International Journal of Communication Networks and Information Security
2024, 16(1), 6221
ISSN: 2073-607X,2076-0930
https://ijcnis.org/
Research Article

Adaptive Scheduling Algorithm for Live Video Streaming in P2P Network

Satyanarayana Reddy Marri [1*], Chenna Reddy P [2]

[1.] Research Scholar, Department of CSE, JNTUA Anantapuramu, Andhra Pradesh, India
[2.] Professor, Department of CSE, JNTUA Anantapuramu, Andhra Pradesh, India
*Corresponding Author: satyam.marrijntu@gmail.com

Citation: S. R. Marri and C. R. P, "Adaptive Scheduling Algorithm for Live Video Streaming in P2P Network," *International Journal of Communication Networks and Information Security (IJCNIS)*, vol. 16, no. 1, pp. 186-197, Apr. 2024.

ARTICLE INFO

Received: 15 November 2023
Accepted: 6 April 2024

ABSTRACT

The goal of this paper is to develop a network code that will help improve the performance of live streaming systems. The major factors that influence the network bandwidth is the buffer map updating delay between the peers, The proposed model deals with the scheduling algorithm which identifies the required number of packets to the various parent nodes. The scheduling algorithm also considers the encoding and forwarding mechanisms in the network. The proposed method improves the streaming continuity in the network and reduces redundancy. The experimental results claim that the proposed algorithm improves the video quality and minimizes the rate of redundancy. It also provides a high packet rate compared to the existing algorithms.

Keywords: P2P Communication Scheduler, Live Streaming, Innovative Packets, Network.

INTRODUCTION

With the rapid development in the field of live streaming, the network will consume more internet bandwidth. More people are interested in streaming online for high quality videos [1]. Many authors are interested in developing the coding techniques for minimizing the internet bandwidth used by customers. One of these coding techniques is HEVC, which will minimize the bandwidth usage in high quality videos [2]. The researchers also concentrated on the 3D techniques to interact with the network. The ability to provide the communication in 3D videos by using the internet also required more bandwidth. This type of mechanism will help people to interact by using the new communication technology [3], [4], [5].

Different methods are employed to improve the quality of the video streaming by combining both the channel coding and video coding [6]. But, the research on multi-source networks is still in weak condition. The authors in [7] provide the different techniques for single source P2P networks.

The researchers face the difficulty in enhancing the P2P network transmission efficiency, where the existing model utilizes the full bandwidth of the network connections. In [8], the authors address the P2P communication model transmission efficiency. It involves a novel transmission mechanism for communication. The experimental results proved the efficiency with respect to the delivery ratio of the packets in the video streaming.

In [9], the authors proposed the P2P communication methods for live streaming and file transmission which is more efficient in sending and receiving the data. These systems are highly cost-efficient and can be developed with considerable capacity. The first generation of P2P networks used multicast trees, which consist of one or more peers where data packets are instantly pushed to the lower level of the network after receiving a new packet. Conversely, swarm-based systems [10] are easy to implement and more adaptable than traditional networks. In swarm-based schemes, each peer updates its buffer-map upon receiving a new video block and selects which segment to fetch from its neighbour based on the packet's schedule. Representative pull schemes used in this system include LayerP2P and Cool streaming.

An optimized algorithm for push-based networks is introduced in this study, which can effectively reduce overhead and improve transmission efficiency. The proposed packet scheduling algorithm guarantees that each packet sent to every receiver is encoded as per the network requirements, eliminating the need to wait for the buffer-map update. It also considers the minimum number of packets required to be sent to each receiver. In comparison to earlier swarm-based push schemes, the new algorithm demonstrates significantly improved efficacy.

LITERATURE REVIEW

Packet scheduling is a critical factor for P2P networks. Although [11] proposed an optimized method to handle distributed networks for error recovery, their approach only considers network encoded packets and not sub-streams. Consequently, this method can negatively impact error recovery and coordination of the peer's network.

[12] proposed a scheduling algorithm that prioritizes packets to enhance the performance of distributed networks. However, their method only addresses the delay problem in swarm-pull networks.

In order to achieve a more efficient and cost-effective streaming solution, [13] proposed a proactive rate-based selection algorithm and a bandwidth estimation. Unfortunately, their method requires frequent buffer-map exchanges, which can increase the communication overhead.

The authors in [10] proposed a hierarchically clustered P2P streaming system called HCPS, which can support the optimal streaming rate with a short delay. It is easy to implement in practice. The system's hierarchy is composed of small size clusters that are grouped together to retrieve video data from the server. By carefully managing the uploading capacities of each cluster, the system can be efficiently utilized.

There are various efforts being made to improve the resource utilization of P2P live streaming. One of these is a two-phase swarming scheme [11] that involves the quick diffusion of fresh content to the entire system. This method can be used to distribute the available content to the peer-to-peer network in the first phase. One of the most common approaches to improve the performance of P2P live streaming is by implementing network coding [12]. However, this method can't guarantee the maximum streaming rate. In [13], the authors present a novel algorithm that can improve the streaming rate by converging the packet forwarding requirements of the network. They also study the various factors that affect the performance of the stream.

