



# Gravitational search algorithm based data scheduling for peer to peer video on demand system

K. P. AjithaGladis<sup>1</sup>

Received: 24 January 2018 / Revised: 8 April 2019 / Accepted: 11 April 2019 /  
Published online: 17 June 2019

© Springer Science+Business Media, LLC, part of Springer Nature 2019

## Abstract

In peer to peer (P2P) video streaming systems, peers in network assist to forward the data to other peers without the interference of central servers. Video on Demand (VoD) is widely using internet service, which offers video to users with effective control when they need it. The major significant problem in developing a P2P VoD system is data scheduling, which concentrates on dealing with transmitting and dispatching data segments within a system efficiently. So a Gravitational Search Algorithm based data Scheduling (GSAS) is presented in this paper. Initially, the network is developed in the form of hierarchical topology. Video file is cached as data segments in each peer in the top layer of the network. Using the priority function, these data segments are sorted or prioritized. Using the proposed GSA algorithm, optimal or suitable peers which cache the requested data segments in the top layer are selected. Then the prioritized data segments from the selected peers are scheduled to the peer which has requested for video sequences. The video file “Grandmother” with the size 53 MB is examined in this approach. This proposed approach is simulated in the network simulator. Simulation results show that the performance of this proposed approach is superior to that of the existing work in terms of throughput and scheduling time.

**Keywords** Peer to peer video streaming systems · Video on demand (VoD) · Gravitational search algorithm · Data scheduling · Priority function

## 1 Introduction

P2P VoD streaming is the most prominent video applications over the Internet as of late [6]. In P2P VoD system, peers require a short interval between the start time of playback and video request [7]. Nevertheless, these peers have dynamic nature i.e. they can leave and joint to the network at any time. Also, the peers can view distinctive parts of a video

---

✉ K. P. AjithaGladis  
kpajithagladis1503@gmail.com

<sup>1</sup> Department of IT, C.S.I. Institute of Technology, Thovalai, India

in the meantime. This declines their capacity to help each other. Clients ought to have the capacity to control the video stream utilizing commands like stop and rewind [1–3, 9–11, 15, 18, 24, 26]. In a P2P VoD framework, every peer can store and stream video to demanding neighbor peers. The transmission capacity cost of streaming video is spread among the peers as opposed to being centered on service providers of VoD [16, 17]. However, Expanding clients and gigantic multi video information convey awesome difficulties to vast scale VoD service framework. Because of the load set on video servers, VoD is exorbitant. This may make QoS issues, causing much more peers to quit viewing. To evade this, P2P VoD streaming frameworks require a solid error recovery convention that can reconnect left peers rapidly and proficiently to stay away from lost frames and critical deferrals [4, 5, 12, 21, 22, 24, 27, 28]. Also, this may cause effects on the utilization rate of the bandwidth of general peers for P2P VoD streaming frameworks.

So data scheduling is the solution to improve the utilization rate of the system. Many data scheduling schemes have been proposed in P2P. However, few works are available for scheduling in P2P VoD. In [25], Q. Yu et al have presented optimal data scheduling for P2P VoD. The authors have proposed a streaming data scheduling based on a max-flow scheme. Using this scheme, the authors have achieved minimum server load and maximum user playback continuity. However, scheduling time and throughput of the system are to be improved further. To overcome this problem, an efficient scheduling algorithm is presented. Contributions of this proposed approach are described as follows.

- In this approach, a P2P network is developed in the form of hierarchical topology.
- Video files are stored as data segments in each peer. These data segments are prioritized using the priority function.
- For optimal peer selection, GSA algorithm is presented. Then the prioritized data segments are scheduled from the selected peer.
- This proposed approach is implemented in the platform of Network Simulator-2 (NS2).

Rest of this paper is organized as follows. Section 2 reviews some previous work. Section 3 presents the proposed GSAS method. Results of the proposed work are discussed in Section 4. This paper is concluded with Section 5.

## 2 Related works

Only a few research works which related to data scheduling for P2P VoD system are available. So, in this section, video streaming and caching techniques in P2P VoD are discussed. Q. Yu et al [25] have proposed a max-flow-based optimal data scheduling method. The authors have aimed to improve the quality of playback continuity and minimize the server load. They have achieved their goal by optimizing the data scheduling using polynomial max-flow-based streaming data scheduling algorithms. However, scheduling time and throughput of the system are to be improved further. Eunsam Kim and Jonathan CL Liu [14] have presented an integrated prefetching and caching scheme in multi video servers that was aimed at enhancing the overall performance of the system by using cache memory space efficiently. They have considered integrating the prefetching and caching technique to improve the function of the framework. However, the interval between multiple consecutive streams has to be reduced further to improve the performance. Thibaud Rohmer et al [19] have implemented a dynamic

strategy selection based on Evidence theory to be applied on a P2P VoD system that was aimed at improving the execution of a P2P VoD framework by utilizing a less complex learning approach. They have taken into account using the strategy selection mechanism as the implementation technique. However, the efficiency of the system has to be increased when the demand increases. Kyung-Wook Hwang et al [13] have presented joint family adaptive bit rate video on demand streaming over P2P networks with realistic abandonment which was aimed at gaining the more playback rates with lesser interruption than existing systems. They have utilized the technique of converting peers to partial seeds. However, the impact of abandonment on the performance of the system has to be further reduced.

