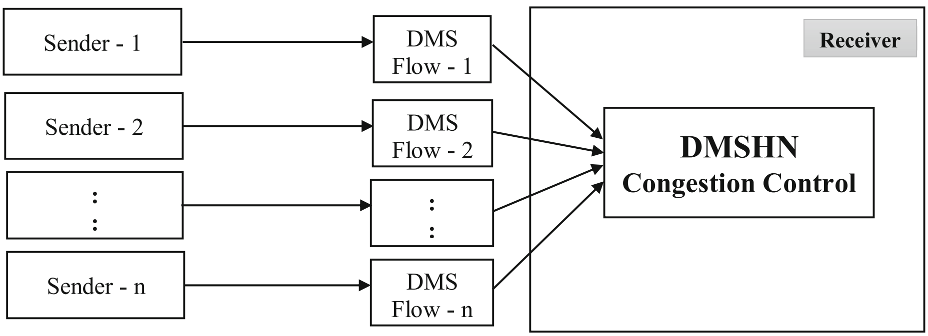

Fig. 4 Congestion Control Mechanism for DMSHN

3.2 Congestion location

3.2.1 Identification of bottleneck

Initially, the system analyzes the various flows to identify the lost packet's flow. That particular flow will be considered as $f_{i.}$ Then system needs to identify the reason for packet loss. Here, the system will have information about all the flows (F) and the links (L) that have participated in the task. The f_i is compared with all the DMS flows (F) to identify the correlated flows. After identifying the correlated flows, the least dominant link among the flows is considered as a bottleneck. The bottleneck will have more than one flow, so it cannot handle all the flows smoothly. According to the capacity of bottleneck, all the flows that are sharing the bottleneck need to be regulated which is discussed in Algorithm-3. The algorithm for the identification of the bottleneck is given in Algorithm-1.

Algorithm 1 Bottleneck ()

Input:

1) The flow whose packet is lost (f_i).

2) Set of all flows (F).

3) Set of all links (L).

Procedure:

For ($f_j = F_{1;}$ j<n; j++)

G=Correlation (f_i, f_j)

If G ==1

Congested flow $C_f \leftarrow f_i$

End loop

$C_l \leftarrow$ Least dominant link in the C_f

Output:

l $\leftarrow$ least dominant link which is reason for packet loss in f_i (Bottleneck)

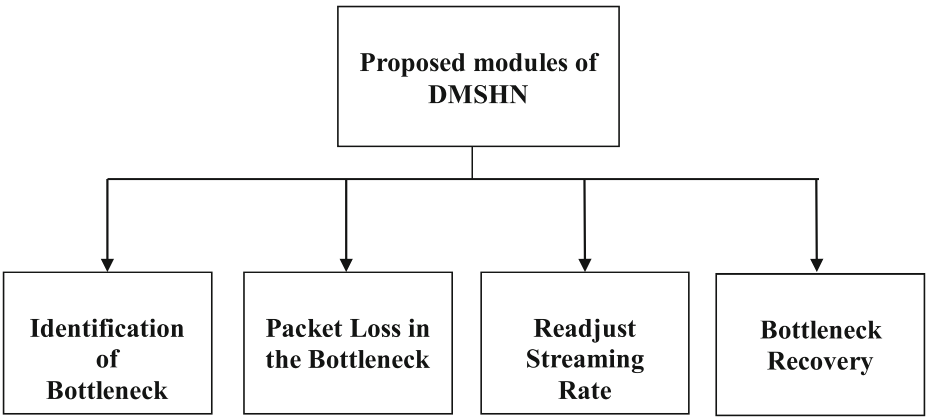

Fig. 5 Proposed modules of DMSHN

3.2.2 Packet loss in the bottleneck

The current congested link can identified by the recent packet loss. Here, the system is maintaining the queue which is a having history of the bottleneck detection record. The queue is updated dynamically based on the recent packet loss. C is a set of Current Congested Links and H is history of congested link in the system which is depicted in Algorithm-2. After updating the queue, C and H has been sent to next phase to estimate the new streaming rate of each flows.

Algorithm 2 PacketLossInBottleneck()

Input: 1) The flow whose packet is lost (f_i).

2) Queue which is maintaining history of bottleneck detection record (H).

3) Recent packet loss (h).