By using the above packet scheduling mechanism, our approach suggests a straightforward transmission mechanism that estimates the total number of packets to be scheduled in advance to decrease the frequency of buffer-map exchange requests. This mechanism can be combined with other scheduling methods to deliver superior video quality through P2P networks (table 1).

Table 1. Comparative Analysis of the Scheduling Algorithms in P2P Systems

Scheduling Algorithms	Description	Advantages	Disadvantages
Round Robin [18]	Each peer in the network takes turns sending data packets to other peers in a sequential order.	Simple and easy to implement Fair distribution of bandwidth among peers	May not consider network conditions Higher latency due to sequential order
Randomized [19]	Peers are selected randomly to send data packets to other peers.	Provides a degree of randomness which can help in load balancing Simple to implement	May lead to uneven distribution of bandwidth Limited control over network congestion
Tit-for-Tat [20]	Peers reciprocate by sending data packets to other peers who have previously sent data packets to them.	Encourages cooperation among peers Helps in maintaining fairness and balance in bandwidth usage	Vulnerable to free-riders who do not reciprocate May not adapt well to dynamic network conditions
Content-Aware [21]	Peers prioritize sending data packets based on their content relevance or popularity.	Improves user experience by prioritizing popular or relevant content Can reduce network congestion by directing traffic towards fewer popular peers	Requires knowledge of content popularity or relevance May lead to biased distribution of bandwidth

Scheduling Algorithms	Description	Advantages	Disadvantages
Hybrid Approach [22]	Combines multiple scheduling algorithms based on network conditions, content popularity, and peer behavior.	Offers flexibility and adaptability Can leverage the strengths of different scheduling approaches	Complex to implement and manage May require significant overhead for coordination and decision-making

System Model

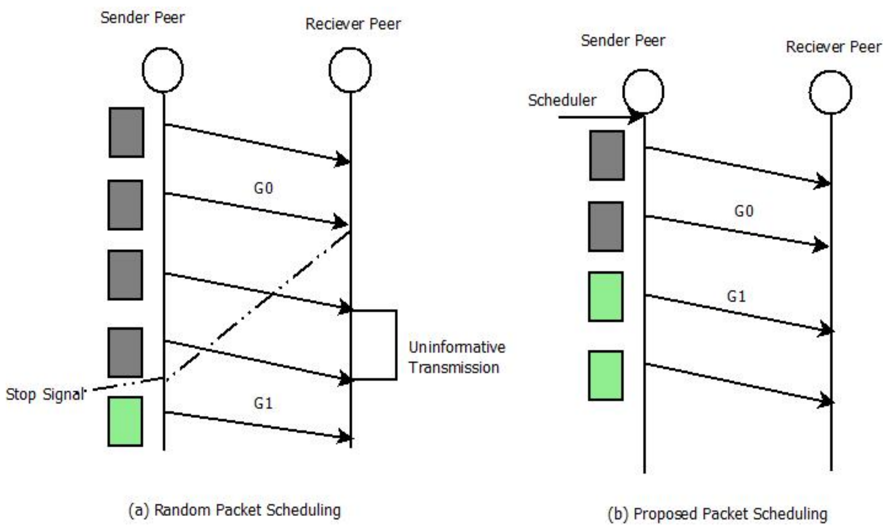

Figure 1. Proposed Packet Scheduling Method

Figure 1 shows the proposed packet scheduling method, whereas in Figure 1(a), it is observed that two different video streaming are given as G1 and G2. Non-informative transmissions become inevitable when a stop signal lately emerges in the P2P overlay, resulting in the immediate discarding of these packets since the receivers have already encoded the original ones. In Figure 1(b), the proposed method considers the swarm-p2P streaming network optimally employs the available transmission opportunities. A swarm-p2P streaming network consists of two components: an overlay network and a scheduler. The overlay network is responsible for establishing a network for transmitting and receiving packets, whereas the scheduler manages the peers in the system. The primary objective of the overlay network is to form a cluster of interconnected peers that are linked through the tracker server. Whenever a peer group contacts the tracker server, it obtains information about the upload and download bandwidth of its neighbours.

The objective of the packet scheduling module is to identify peers that have incomplete generations and transmit the necessary combinations. The Packet Scheduling module can be split into three parts. An optimal algorithm for scheduling packets considers the number of available slots during the priority window's shift. A selection mechanism chooses the appropriate packet at random throughout the process. In case the client nodes' upload bandwidth cannot accommodate the request, an adaptive push scheduling algorithm may be utilized to skip certain generations in the streaming source. Each generation of the Packet Scheduling module handles different algorithms for the client and streaming source.

The streaming source divides each video packet generation into multiple blocks, each lasting t seconds. These network-coded packets are then transmitted to other users in the P2P network. A fresh network-coded packet is generated by dividing it into several units using Eq.1. The selected packets from the priority window are then forwarded to the next node in the queue. As the playback window changes, the priority period becomes a moving window.

$$x = \sum_{i=1}^{n} w_i y_i \tag{1}$$

Where the coded packet is denoted with x, the received packets are represented with [w1, w2...wn] and [y1;y2; ...yn] are random coding coefficients. When the receiver gets x, it determines if the packet is innovative or non-innovative. If it is not, it is discarded. A client node can reconstruct the generation after it has obtained a sufficient number of independent packets [16].