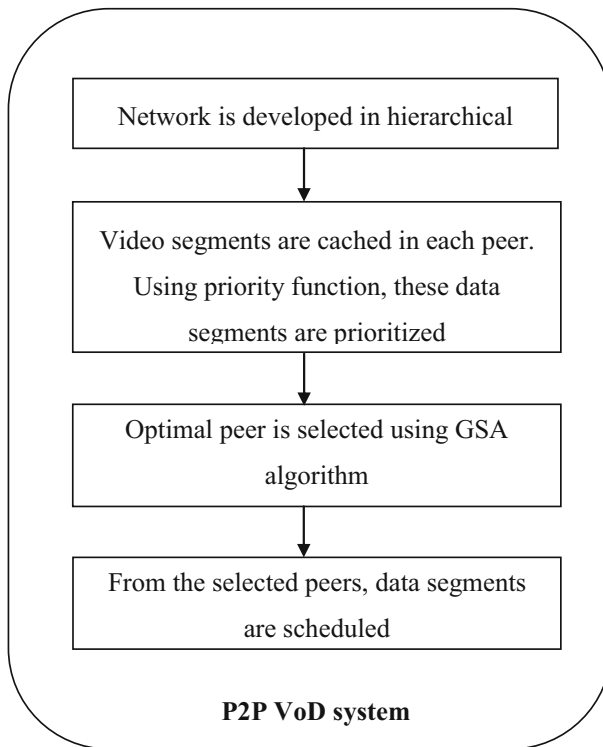
Robert Wojcik and Jerzy Domzal [23] have implemented a performance evaluation of traffic management mechanisms for P2P networks that was aimed at handling the traffic in a P2P network with efficiency. They have decided to utilize the technique of economic traffic management mechanisms to manage the traffic in the P2P network. However, the cost efficiency of the method has to be enhanced more. Sercan Demirciet al [8] have presented a hierarchical P2P clustering framework for video streaming systems that was aimed at enhancing the clustering process among peers. They have utilized the method of incorporating merge, join, split and cluster head selection mechanisms to further enhance the clustering process. However, the implementation of Vivaldi algorithm in this method has to be more productive without failing to gain full use of resources. Abdulaziz Shehabet al [20] have proposed an efficient data distribution model in P2P networks for video streaming. Using this proposed approach, authors have delivered the suitable video content to peers by prefetching requested videos based on their interests. The efficiency of video delivery systems has been measured based on the metrics server load and latency of payout.

From the above-reviewed literature, the authors attained their aim by presenting different data scheduling techniques. However, throughput and scheduling time of the network has to be further improved. So, to achieve the aim of this literature, optimal peer selection is presented. For optimal peer selection, the GSA algorithm is proposed. This algorithm selects the optimal peer efficiently based on the service capability of the peers.

### **3 Gravitational search algorithm based data scheduling P2P video on demand**

#### **3.1 Overview**

For data scheduling in P2P VoD system, GSA algorithm is presented in this paper. Initially, P2P network is developed in hierarchical form. In this approach, videos are split into data segments and these data segments are cached at each peer in the top layer. Also, these data segments are prioritized based on the priority function. If a peer from the other layers requires video to play or download, it forwards the data request to the proxy server which tracks all peers in the network. Then, using the proposed GSA algorithm, optimal or suitable peers which cache the requested data segments in the top layer are selected. Finally, the requested video segments are scheduled from the chosen optimal peer (parent peer) to the peer that requested for data. If the requested data is not available in the neighbor parent peer, the peer



**Fig. 1** Block diagram of the proposed approach

directly forwards requests to the VoD server. The block diagram of the proposed approach is shown in Fig. 1.

### 3.2 System model

The hierarchical network is shown in Fig. 2. The network is developed in a hierarchical form. Based on the available bandwidth of the peers, the top layer is formed. The first or top layer of peer peers around the VoD server is considered as the parent peers P1, P2, and P3 as shown in the figure. New peers are joined to the respective parent peers who provide the available bandwidth to the new joining peers and are formed the consecutive layer. To all the peers in the next layer, these parent peers cache the data segments. The peers in the second layer are considered as its child peers by the peer in the first layer that is denoted as C1, C2, C3, C4, C5, and C6. The peers in the same layer are considered as a sibling of one another. Proxy is situated at the prime locations of the network. Each peer in the other layers forwards video request to its parent peer. Then the requested video will be delivered to the peer with the help of a proxy server. The peer forwards the video request to the VoD server directly if the requested video is not available on the parent peer and proxy server.

Peer C4 sends a request to the proxy server through its parent peer P2 in the figure. If the requested data is available the proxy server schedules the data from the parent peers P1, P2, and P3. The videos are cached as data segments by each peer in the top layer. Using the Eq. (1) the data segments in the cache of a peer in the top layer are prioritized. The playback time of

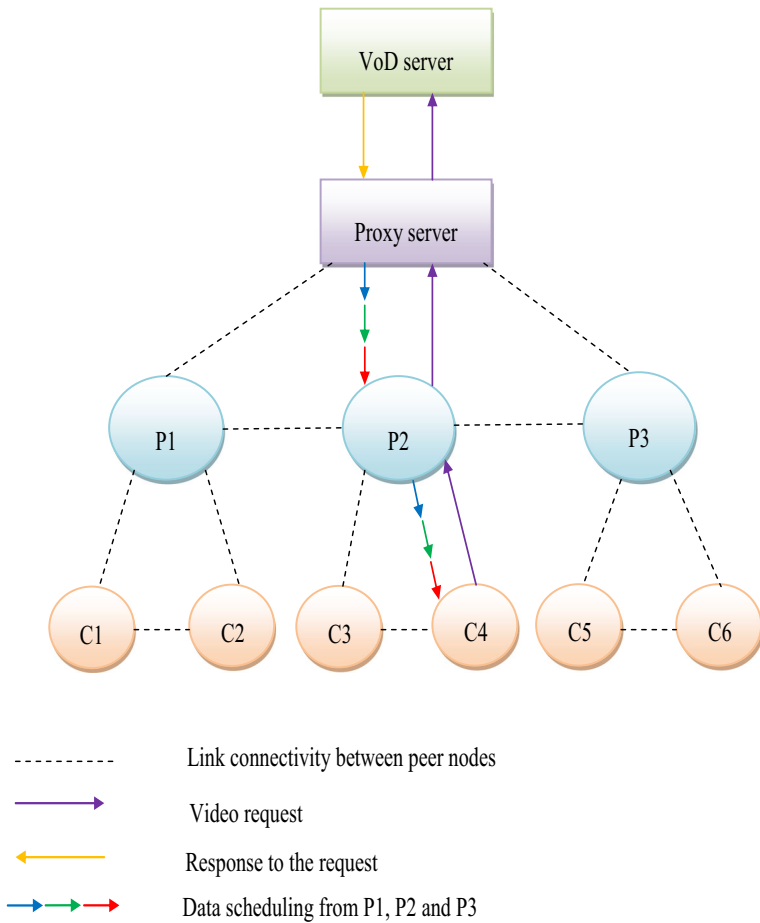


Fig. 2 System model

each data segment is considered for prioritization. The segments have higher priority to be scheduled to the peer. The priority calculation is defined as follows:

