Second International Conference on Smart Systems and Inventive Technology (ICSSIT 2019)
IEEE Xplore Part Number: CFP19P17-ART; ISBN:978-1-7281-2119-2

Optimized Cache and Forward Video Streaming in High Bandwidth Network

Duraimurugan S
Associate Professor
Department of IT
St. Joseph's College of Engineering
Chennai, Tamil Nadu, India

Muthukumaran K
Assistant Professor
Department of ICE
St. Joseph's College of Engineering
Chennai, Tamil Nadu, India

Leander Paul F
IEEE Student Member
Department of IT
St. Joseph's College of Engineering
Chennai, Tamil Nadu, India.

***Abstract* - As the technology continues to rapidly develop, the need for internet has also increased. Since all the new technological development and device require the user to be connected to the internet so as to reduce the burden on the user's device, the bandwidth required has increased tremendously. The introduction of virtual reality and augmented reality has further increased the bandwidth requirement for video streaming due to the large amount of data that would need to be delivered through the internet in a short period of time. The lack of any sophisticated methods to combat these challenges have made it difficult to stream high quality video while ensuring a good quality of service to the user. This paper provides a novel approach to overcome the difficulties faced while streaming high quality video content that requires a high bandwidth. There are two algorithms that are proposed in this approach, the first algorithm involves calculating the number of users who are viewing a video content and then determining the path through which the video packets would be sent to the user. The video content is then cached in the node which is closer to all the nearby video requesters, so that the user can get the video content from the cached data thereby reducing the resources that are needed to stream the video content. The second method involves prefetching the video content from the cache during the presence of a good channel condition so that the user can view the video even during bad channel conditions. The hidden Markov model is used to predict the channel conditions and to calculate the amount of video content that needs to be pre fetched to avoid overflow and under flow of video content data. The two approaches work together to provide video streaming services to a large number of users at a lower resource usage. Since the video packets are only transmitted to the cached location once, while there would be many users streaming the same video from that cache in that area, we can effectively reduce the resources needed by a large margin under certain conditions.**

Keywords - video streaming, bandwidth, Markov Model, Channel Condition, quality.

The use of internet has drastically grown in the recent times such that almost all of the devices in the world are connected to the internet. Many applications such as games, mail service, video stream and file downloading all the internet as the mode of transporting their data content. Among these applications, video streaming services is found to be the major cause of traffic in the internet due to its wide usage.

According to a report taken on the percentage of internet used in the recent years, it is found that there were almost a million minutes of video data being transferred across the internet every second. Moreover, the video streaming has also dominated 80 to 90 percent of the internet traffic. The report shows that the consumer Internet video flow, which excludes P2P video file exchanges, has taken 57 to 67 percent of the internet traffic in 2017, which in turn shows the important role that video streaming plays in today's world.

The characteristics of video streaming services include long duration, highly real-time, large volume of data, high quality of service and strict requirements for synchronization. This results in a huge traffic in the network, as a result of which congestion occurs in the network which in turn result in dropping the video packet and thereby reducing the quality of service experienced by the user. Therefore, there is a need for a sophisticated algorithm that is capable of managing the video streaming mechanism in both the senders' side and the client's side to ensure that there would be a good quality of service. Most of the applications either control the buffer to optimize the streaming effect or prevent the loss of video data packets by adjusting the video content quality according to the available bandwidth which is found to be not enough in reducing the network traffic and providing good quality video services.

Another solution to this problem is using cache to load or stream the video from nearby locations. In computer terms, cache is a high-speed data storage layer which stores the subset of video content, it is transient in nature, so that the future requests for the sane resource or video content are served up faster than it is possible by accessing the video's primary storage location. It allows to reuse the previously stored data efficiently. This ensures that the user can easily view the data from the cache thereby ensuring that the quality of service increases tremendously [9-13].

The next section (Section II) on this paper talks about the works that were done by other authors on video streaming. This is then followed by providing a detailed description of the proposed algorithms (Section III) – Cache location and channel condition prediction algorithm. The results and simulation of the proposed algorithms are then presented in the section (Section IV) that follows after. The final section (Section V) presents the conclusion derived from the proposed work.