Procedure:

```
l ← Bottleneck (fi)
if | H | = = h then
dequeue (H) // taken out old bottleneck from the queue
end if
enqueue (H, l) // insert bottleneck into the queue
C ← l // current congested bottlenecks
```

Output:

Return C and H // Set of Current congested link (C) and History of congested link (H)

3.3 Throughput control

The proposed system uses a FLCC to control the throughput of the DMS flow. As it only achieves a fixed fraction of the throughput of each flow in the system, to ensure a fair flow for k DMS flows, the proposed algorithm controls the throughput of the DMS flow by setting it to 1/k of the conformant flow's throughput.

3.3.1 Readjust streaming rate

Before the streaming rate of the flows are readjusted, the maximum bandwidth utilization of every congested bottleneck at the particular session should be known. Current congested link is already

Table 1 Congestion level: *CN*

Value	Meaning
L	Low
M	Medium
H	High
VH	Very high
EH	Extremely high

identified using algorithm-1 and algorithm-2, then the flows that are sharing the link is extracted. Using this, the system can identify the number of flows in that particular link. Then Algorithm-3 can be used to obtain the maximum bandwidth utilization based on the number of flows.

Algorithm 3 ReadjustStreamingRate

Input:

1) C // Current congested link

Procedure:

Loop until "none of the link" then

C' = C //Copy all link from C to C' one by one

i=1

B=bandwidth of Congested link

l = C'

Loop until "none of the flow" then

N+=1

End loop

$BU_i=BU_i/N$

i++

End Loop

Output:

BU_i - Maximum bandwidth utilization of each flows in every bottleneck

3.3.2 Bottleneck recovery

The maximum bandwidth utilization of the bottleneck has been identified using Algorithm-3. Congestion level and rate of change in congestion level, client side buffer threshold, maximum bandwidth utilization as input given into Fuzzy Logic Congestion Controller. The congestion level and rate of change in congestion are depicted in Tables 1 and 2 which can be used for fuzzification. The fuzzier will calculate the new streaming rate for each and every flows.

Table 2 Rate of Change in Congestion: δCN

Value	Meaning
NVH	Negative very High
NM	Negative Medium
NH	Negative High
NL	Negative low
Z	Zero
PL	Positive low
PM	Positive medium
PH	Positive high
PVH	Positive very high

According to the fuzzification the sender transmit the frame to receiver side as depicted in Algorithm-4. At the same time the client side buffer threshold has to maintain for smooth video streaming is achieved using Greedy Streaming and Threshold Stop (GSTS) algorithm.

Algorithm 4 Bottleneck Recovery

Network Setup: Heterogeneous Wireless Network

1. Initial packet transmission based channel condition
2. If congestion on f_i then

    Input 1: Estimation of Congestion level - *CN*

    Input 2: Estimation of Rate of Change in Congestion - δCN

    Input 3: BU_i=ReadjustStreamingRate ();

    Input 4: Client Side buffer threshold

    newstreamingrate of flow f_i=Fuzzy(Input1, Input2, Input3, Input4)
3. Else

    Normal transmission
4. End if

4 Experimental results

4.1 Simulation environment

The simulations for the Distributed multimedia streaming in heterogeneous wireless network (DMSHN) algorithm were done using ns-2 with a dumbbell topology size of 1000 m ×1000 m. Non-Real time video was set as the traffic type. Independent source and destination nodes that were linked to a bottleneck link by means of independent access links where each link has a one way delay in the range 1 ms and 5 ms were given to each flow. To ensure proper transmission of video streams, DMSHN requires the videos to be TCP friendly with the flows and to be media friendly to the streaming application. An average bitrate of 1950 kb/s over a period of 600 s is used in the bitrate distribution.

4.1.1 Performance metrics

The performance of the proposed DMSHN algorithm is analyzed using measures such as End to End delay, Initial play-out delay, Loss event rate, Client buffer fill level and throughput.

End-to-end delay is the accumulation of processing and queuing delays in routers, propagation delays, and end-system processing delays.

Initial Playout delay - the receiver attempts to playout each chunk exactly *q* milliseconds after the chunk is generated. So if a chunk is time stamped at time *t*, the receiver plays out the chunk at time $t+q$, assuming the chunk has arrived by the scheduled play-out time $t+q$. Packets that arrive after their scheduled playout times are discarded and considered lost.

Throughput – It means the measurement of how many units of information a system can process that too in a given amount of time.