$$P_l = 1/(t_l - t_{current}) + \delta * (1/N(S_l)) + ran(0, 1) * \phi \tag{1}$$

Where,

- $P_l$  denotes the priority of segment  $l$
- $t_l$  denotes Playback time length of segment  $l$
- $t_{current}$  denotes Current playback time
- $N$  denotes Number of neighbor peers which hold the segment  $l$
- $(S_l)$
- $\delta$  denotes Coefficient of local rarest first strategy which is used to adjust the priority between the rarest priority and urgent priority of segments.
- $\phi$  denotes trembling coefficient used to adjust the trembling scope when two segments have the same rarest priority.

The proxy server schedules the requested data from the other peers if a peer in the second layer sends data request to the proxy server via the parent peer. Optimal peer is selected among the peers in the top layer using GSA for data scheduling. Service capabilities of each peer are considered as the fitness function of the GSA algorithm for optimal peer selection. The attributes taken as input to the GSA algorithm are the Uplink bandwidth, Euclidean distance and online time of the peer. Service capability of the neighbor peer  $n$  of peer  $m$  is described as follows

$$Fitness = C_m^n = \left[ t_m^n * (b_m^n)^2 \right] / d_m^n \tag{2}$$

Where,

- $t_m^n$  represents the online time of the source/parent peer  $n$  of peer  $m$
- $b_m^n$  represents the uplink bandwidth of source/parent peer  $n$  of peer  $m$
- $d_m^n$  represents the Euclidean distance between source/parent peer  $n$  and peer  $m$ .

In spite of the fact that the best distance equivalents to the minimum routing hops in the real network, Euclidean distance is calculated in this paper.

### 3.3 Optimal peer (OP) selection using GSA algorithm

For optimal peer selection from the top layer, GSA algorithm is presented in this approach. GSA performs depend on the gravity law. In GSA, objects (candidate solutions) are considered as the agents (masses). Due to the force of the gravity, the agents in the region attract each other. So the agents with heavier masses attract the agent with less mass. Thus, masses assist with the support of a straight form of communication via gravitational force. An agent with the heavy masses (that related to optimal solutions) starts to move more slowly than lighter ones. A solution of the problems describes the position of the agent or mass. Inertial and gravitational masses are determined with the support of a fitness function. The solution with heavy mass is considered as an optimal solution in the search arena.

A solution to this algorithm is the position of the OPs which are to be selected. Now, let us assume a scheme with  $i$ th agents (masses) i.e.,

$$A_i = [X_{i,1}(t), X_{i,2}(t), \dots, X_{i,D}(t)] \tag{3}$$

Where  $X_{i,d}(t)$  denotes the position of the  $i$ th agent or OPs in the  $d$ th dimension. This is also represented as,

$$X_{i,d}(t) = (x_{i,d}(t), y_{i,d}(t)), 1 \leq i \leq N_p, 1 \leq d \leq D \tag{4}$$

The force of the  $i$ th mass from the  $j$ th mass at a time  $t$  is well-defined

$$F_{ij}^d(t) = G(t) \times \frac{Mass_{PG_i}(t) \times Mass_{AG_j}(t)}{R_{ij}(t) + \epsilon} \times (x_i^d(t) - x_j^d(t)) \tag{5}$$

Where  $Mass_{AG_j}(t)$  represents the active gravitational mass associated with the  $j$ th agent at the time  $t$ .  $Mass_{PG_i}(t)$  denotes the passive gravitational mass associated with  $i$ th agent at the time

$t.\varepsilon$  and  $G(t)$  represent a small constant and gravitational constant correspondingly.  $G(t)$  is well-defined as follows

$$G(t) = G_0 \times \exp(-\gamma \times iter / maxiter) \tag{6}$$

Where,  $\gamma$  and  $G_0$  represent descending coefficient and initial value respectively. The current iteration is represented as  $iter$  and a maximum number of iteration is represented as  $maxiter$ .  $R_{ij}(t)$  denotes the Euclidian distance between the agents  $i$  and  $j$ . For each iteration, the total force on the agent  $i$  is calculated using the below equation.

$$F_i^d = \sum_{j \in K_{best}, j \neq i} ran_j F_{ij}^d(t) \tag{7}$$

Where,  $K_{best}$  represents the set of  $K$  agents which having the biggest masses with the best fitness values.  $ran_j$  represents a random number within the interval  $[0, 1]$ . By mapping the fitness, the inertial mass of each agent is designed as follows

$$mass_i(t) = \frac{Fit_i(t) - worst(t)}{best(t) - worst(t)} \tag{8}$$

$$Mass_{in_i}(t) = \frac{mass_i(t)}{\sum_{j=1}^N m_j(t)} \tag{9}$$

Where,  $Fit_i(t)$  signifies the fitness value of the  $i$ th agent at time  $t$ . In this work, the fitness value is calculated using the parameters average distance and residual energy. Using Eqs. (2), the fitness value is derived.

An agent with minimum fitness value has a heavier mass and has a better position, i.e., the better is the cluster head selection.