II. EXISTING APPROACHES

Niklas Carlssonet al (2017) describes an approach by the user's would be able to enjoy a seamless streaming of multi-view video. The video packets for each angle of the video are bundled together and sent to the user. Since the video content for all the angle is send in parallel [2], the user can view the video in any angle whenever he wants.

Ahmed H. Zahran et al (2010) proposed a generic optimal framework to optimize the video streaming service through the internet or a network in heterogeneous systems. The framework achieves this by taking advantage of the diversity that is present in the cost and resource of the integrated access technologies and buffering ability of the streaming applications.

The framework also taken into consideration the different design challenges that are present which include limited processing capacity, handoff delay requirements and mobility randomness when it is optimizing the video streaming service [3].

Peng Zhao et al (2018) proposes a bitrate adaption scheme that allocates bitrate dynamically in a heterogeneous wireless network to maximize the user's quality of experience [4]. The video bitrate is selected based on the current state of the queue buffer and the throughput during real-time. Moreover, the scheme is also capable of balancing the trade-off between the user's quality of experience and the optimization of the buffer.

Sanam Sadr et al (2013) describes a new approach to allocate resource for video streaming services. A new model that captures the dynamic of a video streaming buffer has been introduced and the allocated spectrum in an optimized problem.

The final result of this approach is a linear problem that provides a trade-off between the buffer size and the bandwidth that is allocated for the video streaming services [8].

III. PROPOSED MODEL

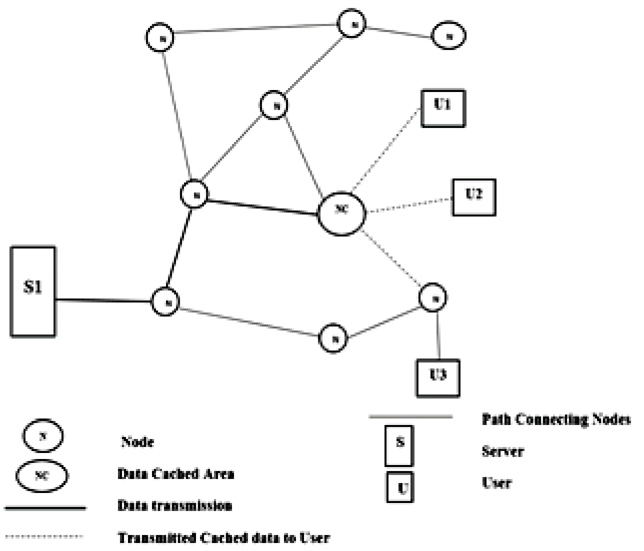

Fig. 1 Topology of proposed model (OCFVS)

The proposed model consists of two algorithm that are used in congestion with one another to ensure that video streaming service has a good quality of service. The first algorithm involves caching the video content so that it can be reused by the other users who are viewing the same video in the nearby locations. While the second algorithm involves using the Hidden Markov model to predict the channel condition of the path that is used for the video transmission. The two algorithms together ensure that the user can view the video content seamlessly in a lower time period since the video content is being loaded from the cached data and the path of video transmission channel is also found to be good. The two algorithms are explained below:

A. Cache Location

In this algorithm, the area or the location where the video content needs to be cached needs to be decided. This is done by locating the node that is closer to all the users who have requested for the said video content in a particular area. The node that is closer to all the nodes in the area is found using the modified AO* algorithm which can be used to find the nearest node to other users rather than the effective path to the destination.

From the Fig. 1 we can see how the algorithm finds the node that is close to all the users who have requested for a video content in a particular location and then the video content gets cached to that location so that it can be available to the user's nearby and thereby saving the time required to get the video content from the source location itself which would require more than double the resources for the same. The pseudo code for the algorithm is written below:

Algorithm 1 – Cache Location

```
Function PathToCacheLocation (Previous-node,
previous-distance)
   {
    Set shortest-distance to infinity
    For all nearby nodes
distance← Sum of distance between node and users
       If (shortest-distance > distance) then
shortest-distance = distance
If(previous-distance< shortest-distance) then
return current-node;
else
return current-node→PathToCacheLocation (current-
node, shortest-distance)
}
```

Using the above algorithm, we can find the path to the node that is close to all the users who has requested for the said video content. The function first starts from the server's location and then searches for all the nodes that are near to it. Then it locates the node that reduces the distance between itself and the other users and travels to that node. This step keeps repeating until the distance between the node and the user starts increasing instead of decreasing. Once the node that is close to all the other users are found, the server starts transmitting the video content to the said node to cache the video data packets.