Adaptive Packet Scheduling Model for P2P Network

The objective of the Adaptive scheduling algorithm is to enhance the average video quality for a given network connection in cases where the upload bandwidth is insufficient to support full-rate video transmission. All packets are encoded, and if the receiver node cannot get enough to decode them, they will become useless. This issue can be solved by implementing an adaptive push algorithm, which will skip some generations to ensure that all the other ones are successfully encoded. The streaming source performs the algorithm. Once the skipped generation has been determined, the source transmits the new streams to end-users. Algorithm 1 shows the adaptive scheduling algorithm in the priority period. In the proposed algorithm, the node i upload capacity is denoted with Qi. The node average upload capacity in the network is given in Eq. 2.

$$Q_{avg} = \sum_{j=0}^{m} Q_j \Big/ (m) \tag{2}$$

Where m is the number of nodes in the network, until the current generation the cumulative available bandwidth is denoted with Q^c_g..

Algorithm 1. Adaptive Scheduling Algorithm in P2P Networks

PRS-> Priority Region Start

PRE-> Priority Region End

Input: k-> streaming packets

Q_{avg}-> Average upload capacity of the node

Output: vector V

For g <- PRS to PRE do

$Q_g^c = Q_{g-1}^c + Q_{avg}$

if $Q_g^c > k_g$ then

$V_g = k_g$

Else

$V_g = 0$

End

$Q_g^c = Q_g^c - V_g$

Store the element V_g in to vector V

End

Return V;

Streaming Source Node Packet Scheduler

The goal of the streaming source scheduler is to ensure that the optimal amount of packets are sent to the client nodes during each generation. The distribution's number of scheduled packets is set by V_{ig}^k in generation g. The streaming source's allocation of bandwidth is determined according to the upload bandwidth of the client node. The current push factor β is the maximum amount of bandwidth that a streaming source can upload at a speed greater than steaming. For each peer node i, the number of scheduled packets is calculated using the following Eq. 3.

$$V_{ig}^k = [\beta \frac{Q_i}{\sum_{j=0}^{m-2} Q_j} V_g] \tag{3}$$

When a new generation is added to the priority period, it uses a selection algorithm to determine the optimal number of packets that should be sent to the client nodes. The outcomes of this process are recorded locally. The selection algorithm subsequently compares the scheduled packet count for each node during the transmission opportunity with the number of packets actually transmitted. If the video packet size of the first generation is less than that of the second generation, an encoded version is delivered to the client peer, and the log of transmitted packets is incremented.

Client Node Packet Scheduler

The client node's packet scheduling algorithm considers the required number of packets to transmit video. The goal of the optimization process is to reduce the number of non-innovative packets that are sent by the client node. It will allow the client node to allocate more bandwidth for innovative packets during the priority period. This enhancement can enhance the video quality and delivery ratio.

The algorithm decides that how many packets need to be transferred to the receiver node at each generation. Computing the optimal amount of packets before sending them to client nodes during each generation in a P2P streaming system typically involves a combination of factors, including network conditions, peer availability, and content popularity. This problem is then solved using Integer Linear Programming (ILP). E_{ij} represents the non-negative integer where the number of packets from client i is transferred to client j. All E_{ij} of k form the matrix E. The factor ϑ serves as an aggressive measure against the impact of non-innovative transmission and loss rate, while the variable V_{jg}^{k} represents the total packets received by j^{th} node from the source node. To minimize the non-innovative packets, we formulate g within the priority period, subject to certain constraints outlined in Eq. 4.

$$E = \arg\min_{E_{ij}} \sum_{i=0}^{M-2}\sum_{j=0}^{M-2} E_{ij} \tag{4}$$

$$subject\ to\ \sum_{j=0}^{M-2} E_{ij} \le Q_i \quad for\ \forall i \quad (i)$$

$$\sum_{j=0}^{M-2} E_{ij} \ge [\vartheta(V_g - V_{jg}^{k})] \quad for\ \forall j \quad (ii)$$

$$E_{ij} \le V_{ig}^{k} \quad for\ \forall i\ and\ \forall j \quad (iii)$$

$$E_{ii} = 0\ for\ \forall j \quad (iv)$$

Rule (i) stipulates that the total number of scheduled packets transmitted by sender node i to other nodes cannot exceed its maximum upload capacity. To decode g, rule (ii) stipulates that all the nodes at the receiver's end require $[\vartheta(V_g - V_{jg}^{k})]$ encoded packets. Rule (iii) is designed to ensure that the scheduled packets from node i to node j are fewer than the packets received by node i from V_{jg}^{k}. This condition guarantees the availability of adequate innovative packets at node i. Rule (iv) ensures that the peers only forward the packets to the other peers.

The sender updates the client schedule once the priority level has progressed. It then predicts the total number of scheduled packets that will be allocated to each receiver. The client scheduling algorithm updates its schedule every generation during the priority period. The generated results are stored in the local record. During each node i transmission opportunity, a selection algorithm evaluates the number of packets previously transmitted to receiver nodes j and E_{ij}. In case the number of previous packets transmitted to receiver j is smaller than E_{ij} for generation g, node j's information and generation g's information are added to a vector. A random element is picked from the vector, and all packets received at node i for this generation g are combined and forwarded to the intended receiver.