The values  $best(t)$  and  $worst(t)$  are well-defined as

$$best(t) = \max_{j \in \{1, \dots, N\}} Fit_j(t) \tag{10}$$

$$worst(t) = \min_{j \in \{1, \dots, N\}} Fit_j(t) \tag{11}$$

Thus, the acceleration of  $i$ th agent at time  $t$  is calculated using Eqs. (7) and (9).  $a_i^d(t)$  calculated as

$$a_i^d(t) = \frac{F_i^d(t)}{Mass_i(t)} \tag{12}$$

Velocity and position of an agent are calculated using below Eqs. (13) and (14) respectively.

$$V_i^d(t + 1) = ran_i \times V_i^d(t) + a_i^d(t) \tag{13}$$

$$X_i^d(t + 1) = X_i^d(t) + V_i^d(t + 1) \tag{14}$$

Where,  $ran_m$  denotes a uniform random variable in the interval  $[0, 1]$ . This random number offers randomized attribute to the search. The iteration counter is repeated until attaining the optimal solution. Figure 3 shows the flow diagram of this proposed approach.

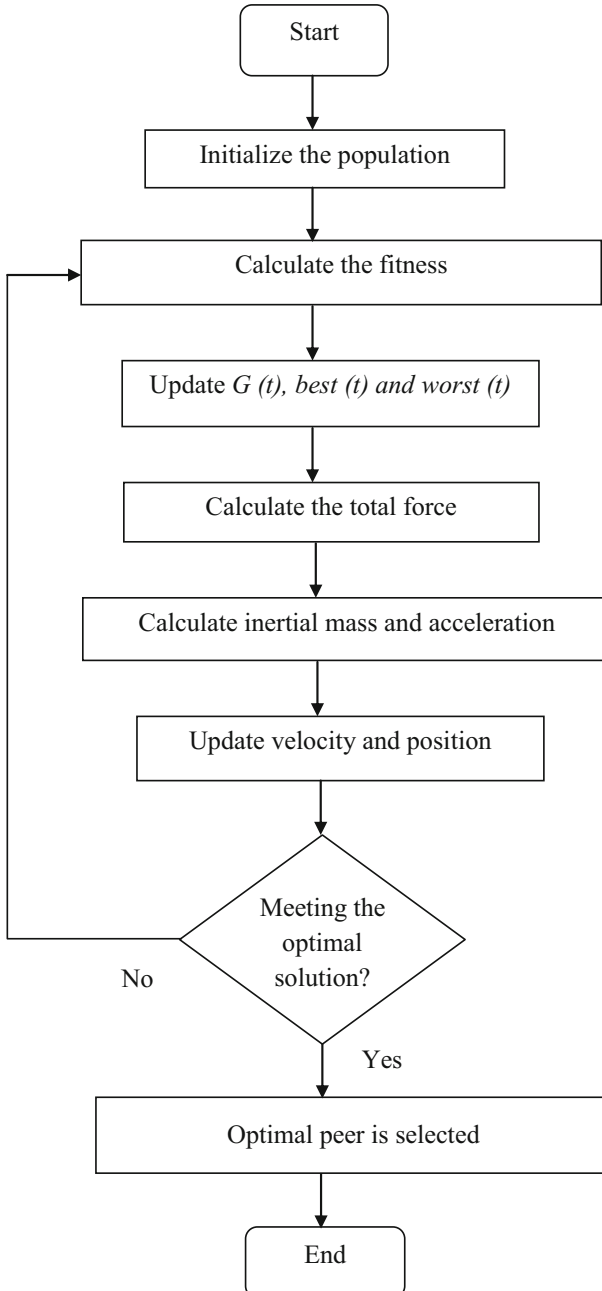


Fig. 3 Flow diagram of the algorithm



### Algorithm

1. *Initialize the Agents i.e.,  $m$  number of CHs.*
2. *Evaluate the fitness function for each agent using (10).*
3. *Update  $G(t)$ , best  $(t)$  and worst  $(t)$  using (6), (11) and (12) respectively.*
4. *Calculate the total force using (5).*
5. *Calculate inertial mass and acceleration using (9) and (13) respectively.*
6. *Update velocity and position using (14) and (15) respectively.*
7. *Process continued until get the optimal solution.*
8. *The agent with optimal solution is selected as an optimal peer.*

### 3.4 Steps of this proposed approach

1. Initially, the network is developed in the form of hierarchical topology. Based on the available bandwidth of the peers, the top layer is formed. New peers are joined to the respective parent peers who provide the available bandwidth to the new joining peers and are formed the consecutive layer. In this topology, first layer peers are known as parent peers and the second layer peers are known as child peers.
2. Each peer in the top layer caches the video file as data segments. These data segments are prioritized using the priority function.
3. Each peer in the other layers forwards video request to its parent peer. The received request will be forwarded to the proxy server. This proxy server holds information about all peers in the network, which applies the proposed GSA algorithm to choose the optimal peers in the top layer.
4. From the selected peers, the prioritized data segments are scheduled to the peer which has sent the video request to the proxy server.
5. The peer forwards the video request to the VoD server directly if the requested video is not available on the parent peer and proxy server.

## 4 Simulation results

The proposed approach is simulated using the network simulator NS2. The BitTorrent packet-level simulator is used for P2P networks. In this work, 1500 peers are performing in the region of  $1000\text{ m} \times 1000\text{ m}$ . The uplink bandwidth of the peers is 4Mbps. Transmission control protocol (TCP) is used for less data loss communication. Peers in the network are grouped in hierarchical form. In the hierarchical structure, parent peers are placed in the top layer. New peers are joined to the respective parent peers who provide the available bandwidth to the new joining peers and are formed the consecutive layer. The video sequence “Grandmother” with a frame rate of 30fps, CIF resolution, and 256 kbps bitrate is examined in the proposed

approach. Size of this video file 53 MB and it is segmented with the size of 2 MB approximately. This video file is segmented into 24 segments. Each peer in the network caches these segmented files. Duration of the simulation is 200 s.