B. Channel Condition Prediction

In this algorithm the channel condition of the transmission path is predicted so that the video content would be transferred during good channel condition which would ensure that congestion can be avoided if there exist, any congestion in the transmission path. The channel condition of the transmission path is predicted using the Hidden Markov model.

The Hidden Markov Model (HMM) is a Markov model in which the results of the model are based on the hidden and unobservable states of the problem. These states affect the final observable results of the prediction even though they are not visible.

The HMM process is represented by the set of hidden states and the set of observable states in the model. A sequence of state is then represented as the set of the state for each length, while the corresponding observation is a set of the corresponding observations for each length.

The HMM is formulated as

$$\lambda = (A, B, \pi) \tag{1}$$

'A' represents the transition array independent of time and continuously tracks the interference state probability following interference state. While 'B' is the observation array, independent of time and records observed probability generated by state. The initial state probability is represented by π.

The four inputs noise variance, average power received, channel bandwidth and the fast fading observed in the channel are provided as the input to the hidden state. These hidden states then uses these inputs to estimate the SINR. The Shannon Harley theorem then uses the SINR to estimate the channel condition in the transmission path as shown in the equation below.

$$CC_k = \log_2(1 + SINR_k) \tag{2}$$

Now that the channel condition (CC_K) is predicted using the HMM model, the results of the prediction are compared with the threshold condition of the channel. The channel condition is determined to be good if the threshold value of channel condition is found to be lower than the predicted channel condition while the condition of the channel is considered to be bad if the threshold value is found to be more than the predicted condition of the channel which is used to transmit the video contents. The video content is streamed to the user only during the presence of good channel condition while during the presence of bad channel condition the video content is stored in the buffer to be streamed when good channel condition returns.

Algorithm 2 – Channel Condition Prediction:

```
While(true)
{
   NV ← Noise variation observed.
   APR ←Average power received.
   CB ←Channel bandwidth observed.
   FF ←Fast fading observed
   CC_K ←HMM-Model (NV, APR, CB, FF)
CC_Th←Setting threshold.
   If CC_K<CC_Th then:
      Transmit video packet
       Break;
}
```

IV. SIMULATION AND RESULTS

A comparative analysis was done between the contributions of this research work and existing schemes.

Table 1. Provides information about the simulation environment used to implement both the algorithms that were proposed in this paper. The results for the proposed model (OCFVS) are compared with the Cross-Layer Optimization of Fast Video Delivery in Cache- and Buffer-Enabled Relaying Networks (FVDCB) algorithm [1].

TABLE I. Simulation Environment

Parameter	Values
Number of Mobile Nodes	1000
Topology Size	1000 × 1000 m^2
MAC	IEEE 802.11 and 802.16
Simulation Time	500 s
Traffic Type	Non-real time and Video
Rate	500,750,1000 and 1250 Kb/s
Buffer Size	500,750,1000 and 1250 Kb
Propagation	Two Ray Ground
Antenna	Omni Antenna

Rate variation:

The effect the change of streaming rate has on the various parameters of video streaming such as delay, delivery ratio and throughput are used as the basis of comparison in the results provided by the proposed algorithm and the FVDCB algorithm below

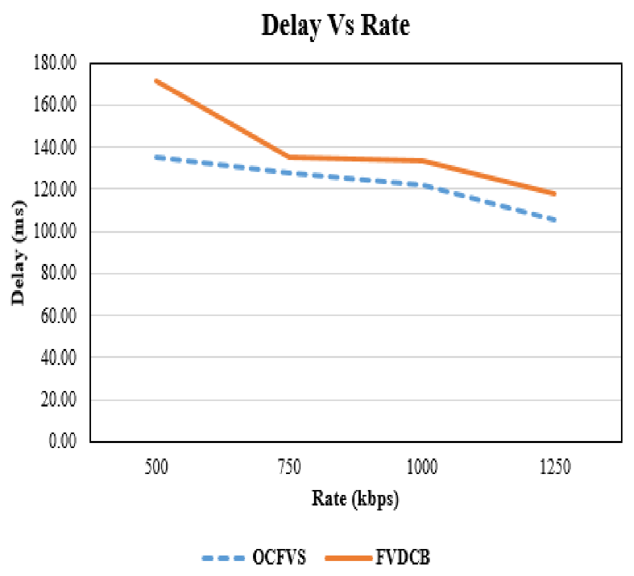

Fig 2. Impact of rate on delay (OCFVS Vs FVDCB)