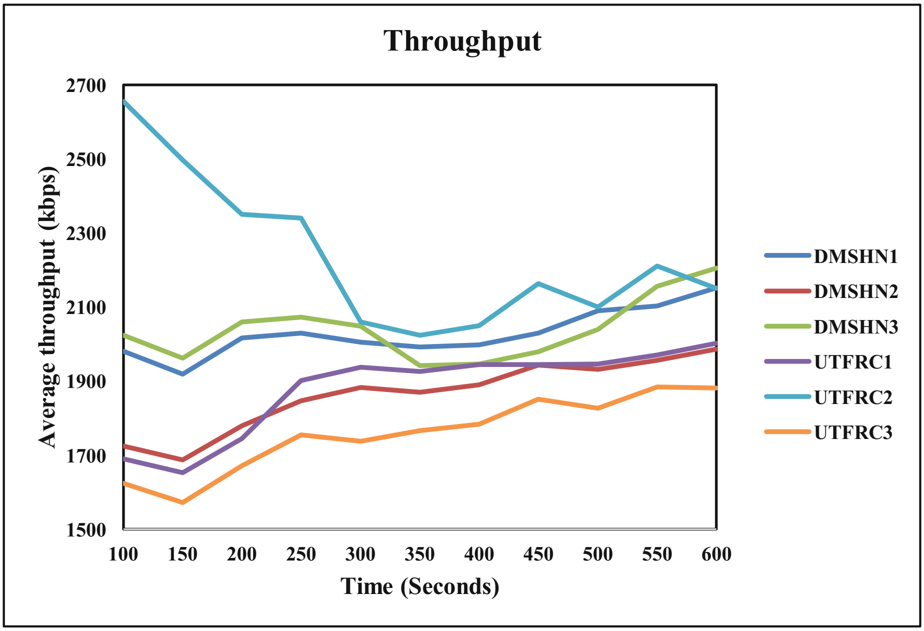

Fig. 6 Throughput analysis (DMSHN vs. UTFRC)

Client buffer fill level - It is the play-out time. This contains packets to be played in a given time t.

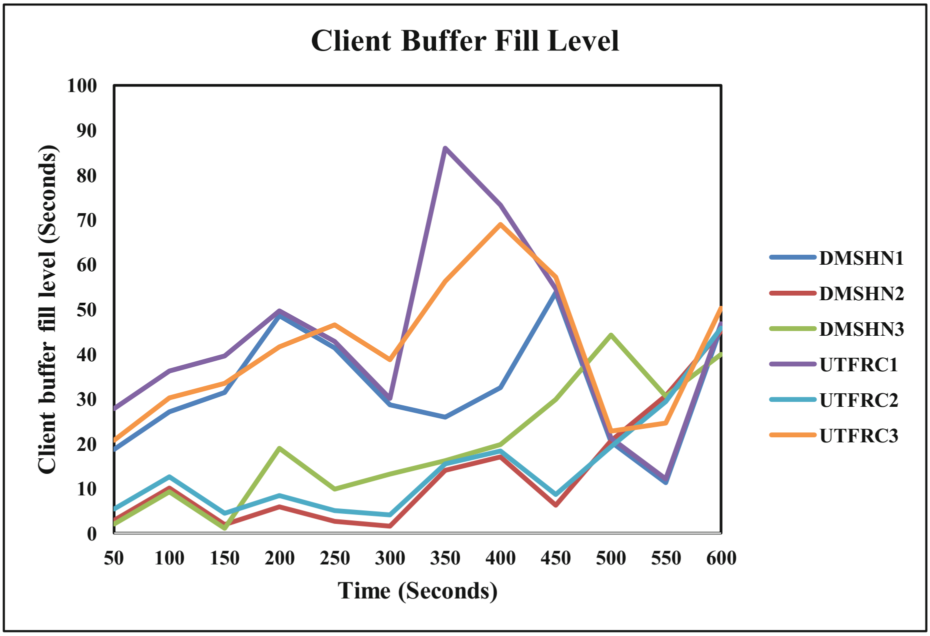

Fig. 7 Client buffer fill level analysis (DMSHN vs. UTFRC)