#### 4.1 Performance metrics

Performance of the proposed approach is evaluated using the following metrics.

**Delivery ratio:** It is the ratio of the number of packets received successfully and the total amount of packets transmitted.

$$\text{Delivery ratio} = \frac{\text{Amount of packets received}}{\text{Amount of packets transmitted}} \quad (15)$$

**Packet delay:** The delay of the network describes how long the network takes to transmit a bit to the destination. Unit of this parameter is seconds (s).

**Throughput:** It is the number of data that can be sent from the sources to the destination per second. Unit of this parameter is kbps.

$$\text{Throughput} = \frac{\text{Amount of transmitted data (kb)}}{\text{Transmitted time (s)}} \quad (16)$$

**Bandwidth:** It is defined the number of bits transmitted per second. Unit of this parameter is bps.

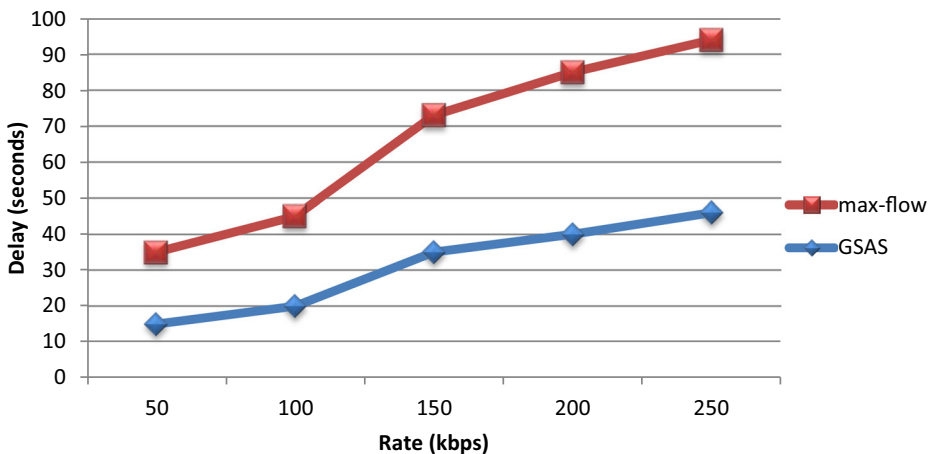


Fig. 4 Rate Vs Delay

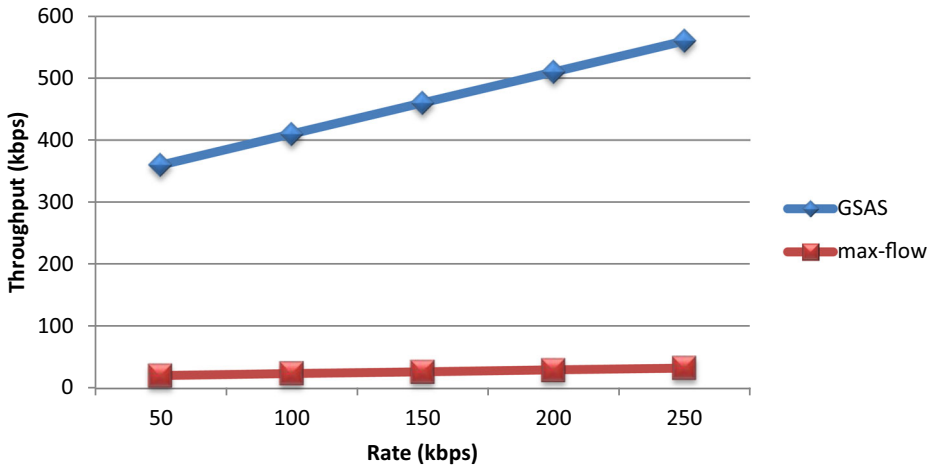


Fig. 5 Rate Vs Throughput

**Overhead:** Number of extra bytes added to the data packet to transmit the data.

**Packet drop:** It is the number of packets dropped during the data transmission.

### 4.2 Performance-based on rates

In this section, the performance metrics of the proposed approach are evaluated for varying rates 100, 200, 300, 400 and 500 kbps. Figures 4, 5, 6, 7, 8 and 9 shows that delay, throughput, delivery ratio, bandwidth, overhead and drop of the proposed approach for varying rates. Performance metrics of the proposed approach GSA algorithm based data scheduling (GSAS) are compared with that of existing work max-flow [25].

Figure 4 shows the comparison of delay of the proposed approach with that of existing work for varying rates. When the rate increases, the scheduling time of the system is also