METHODOLOGY

The experimental analysis involves two primary phases: constructing the overlay and implementing the block schedule algorithm. However, our focus in this experiment is solely on the block schedule algorithm. The main aim of the proposed algorithm is to establish a fair comparison between various networks, which we implemented using the NS2 simulator [17]. The network comprises 50 nodes, where each end node randomly connects to its neighbours. As is typical in peer-to-peer systems studies, the upload bandwidth of peers is often the limiting factor. the complexity of the proposed algorithm is ($O(N^k)$), where N represents the number of peers in the network and k represents the degree of polynomial complexity due to optimization objectives. Table 2 shows the parameters considered for the network establishment.

The video being streamed is using the H.264/AVC encoding and is in the Paris sequence, with a bit rate of approximately 128 Mbps. Once all the packets have been decoded, the resulting video quality is 42.12 dB. Each generation consists of 8 frames, which equals a series of pictures lasting 0.32 seconds, and the number of packets allocated for each generation varies according to the encoding algorithm. On average, 64 packets are allocated for each generation.

To avoid the playback period from ending, the packets must reach the nodes within 2.45 seconds. The streaming source must provide enough bandwidth to satisfy 20% of the end users. To compensate for the low random network coding rate, the aggregate factor will create approximately one redundant packet per generation. The size of the actual video blocks is 200 Mb. The coefficients for the network-encoded packets are stored in the header and transmitted to the video packets, and the size of the video packets depends on the number of encoding units in each generation. To achieve a full streaming rate, the upload bandwidth must be at least 1.26 times the current rate.

Table 2. Parameter Settings

Parameter	Value
Video stream	Paris sequence (H.264/AVC)
Stream bit rate	128 Mbps
Generation size	8 frames
Video block size	200Mb
Packet header size	128 bytes

RESULTS AND DISCUSSION

Figures 2, figure 3 and figure 4 provide valuable perspectives on the performance and dynamics of the P2P networks during the simulation. In Figure 3, the x-axis depicts simulation rounds, and the y-axis displays the number of dead nodes. There is a steady increase at first, but a big change happens after roughly 200 rounds, resulting in a spike of dead nodes. Surprisingly, the dead nodes settle at a high constant after around 700 rounds, marking a critical moment in the simulation where network conditions become difficult.

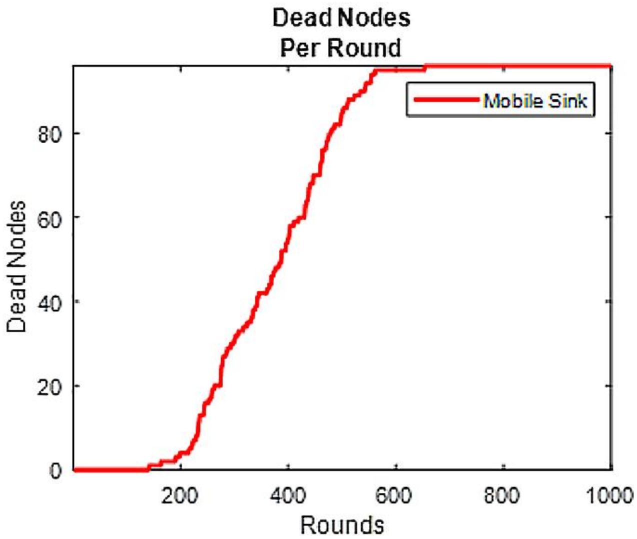

Figure 2. Dead Nodes Per Round

Moving on to Figure 3, the graph depicts the dynamic nature of node activity during the experiment by representing Operating Rounds per Round. Following an initial increase in operational nodes, a fall is noticed after about 200 rounds, which continues until around 700 rounds. The functioning nodes have reached a relatively low constant value at this time, reflecting changes in node availability and activity during the experiment.

Figure 3 shows Packets Received per Round, with the x-axis representing simulation rounds and the y-axis representing the number of received packets. The data shows a consistent increase in received packets per round, peaking at roughly 640 rounds. After this time, the P2P maintains a steady high value of 4000 packets per round, indicating continued communication and data transmission.