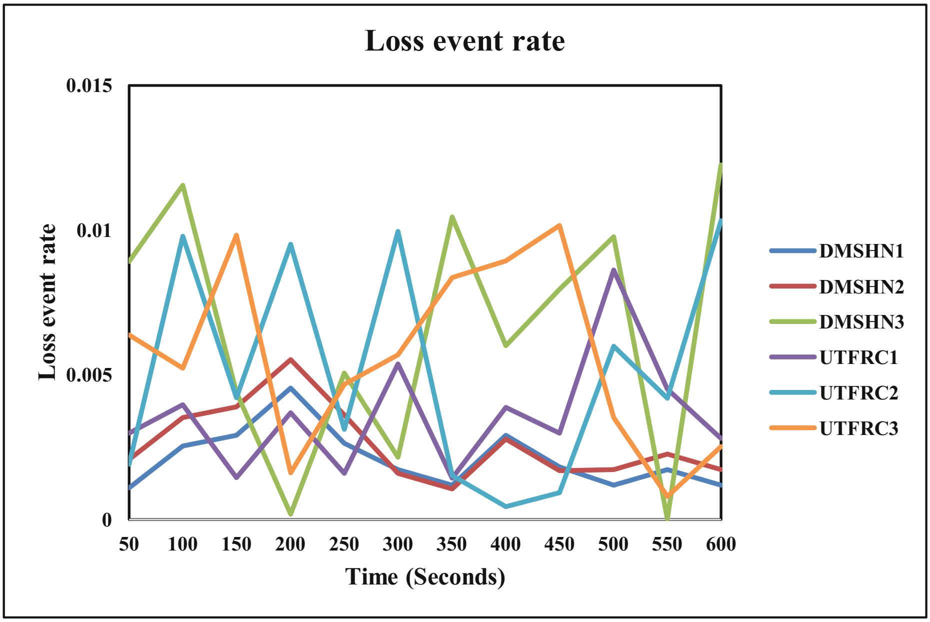

Fig. 8 Loss event rate analysis (DMSHN vs. UTFRC)

The loss event rate *p* is considered the fraction of loss events. This is generally seen in the number of packets sent over a long time interval.

A comparative analysis between the proposed, Distributed Multimedia Streaming in Heterogeneous wireless Network (DMSHN) and the existing Utility-driven TCP-friendly Rate Control (UTFRC) [15] was done and results are published below.

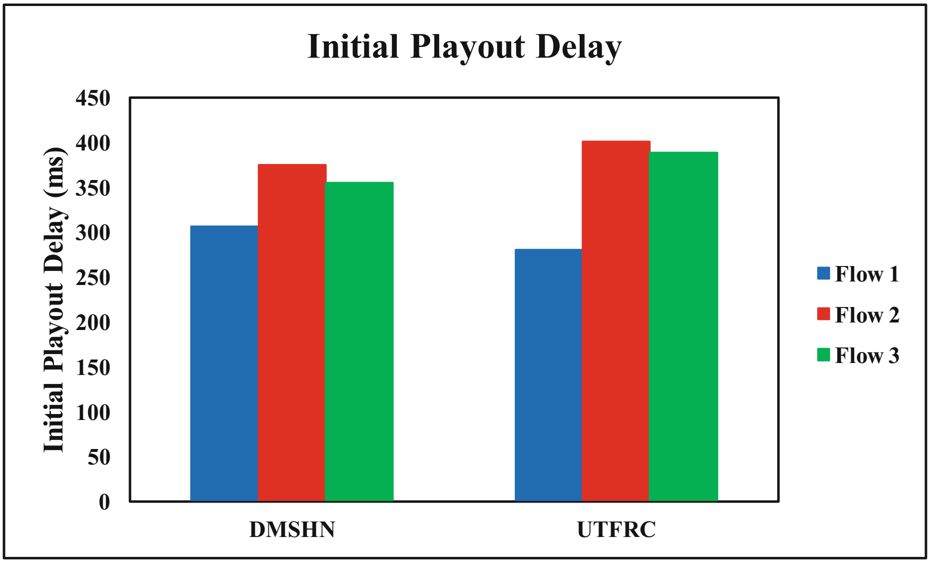

Fig. 9 Initial play-out delay analysis (DMSHN vs. UTFRC)