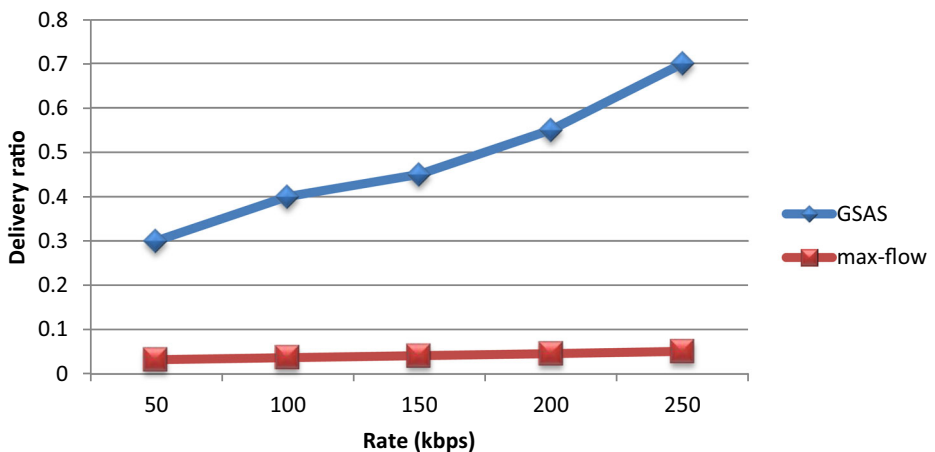


Fig. 6 Rate Vs Delivery ratio

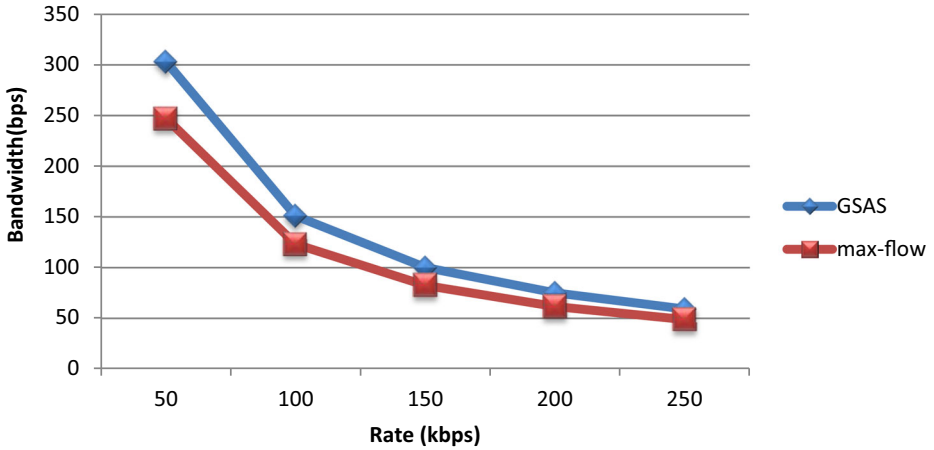


Fig. 7 Rate Vs Bandwidth

increased. But compared to the existing work, delay of the proposed approach is reduced to 68%. Because of the prioritization of the data segments using the derived priority function.

Comparison of throughput of the proposed approach with that of max-flow is shown in Fig. 5. In this work, the prioritized data segments are scheduled from the optimal peers so that the throughput of the proposed approach is increased to 94% than that of the existing work.

Figure 6 shows the comparison of the delivery ratio of the proposed approach with that of existing work. Compared to the existing work, the delivery ratio of the proposed approach is increased to 98%. Based on the selected optimal peers using GSA algorithm, the requested data is scheduled efficiently due to that delivery ratio of the proposed approach is increased.

Bandwidth utilization of the proposed approach is shown in Fig. 7. When the number of rate increases, bandwidth utilization of the system is decreased. But compared to the existing work, bandwidth utilization of the proposed approach is increased to 18%. Comparison of the overhead between GSAS and max-flow is shown in Fig. 8. Compared to the existing max-

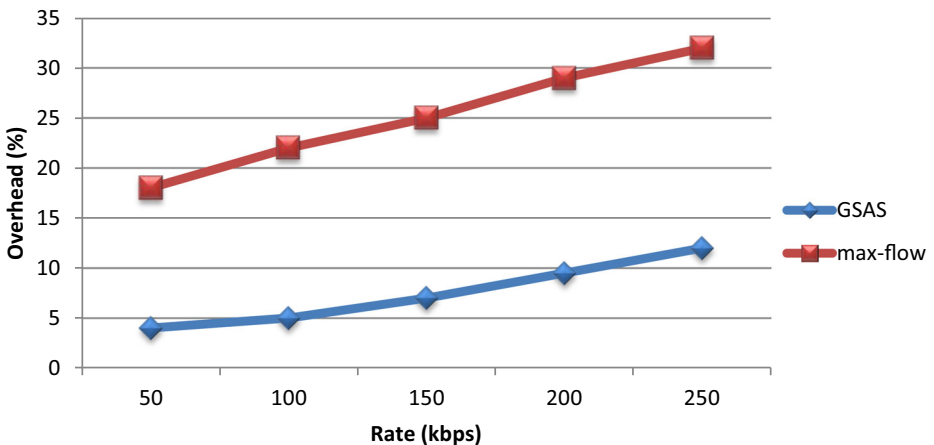


Fig. 8 Rate Vs Overhead

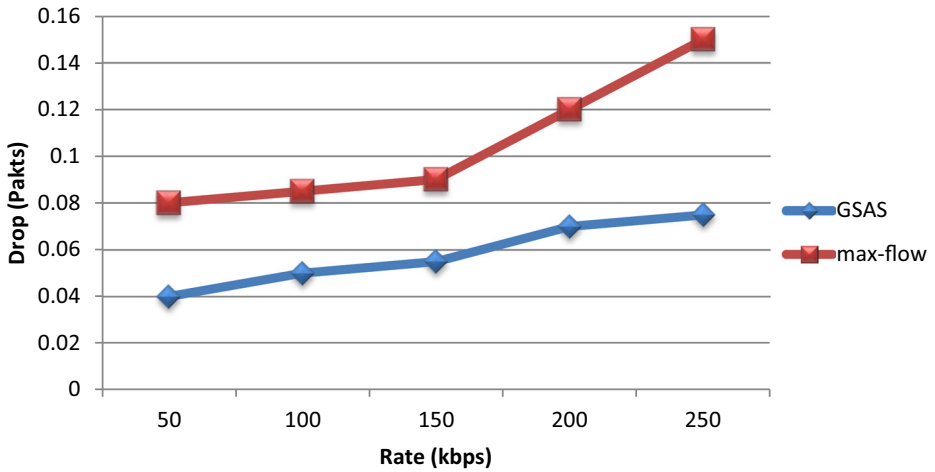


Fig. 9 Rate Vs Drop

flow, the overhead of the proposed GSAS is reduced to 35%. Figure 9 shows the comparison of the packet drop between GSAS and max-flow. Packet drop of the proposed approach is reduced to 25% than that of the existing work.

### 4.3 Performance-based on peers

In this section, performance metrics of the proposed approach are evaluated for varying number of peers 300, 600, 900, 1200 and 1500. Figures 10, 11 and 12 shows that delivery ratio, throughput, and delay of the proposed approach for varying number of peers. Performance metrics of the proposed approach GSAS are compared with that of existing work max-flow.