The Fig. 2 shows the comparison done between the proposed algorithm and the FVDCB algorithm in terms of delay of video packets to reach the users location when the streaming rate of video packets are increased. The results show that there is a reduction of 13% in delay when the proposed algorithm is used.

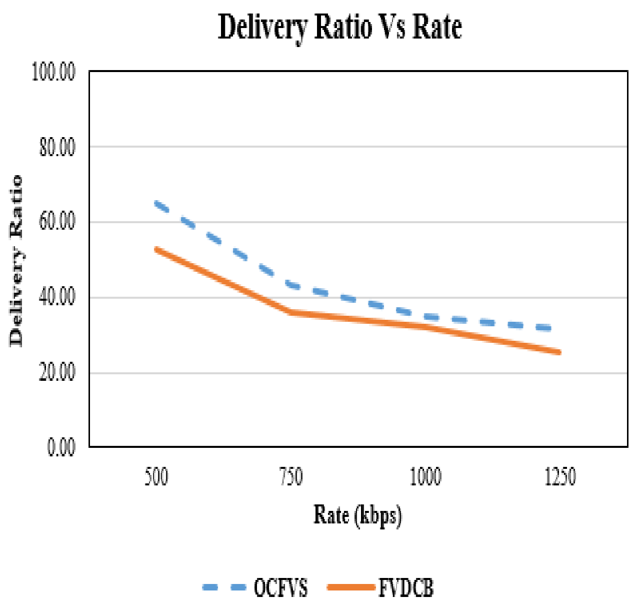

Fig 3. Impact of rate on delivery ratio (OCFVS Vs FVDCB)

The Fig. 3 denotes the comparison of results between the proposed algorithm and the FVDCB algorithm. The effect on delivery ratio, defined as the number of video packets received at the user's end divided by the total number of video packets send from the server, when the streaming rate is changed.

The results prove that the proposed model has an increase in delivery ratio by 15% when compared to the existing algorithm.

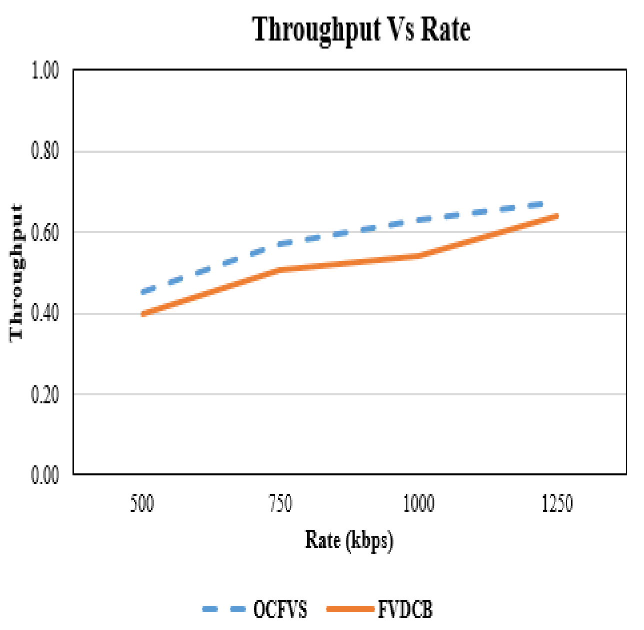

Fig 4. Impact of rate on throughput (OCFVS Vs FVDCB)

The Fig. 4 shows the effect on throughput when the streaming rate is changed. The results of the proposed model are compared to the FVDCB algorithm and proves that the proposed algorithm provides a 10% increase in the throughput than its counterparts.

Buffer Variation:

The effect the change in the size of the buffer has on the various parameters of video streaming such as delay, delivery ratio and throughput are used as the basis of comparison in the results provided by the proposed algorithm and the FVDCB algorithm below.