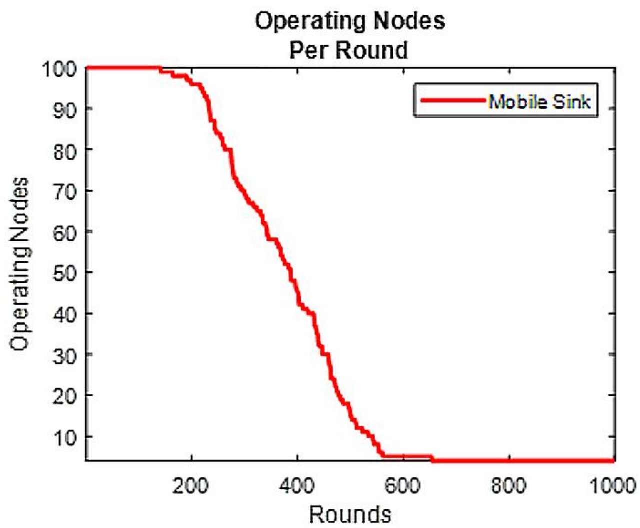

Figure 3. Operating Rounds Per Round

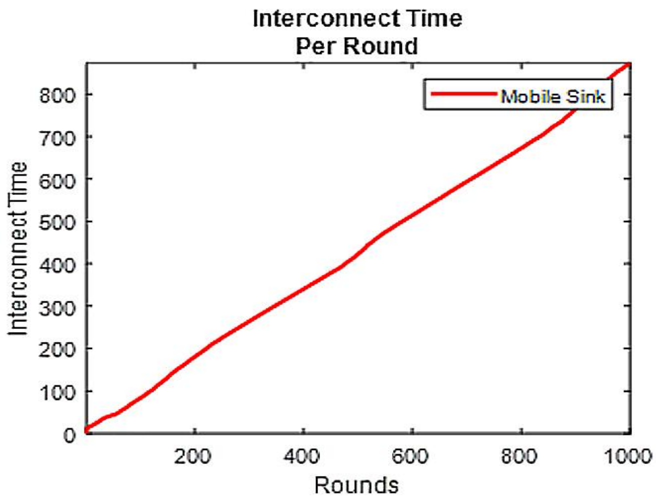

Figure 4. Interconnect Time Per Round

Figure 4 depicts Interconnect Time per Round, which has a linear relationship with simulation rounds. This implies that nodes maintained a consistent level of interconnectivity throughout the simulation, showing the network's resilience in sustaining communication ties.

The proposed system is compared with the ARND method [14] and ADS method [15]. The performance of the proposed and existing method is evaluated using the Average video Quality, Delivery Ratio, Redundancy ratio and Innovative packet rate. Figure 5 evaluated three different methods to compare their performance under varying upload rates of the peers. The proposed scheme outperformed the other two approaches in cases where the upload rate was insufficient to deliver the video, and continued to perform better as the upload rate increased. Conversely, the existing schemes were unable to deliver enough packets for successful decoding. The proposed scheme employs an adaptive scheduling algorithm that skips frames to ensure all packets reach their intended destination before the playback deadline.

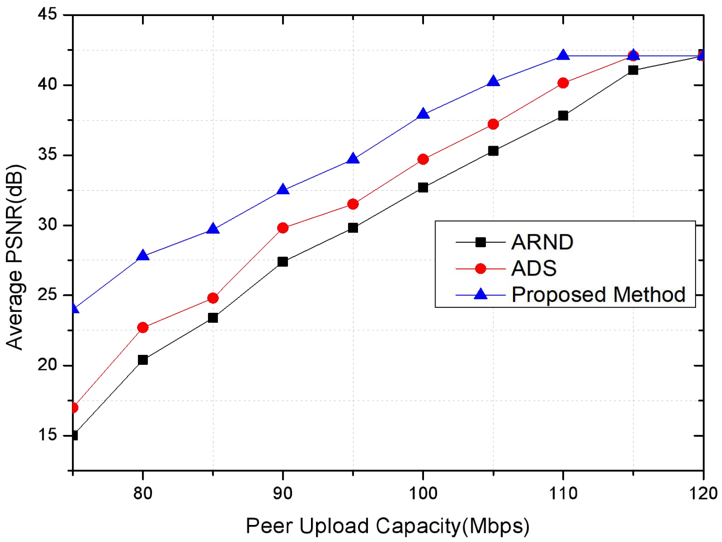

Figure 5. Average PSNR Vs Peer Upload Capacity

Figure 6 presents an analysis of the performance of three different methods over a range of peer upload rates. The delivery ratio, which reflects the average packet rate received compared to the streaming rate, is crucial for determining video quality in non-scalable video delivery. The proposed scheme significantly outperforms the ARND algorithm in delivering high-quality videos, achieving a better delivery ratio even in low-bandwidth networks. The issue with the ARND algorithm is that it loses accuracy in packet pushing when the buffer update interval is 200ms, as it has a longer update interval.

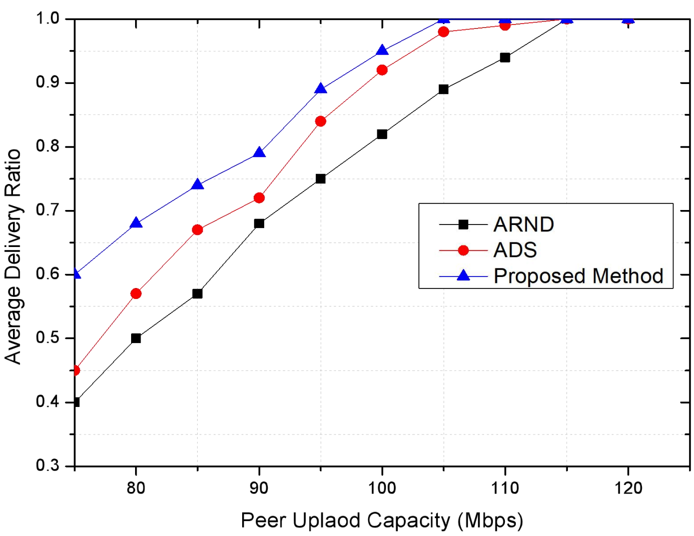

Figure 6. Average Delivery Ratio Vs Peer Upload Capacity

The results presented in Figure 7 indicate that the ADS approach achieves a packet ratio of between 0.1% and 0.3%, while the proposed scheme has a redundancy rate ranging from around 2% to 4%. However, with an increase in upload bandwidth, the redundancy packet ratio of the ARND method increases to 19.57%. The low number of linear dependent packets sent by the sender is mainly due to the short update interval between the buffer map and the receiver's buffer. This prevents the process from continuing even if the interval exceeds 200 milliseconds. The proposed scheme is more competitive in real-world systems as there is a low probability of sending buffer-map updates within 50ms.