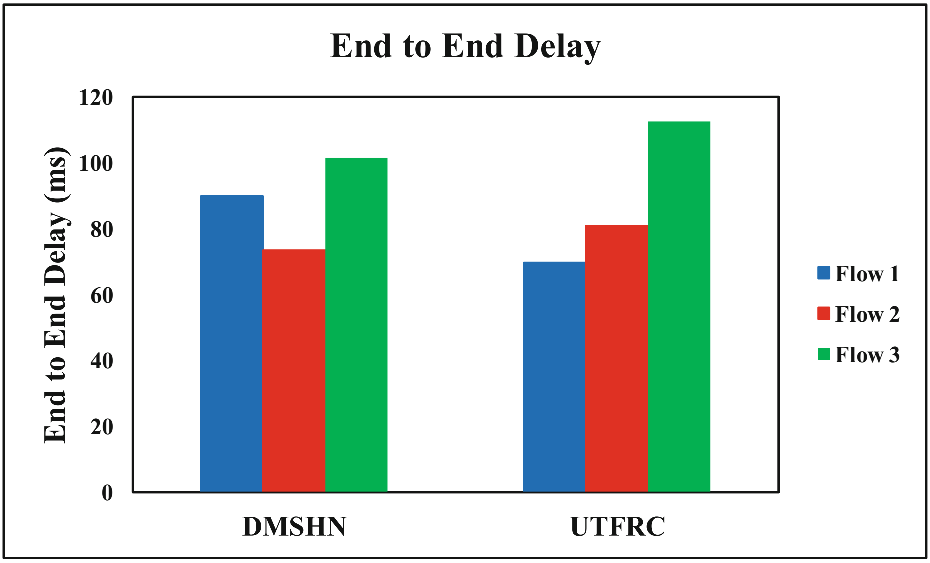

Fig. 10 End to End delay analysis (DMSHN vs. UTFRC)

4.2 Results and analysis

A comparison between DMSHN and UTFRC based on the throughput is shown in Fig. 6 and it is clear from the graph that, with increase in time t, the throughput increases, but the combined throughput deviation in DMSHN is less when compared to UTFRC.

Figure 7 depicts the variations in "Client buffer fill level" for both UTFRC and DMSHN and the level of client buffer fill in DMSHN is reduced in relative comparison with UTFRC even though the of certain local peaks are inevitable due to the random nature of networks .

A comparison between DMSHN and UTFRC based on "Loss event rate" is shown in Fig. 8 and even though the loss event rate for the algorithms are quite stochastic in nature, DMSHN provides acceptable levels than UTFRC.

Figure 9 depicts the "Initial play-out delay" for various flows using both UTFRC and DMSHN. When analyzing the graph, it is quite evident that, DMSHN provides a slightly better result on initial play-out delay when compared with UTFRC.

A comparison between DMSHN and UTFRC based on "End to End delay" is shown in Fig. 10 and DMSHN achieved good results when compared with UTFRC on basis of end to end delay as the delay is spread throughout various flows in the network whereas in UTFRC, the delay in concentrated in a single flow.

5 Conclusion

In this paper, we discussed about the proposed scheme DMSHN through which the Quality of Service is maximized while transmitting the data to the clients. The multimedia data pulled by the client from different clients are present in the network which is based on the buffer size of the receiver. The proposed technique combines the throughput for different flows of video packets that originate from different servers in the network, which is responsible for achieving a good QoS. This proposed scheme DMSHN provides a better quality of service in the distributed multimedia streaming in a heterogeneous wireless network by allowing the client to receive data from all the servers without affecting the other flows. The future enhancement of the proposed work is achieving the combined throughput in live audio/video applications using heterogeneous wireless networks.