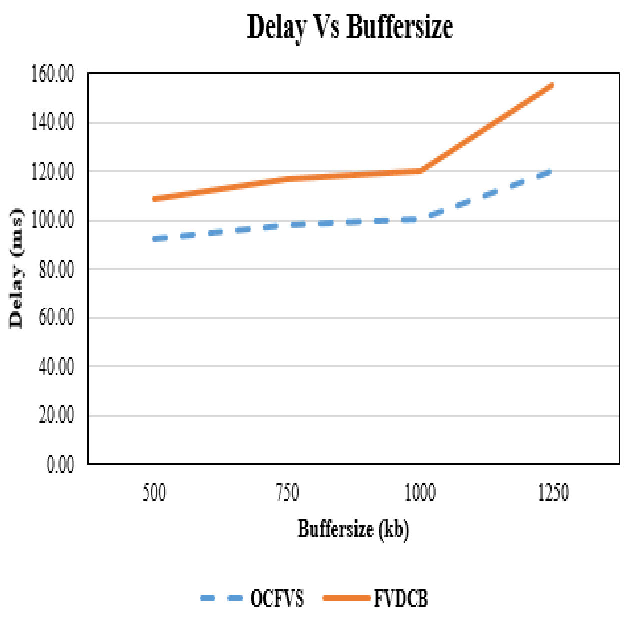

Fig 5. Impact of Buffer size on delay (OCFVS Vs FVDCB)

The proposed algorithm is compared with the FVDCB algorithm on the effect of delay in receiving the video packets in the users end when the size of the buffer is changed. The Fig.5 that the proposed algorithm has a reduced delay by 20% in comparison to the existing algorithms.

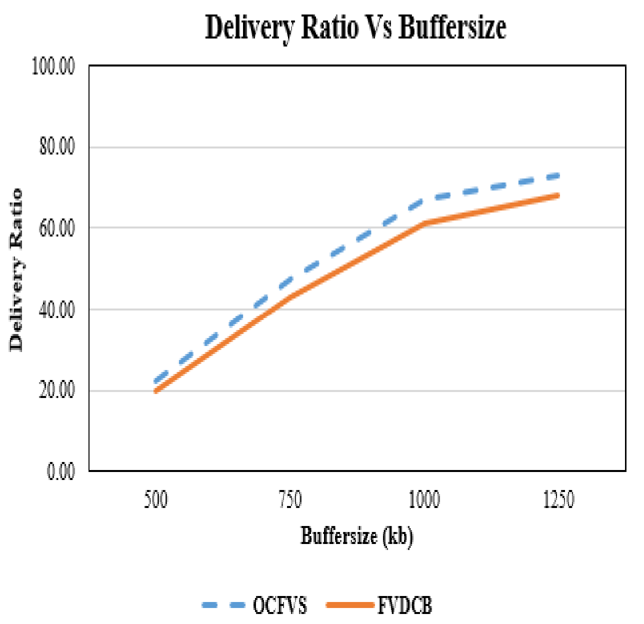

Fig 6. Impact of Buffer size on delivery ratio (OCFVS Vs FVDCB)

The Fig. 6 denotes the results of the effect on delivery ratio when the size of the buffer is changed between the proposed algorithm and the FVDCB algorithm. The results prove that the proposed algorithm provides a 8% increase in delivery ratio compared to its counterparts.

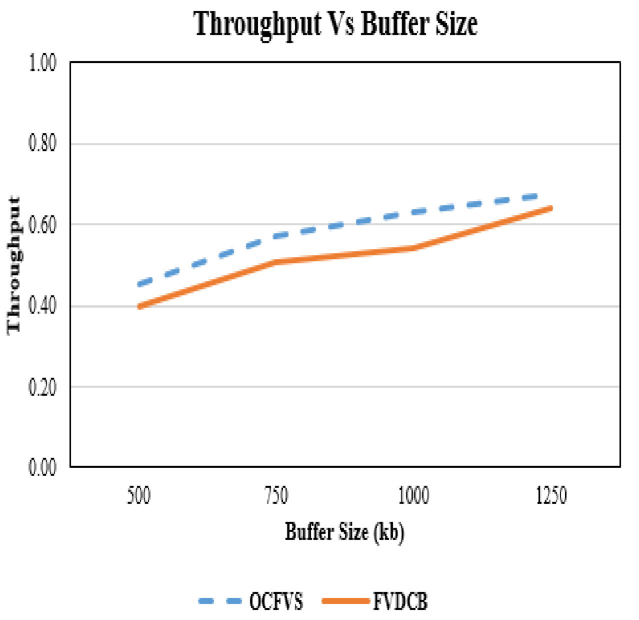

Fig 7. Impact of Buffer size on throughput (OCFVS Vs FVDCB)