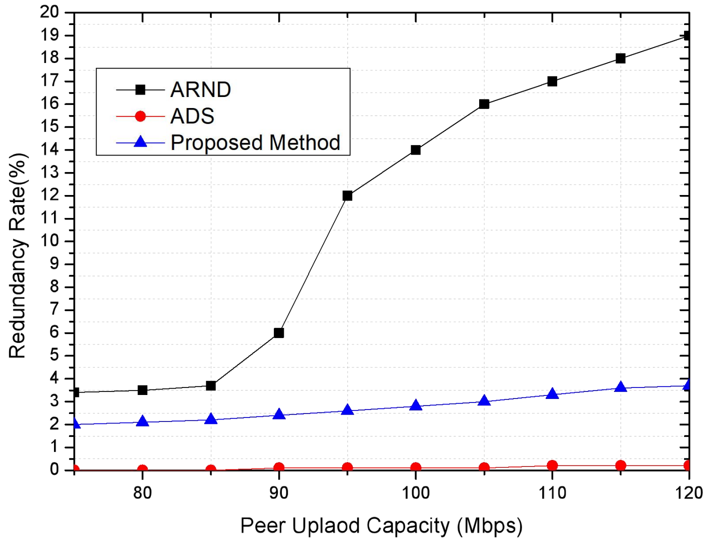

Figure 7. Redundancy Ratio Vs Peer Upload Capacity

The results presented in Figure 8 show the average packet rate of each method. It also takes into account the varying upload bandwidth of the peers. After conducting a comprehensive analysis of the proposed algorithm, we discovered that it offers a higher packet rate. Extending the buffer mapping update interval to 200 milliseconds can result in a significant increase in the packet redundancy ratio. This can lead to a decrease in packet rate innovation.

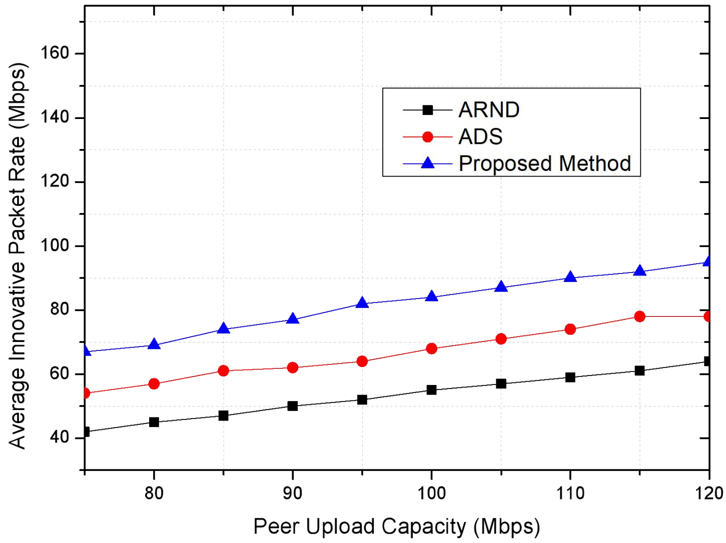

Figure 8. Average Innovative Packet Rate Vs Peer Upload Capacity

The graphical representation of Figure 9 provides a comprehensive analysis of the network parameters that are critical to a 100-node network. The network parameters considered are Stability period sequence, Lifetime round and stability period. The three different simulation parameters considered for evaluation are 500, 750 and 1000 rounds.