References

1. Bouten N, Latré S, Famaey J, Van Leekwijck W, De Turck F (2014) In-network quality optimization for adaptive video streaming services. IEEE Transactions on Multimedia 16(8):2281–2293
2. Carlsson N, Eager D, Krishnamoorthi V, Polishchuk T (2017) Optimized adaptive streaming of multi-video stream bundles. IEEE Transactions on Multimedia 19(7):1637–1653
3. D'Aronco S, Toni L, Frossard P (2017) Price-based controller for utility-aware HTTP adaptive streaming. IEEE Computer Society 24:20–29
4. S. Duraimurugan, P. Jesu Jayarin, (2018) Maximizing quality of multimedia streaming services in heterogeneous wireless networks using optimal multi-server technique. Springer- Cluster Computing
5. Duraimurugan S, Jayarin P. J (2019) Analysis and Study of Multimedia Streaming and Congestion Evading Algorithms in Heterogeneous Network, Second International Conference on Intelligent Computing and Control Systems (ICICCS), 978–1–5386-2842-3
6. Duraimurugan S, Jesu Jayarin P (2018) Priority based proactive buffer Management for Aggregated Streaming in heterogeneous wireless networks. International Journal of Engineering & Technology 7(3.34):97–102
7. Egilmez HE, Tekalp AM (2014) Distributed QoS architectures for multimedia streaming over software defined networks. IEEE Transactions on Multimedia 16(6):1597–1609
8. He Q, Liu J, Wang C, Li B (2016) Coping with heterogeneous video contributors and viewers in crowdsourced live streaming: a cloud-based approach. IEEE Transactions on Multimedia 18(5):916–928
9. Hochreiner C, Schulte S, and Dustdar S. (2015) Elastic Stream Processing for Distributed Environments, IEEE Internet Computing
10. Javadtalab A, Semsarzadeh M, Khanchi A, Shirmohammadi S, Yassine A (2015) Continuous one-way detection of available bandwidth changes for video streaming over best-effort networks. IEEE Trans Instrum Meas 64(1):190–203
11. Lentisco CM, Bellido L, Pastor E (2017) Reducing latency for multimedia broadcast services over Mobile networks. IEEE Transactions on Multimedia 19(1):173–182
12. Li X, Veeravalli B (2015) A differentiated quality adaptation approach for scalable streaming services. IEEE Transactions on parallel and distributed systems 26(8):2089–2099
13. Mielke M, Aygun RS, Song Y, Zhang A (2002) PLUS: a probe-loss utilization streaming mechanism for distributed multimedia presentation systems. IEEE Transactions on Multimedia 4(4):561–577
14. Rainer B, Petscharnig S, Timmerer C, Hellwagner H (2017) Statistically indifferent quality variation: an approach for reducing multimedia distribution cost for adaptive video streaming services. IEEE Transactions on Multimedia 19(4):849–860
15. Sterca A, Hellwagner H, Boian F, Vancea A (2016) Media-friendly and TCP-friendly rate control protocols for multimedia streaming. IEEE Transactions on circuits and systems for video technology 26(8):1516–1531
16. Tolosana-Calasanz R, Diaz-Montes J, Rana OF, Parashar M (2017) Feedback-Control & Queueing Theory-Based Resource Management for streaming applications. IEEE Transactions on parallel and distributed systems 28(4):1061–1075
17. Vijayakumar K, Arun C (2017) Continuous security assessment of cloud based applications using distributed hashing algorithm in SDLC, Cluster Computing https://doi.org/10.1007/s10586-017-1176-x, Springer
18. Vijayakumar K, Arun C (2017) Automated risk identification using NLP in cloud based development environments, J Ambient Intell Human Computing, https://doi.org/10.1007/s12652-017-0503-7, Springer
19. Vivekananda G N, Chenna Reddy, PAydin Ilknur, (2019) A congestion avoidance mechanism in multimedia transmission over MANET using SCTP multi-streaming, Springer – Multimedia Tools and Applications
20. Wu Y, Sheng QZ, Shen H, Zeadally S (2013) Modeling object flows from distributed and federated RFID data streams for efficient tracking and tracing. IEEE Transactions on parallel and distributed systems 24(10):2036–2045
21. Wu J, Cheng B, Wang M, Chen J (2018) Energy-aware concurrent multipath transfer for real-time video streaming over heterogeneous wireless networks. IEEE Transactions on circuits and systems for video technology 28(8):2007–2023

22. Yang J, Yao Z, Yang B, Tan X, Wang Z, Zheng Q (2019) Software-defined multimedia streaming system aided by variable-length interval in-network caching. IEEE Transactions on Multimedia 21(2):494–509
23. Zhu X, Agrawal P, Singh JP, Alpcan T, Girod B (2009) Distributed rate allocation policies for multihomed video streaming over heterogeneous access networks. IEEE Transactions on Multimedia 11(4):752–764

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

S. Duraimurugan obtained his Bachelor's degree in Computer Science and Engineering from Noorul Islam College of Engineering, Tamilnadu, India and did his Master's degree in Systems Engineering and Operations Research from College of Engineering Guindy, Anna University, Chennai, Tamil Nadu India-25. Currently, he is an Associate Professor in the Department of Information Technology at St.Joseph's College of Engineering, Chennai, Tamil Nadu India-119 and a research scholar in Sathyabama Institute of Science and Technology, Chennai,Tamil Nadu,India-119. He has more than 10 years of teaching experience in Engineering College. His research interests include Congestion control in Multimedia Streaming.

Dr. P. Jesu Jayarin received his Ph.D. Degree from sathyabama University, Chennai, Tamil Nadu India-119. Currently, he is an Associate Professor at the Department of Computer Science and Engineering, Jeppiaar Engineering College, Chennai, Tamil Nadu, India-119. He has more than 15 years of teaching experience in Engineering College. He has published more than 25 papers in international conference and Journals. His current research interests are Computer network, Congestion control in Multimedia streaming, Mobile and wireless network.