The results of the proposed algorithm in comparison with the FVDCB algorithm is shown in the Fig.7. The comparison is based on the effect, the change in the size of the buffer has on the throughput of the video packets. The results prove that the proposed algorithm is 11% more effective when compared to the existing algorithm.

V. CONCLUSION

Delivery of seamless video streaming service has become the challenge faced in current times as even though the amount of data that can transferred through the internet has increased tremendously, it still is unable to keep up with the increase in the network traffic. To overcome this challenge, two algorithms are proposed in this paper. The first algorithm – Cache Location is used to locate the node which is close to all the other users who have requested for the same video content in the nearby location and the video content is cached in this location. This ensures that the video content would be streamed from this location rather than from the original location of the video content thereby reducing the resource needed and the time required to get the video content. The second algorithm that is proposed is the channel condition prediction algorithm which is used to stream the video content only when there is good channel condition meaning the absence of congestion in the network. These two algorithms work together to provide results that has better results in comparison with the existing models. The future work includes the creation of a buffer management scheme that would ensure that the video content would be efficiently streamed.

REFERENCES

[1] Lin Xiang et.al, Cross-Layer Optimization of Fast Video Delivery in Cache- and Buffer-Enabled Relaying Networks, IEEE Transactions on vehicular technology, vol. 66, no. 12, December 2017.

[2] NiklasCarlsson et.al, Optimized Adaptive Streaming of Multi-video Stream Bundles, IEEE transactions on multimedia, vol. 19, no. 7, July 2017.

[3] Ahmed H. ZahranTh et.al, reshold-Based Media Streaming Optimization for Heterogeneous Wireless Networks, IEEE Transactions on mobile computing, vol. 9, no. 6, June 2010.

[4] Peng Zhao et.al, Buffer Data-Driven Adaptation of Mobile Video Streaming over Heterogeneous Wireless Networks, IEEE Internet of things journal, vol. 5, no. 5, October 2018

[5] YashuangGuo et.al, Buffer-Aware Streaming in Small-Scale Wireless Networks: A Deep Reinforcement Learning, IEEE Transactions on vehicular technology, vol. 68, no. 7, July 2019.

[6] Wei Zhang et.al, Adaption Resizing Communication Buffer to Maximize Lifetime and Reduce Delay for WVSNs, IEEE Access, Volume 7, 2019.

[7] SeonWook Kim et.al, Design and Implementation of Display Stream Compression Decoder With Line Buffer Optimization, IEEE Transactions on consumer electronics, vol. 65, no. 3, August 2019.

[8] Sanam Sadr et.al, Anticipatory Buffer Control and Resource Allocation for Wireless Video Streaming, https://arxiv.org/abs/1304.3056v1, IEEE Globecom 2013.

[9] Duraimurugan et. al, Maximizing the quality of service in distributed multimedia streaming in heterogeneous wireless network, Multimedia Tools and Applications, Springer, July 2019.

[10]Duraimurugan et..al,Maximizing quality ofmultimedia streaming services inheterogeneous wireless networks using optimal multi-servertechnique, Cluster Computing, Springer, January 2018.

[11]Duraimurugan et..al,Analysis and Study of Multimedia Streaming andCongestion Evading Algorithms in Heterogeneous NetworkEnvironment, Second International Conference on Intelligent Computing and Control Systems (ICICCS 2018), March 2019.

[12]Smys, S., G. Josemin Bala, and Jennifer S. Raj,"Self-organizing hierarchical structure for wireless networks." In 2010 international conference on advances in computer engineering, pp. 268-270. IEEE, 2010.

[13] Smys, S., and Jennifer S. Raj. "A self-organized structure for mobility management in wireless networks" Computers & Electrical Engineering 48 (2015): 153-163.