Figure 10 shows the comparison of the delivery ratio of the proposed GSAS with that of the existing max-flow for varying number of peers. Compared to the max-flow, the delivery ratio of the GSAS is improved to 52%. Throughput and delay of the proposed GSAS are compared

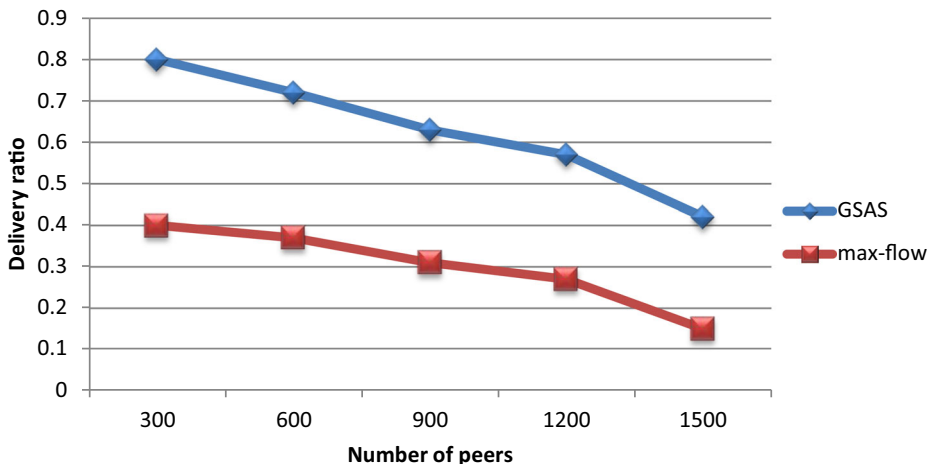


Fig. 10 Number of peers Vs Delivery ratio

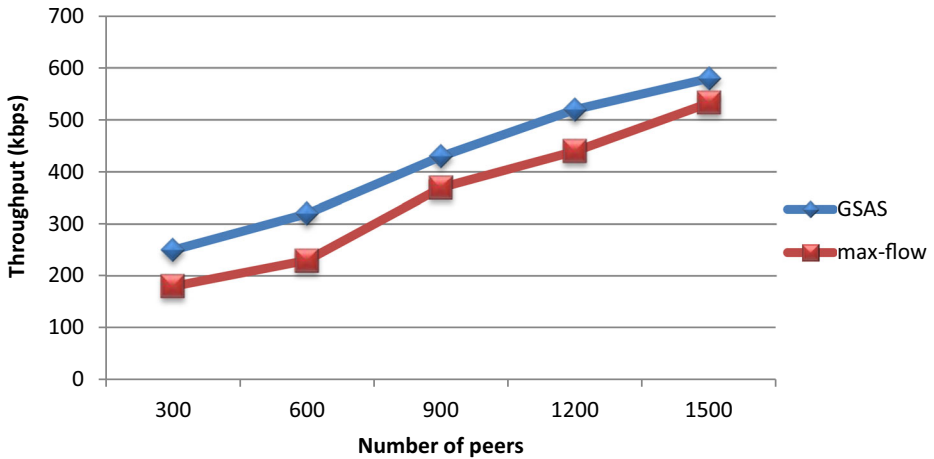


Fig. 11 Number of peers Vs Throughput

with that of max-flow for varying number of peers as shown in Figs. 11 and 12 respectively. The throughput of the proposed GSAS is increased to 15% than that of the existing max-flow. Compared to the max-flow, delay of the GSAS is reduced to 52%.

#### 4.4 Performance-based on simulation time

Figure 13 shows the comparison between a number of data scheduled peers in the proposed and existing methods for varying simulation times 40, 80, 120, 160 and 200 s. As shown in the figure, the number of data scheduled peers is increased while increasing the simulation time. Because of the proposed optimal peer selection using GSA algorithm for scheduling the requested video segments to other peers, the number of data scheduled peers or client peers of GSAS is increased to 24% than that of the existing approach.