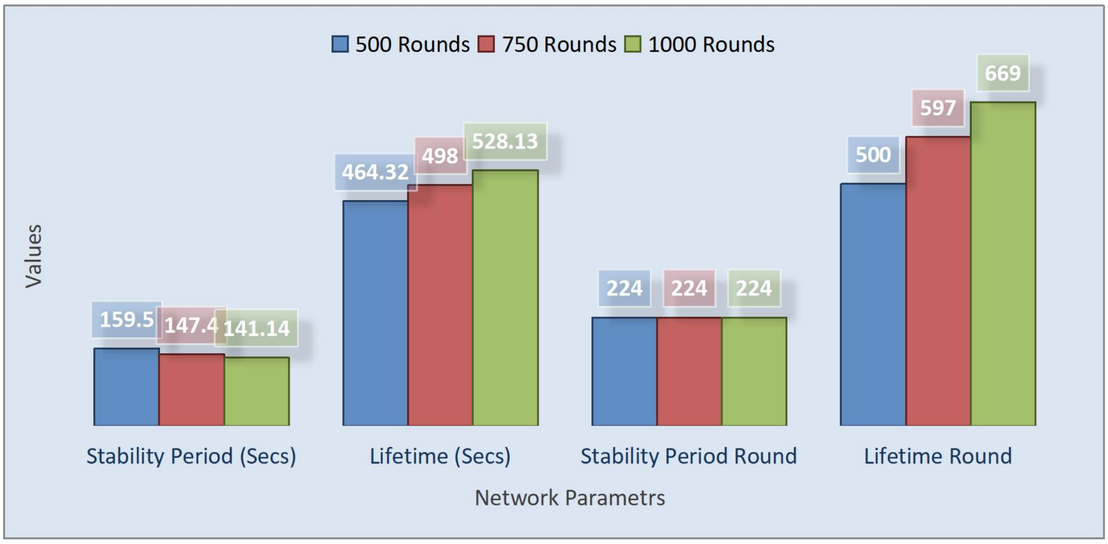

Figure 9. Evaluation of proposed with respect to network parameters (100 Nodes)

From Figure 9, It is observed that the stability period is recorded as 159.5s in 500 rounds, 147.4s in 750 rounds and 141.14s in 1000 rounds. The life time of the network is recorded as 464.32s in 500 rounds, 498s in 750 rounds and 528.13s in 1000 rounds. The life time period of the network is recorded as 224 for 500, 750 and 1000 rounds. The life time round of the network is recorded as 500 in 500 rounds, 597 in 750 rounds and 669 in 1000 rounds. It is observed that the proposed system is efficient in all the network parameters in different rounds.

The graphical representation of Figure 10 provides a comprehensive analysis of the network parameters that are critical to a 400-node network. The network parameters considered are Stability period sequence, Lifetime round and stability period. The three different simulation parameters considered for evaluation are 500, 750 and 1000 rounds.

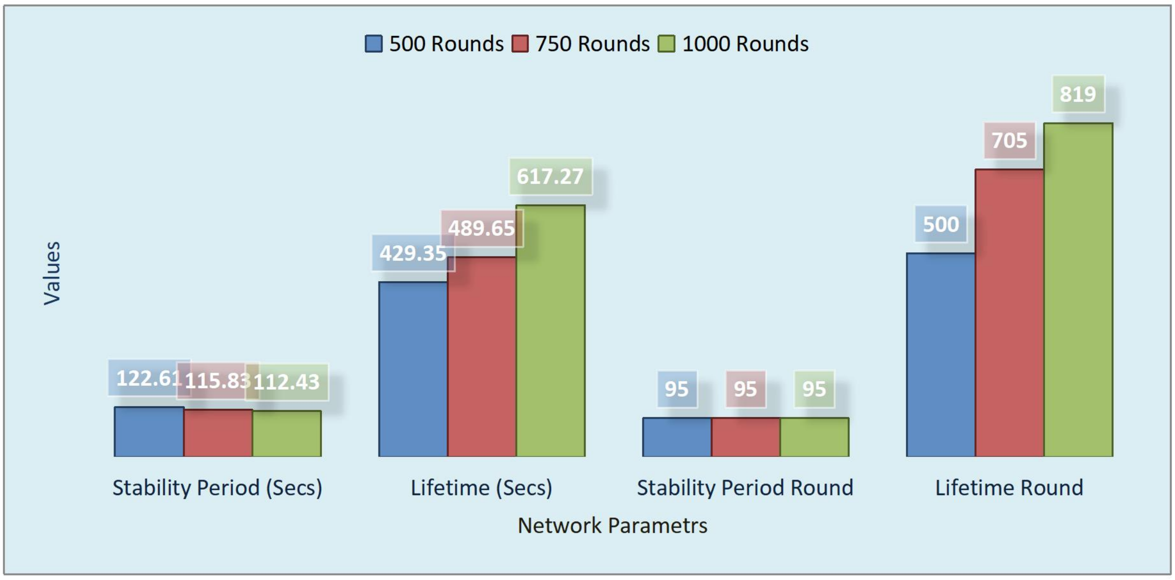

Figure 10. Evaluation of Proposed with Respect to Network Parameters (400 Nodes)

From Figure 10, It is observed that the stability period is recorded as 122.6s in 500 rounds, 115.8s in 750 rounds and 112.43s in 1000 rounds. The life time of the network is recorded as 429.35s in 500 rounds, 498.65s in 750 rounds and 617.27s in 1000 rounds. The life time period of the network is recorded as 95 for 500, 750 and 1000 rounds. The life time round of the network is recorded as 500 in 500 rounds, 705 in 750 rounds and 819 in 1000 rounds. It is observed that the proposed system is efficient in all the network parameters in different rounds.

CONCLUSION

This paper proposed the adaptive mechanism for live streaming in P2P networks. The proposed method utilizes the network coding, scheduling and selective algorithms for efficient transmission of data. It achieves the high video streaming quality and minimizes the redundancy in the network. The proposed method has a redundancy rate ranging from around 2% to 4%. The stability period is recorded as 122.6s in 500 rounds, 115.8s in 750 rounds and 112.43s in 1000 rounds for the proposed method. The proposed algorithm also achieved more number of innovative packets compared to the previous generation of P2P overlay.

ETHICAL DECLARATION

Conflict of interest: No declaration required. **Financing:** No reporting required. **Peer review:** Double anonymous peer review.

REFERENCES

[1] S. Huang, M. Sanna, E. Izquierdo, and P. Hao, "Optimized scalable video transmission over P2P network with hierarchical network coding," in *2014 IEEE International Conference on Image Processing (ICIP)*, Oct. 2014, pp. 3993-3997.

[2] A. M. Sheikh, A. Fiandrotti, and E. Magli, "Distributed scheduling for scalable P2P video streaming with network coding," in *2013 Proceedings IEEE INFOCOM*, Apr. 2013, pp. 11-12.

[3] O. E. Ojo, A. O. Oluwatope, and S. O. Ajadi, "A reliable peer-to-peer streaming protocol in low-capacity networks," *Peer-to-Peer Networking and Applications*, vol. 14, no. 2, pp. 559-584, 2021.

[4] M. K. Bideh, B. Akbari, and A. G. Sheshjavani, "Adaptive content-and-deadline aware chunk scheduling in mesh-based P2P video streaming," *Peer-to-Peer Networking and Applications*, vol. 9, pp. 436-448, 2016.

[5] R. Sharma, N. Kumar, N. B. Gowda, and T. Srinivas, "Packet scheduling scheme to guarantee QoS in Internet of Things," *Wireless Personal Communications*, vol. 100, pp. 557-569, 2018.

[6] M. Sayit, S. Demirci, Y. Kaymak, and E. T. Tunali, "Adaptive, incentive and scalable dynamic tree overlay for P2P live video streaming," *Peer-to-Peer Networking and Applications*, vol. 9, pp. 1074-1088, 2016.

[7] K. P. AjithaGladis, "Gravitational search algorithm based data scheduling for peer to peer video on demand system," *Multimedia Tools and Applications*, vol. 78, no. 19, pp. 27291-27307, 2019.

[8] K. Pal, M. C. Govil, and M. Ahmed, "Slack time–based scheduling scheme for live video streaming in P2P network," *International Journal of Communication Systems*, vol. 31, no. 2, p. e3440, 2018.

[9] S. Zhang, W. Lei, W. Zhang, Y. Guan, and H. Li, "Congestion control and packet scheduling for multipath real time video streaming," *IEEE Access*, vol. 7, pp. 59758-59770, 2019.

[10] F. Li, D. Zhang, and L. Wang, "Packet importance based scheduling strategy for H. 264 video transmission in wireless networks," *Multimedia Tools and Applications*, vol. 74, pp. 10259-10275, 2015.

[11] K. Pal, M. C. Govil, and M. Ahmed, "Priority-based scheduling scheme for live video streaming in peer-to-peer network," *Multimedia Tools and Applications*, vol. 77, pp. 24427-24457, 2018.

[12] S. Duraimurugan, R. Avudaiammal, and P. M. Vincent, "Enhanced QoS through optimized architecture for video streaming applications in heterogeneous networks," *Wireless Personal Communications*, vol. 118, no. 2, pp. 1655-1673, 2021.

[13] B. S. Monikandan, A. Sivasubramanian, S. P. K. Babu, G. K. D. Prasanna Venkatesan, and C. Arunachalaperumal, "Channel aware optimized proportional fair scheduler for LTE downlink," *Peer-to-Peer Networking and Applications*, vol. 13, pp. 2135-2144, 2020.

[14] M. Wang and B. Li, "R2: Random push with random network coding in live peer-to-peer streaming," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 9, pp. 1655-1666, 2007.

[15] A. M. Sheikh, A. Fiandrotti, and E. Magli, "Distributed scheduling for low-delay and loss-resilient media streaming with network coding," *IEEE Transactions on Multimedia*, vol. 16, no. 8, pp. 2294-2306, 2014.

[16] M. Sanna and E. Izquierdo, "Proactive prioritized mixing of scalable video packets in push-based network coding overlays," in *2013 20th International Packet Video Workshop*, Dec. 2013, pp. 1-7.

[17] G. Gautam and B. Sen, "Design and simulation of wireless sensor network in NS2," *International Journal of Computer Applications*, vol. 113, no. 16, pp. 14-16, 2015.

[18] G. Yu, X. Wang, X. Zha, J. A. Zhang, and R. P. Liu, "An optimized round — Robin scheduling of speakers for peers-to-peers-based byzantine faulty tolerance," in *2018 IEEE Globecom Workshops (GC Wkshps)*, Dec. 2018, pp. 1-6.

[19] M. Sina, M. Dehghan, A. Masoud Rahmani, and M. Reshadi, "WidePLive: A coupled low-delay overlay construction mechanism and peer-chunk priority-based chunk scheduling for P2P live video streaming," *IET Communications*, vol. 14, no. 6, pp. 937-947, 2020.

[20] M. Sasabe and M. Kiyomitsu, "Analysis of minimum distribution time of two-class tit-for-tat-based P2P file distribution," *Computer Networks*, vol. 214, p. 109142, 2022.

[21] Z. Ma, M. Yan, R. Wang, and S. Wang, "Performance analysis of P2p network content delivery based on queueing model," *Cluster Computing*, pp. 1-15, 2023.

[22] S. K. Tehariya and M. Ahmed, "New Fuzzy Logic-Based Hybrid Overlay (NFLHyO) in P2P live streaming," in *International Conference on Communication and Computational Technologies*, Jan. 2023, pp. 381-390.