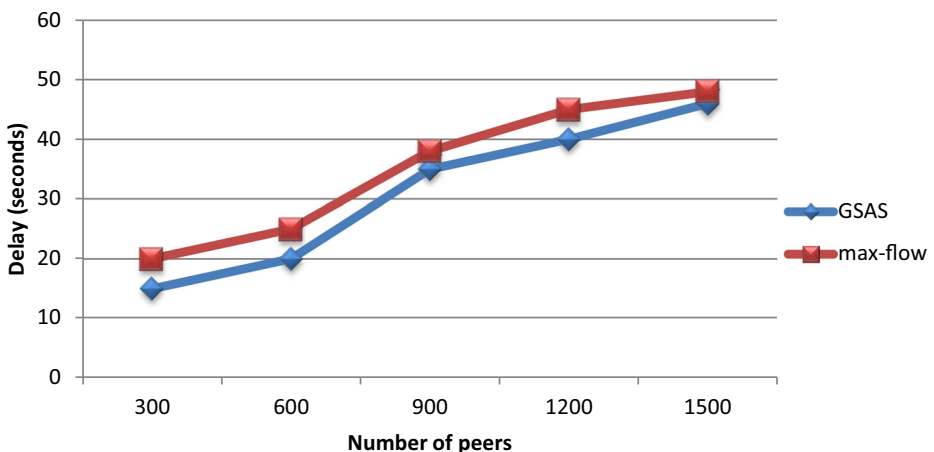


Fig. 12 Number of peers Vs Delay

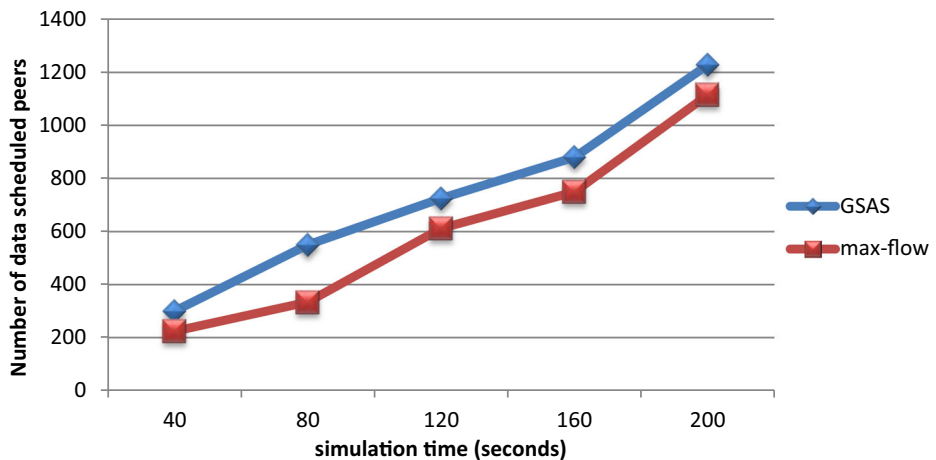


Fig. 13 Simulation time Vs number of data scheduled peers

## 5 Conclusion

In this paper, the GSAS has been presented for P2P VOD system. This proposed approach has been simulated in the platform of Network Simulator-2 (NS2). First, the network has been developed in hierarchical form. In each peer of the network, video file has been cached as data segments. These data segments were sorted or prioritized using the priority function. By caching the video as a data segments at each peer in the top layer of the hierarchical topology, server load of the network has been reduced. Based on the proposed GSA algorithm, optimal peers which cache the requested data segments in the top layer has been selected. Due to the selection of optimal peer, bandwidth utilization of the network has been improved. Simulation results showed that the performance of this proposed approach is better than that of the existing work in terms of throughput and scheduling time. However, this proposed approach compromises the secure data scheduling which is to be improved.

## References

1. Brij G, Dharma PA, and Shingo Y, eds (2016) Handbook of research on modern cryptographic solutions for computer and cyber security. IGI global.
2. Mohammad AA, Yaser J, Islam O, and Brij BG (2016) Automated wireless video surveillance: an evaluation framework. *J Real-Time Image Proc* 13(3):527–546
3. Md Zakirul Alam B, Jie W, Guojun W, Zhigang C, Jianer C, and Tian W (2017) Quality-guaranteed event-sensitive data collection and monitoring in vibration sensor networks. *IEEE Trans on Industria Informatic* 13(2):572–583
4. Ching-Lung C, Wei-Ming C, and Chi-Han H (2014) Reliable consideration of P2P-based VoD system with interleaved video frame distribution. *IEEE Syst J* 8(1):304–312
5. Bin C, Xiuzheng L, Zheng Z, Hai J, Lex S, and Xiaofei L (2008) Evaluation and optimization of a peer-to-peer video-on-demand system. *J Syst Archit* 54(7):651–663
6. Leonardo Bidese de P, and Claudio Luis de A (2006) Assessing the efficiency of stream reuse techniques in P2P video-on-demand systems. *J Netw Comput Appl* 29(1):25–45
7. Konstantinos D, and Spyros D (2017) Tackling energy and battery issues in mobile P2P VoD systems. *Comput Netw* 113:58–71
8. Sercan D, Asil Y, Muge S, Turhan Tunali E, and Hasan B (2017) A hierarchical P2P clustering framework for video streaming systems. *Comp Stand & Interface* 49:44–58

9. Mohamed E, Ahmed F, Nanrun Z, Ming-Ming W, Soliman A, and Josep B (2017) Dynamic multi-hop clustering in a wireless sensor network: performance improvement. *Wirel Pers Commun* 95(4):3733–3753
10. Mohamed Elhoseny, Alaa Tharwat, Xiaohui Yuan, and Aboul Ella Hassanien (2018) Optimizing K-coverage of mobile WSNs. *Expert Syst Appl* 92:142–153
11. Lisheng F, Xianfu L, Nan Y, Trung Duong Q, and George Karagiannidis K (2016) Secure multiple amplify-and-forward relaying with cochannel interference. *IEEE J of Sel Topic in Sig Proces* 10(8):1494–1505
12. Yun H, Zhongyang X, Yufang Z, Xiaolin T, and Zhixing L (2012) Modeling and analysis of multi-channel P2P VoD systems. *J Netw Comput Appl* 35(5):1568–1578
13. Kyung-Wook H, Vijay G, Rittwik J, Seungjoon L, Vishal M, Ramakrishnan KK, and Dan R (2016) Joint-family: adaptive bitrate video-on-demand streaming over peer-to-peer networks with realistic abandonment patterns. *Comput Netw* 106:226–244
14. Eunsam K, and Jonathan CL L (2017) An integrated prefetching/caching scheme in multivideo servers. *J Netw Comput Appl* 88:112–123
15. Ya L, Guangrun W, Lin N, Qing W, and Wenwei T (2018) Distance metric optimization driven convolutional neural network for age invariant face recognition. *Pattern Recogn* 75:51–62
16. Xiaofei L, Hai J, and Linchen Y (2012) A novel data replication mechanism in P2P VoD system. *Futur Gener Comput Syst* 28(6):930–939
17. Naeem R, Hyunggon P, and Ebroul I (2012) Video streaming over P2P networks: challenges and opportunities. *Signal Process Image Commun* 27(5):401–411
18. Esmat R, Hossein N-P, and Saied S (2009) GSA: a gravitational search algorithm. *Inf Sci* 179(13):2232–2248
19. Thibaud R, Amir N, and Abdelhamid N (2016) Dynamic strategy selection based on evidence theory. Application on a P2P-VoD system. *IFAC-PapersOnLine* 49(12):775–780
20. Abdulaziz S, Mohamed E, Mohamed Abd E-A, and Aboul Ella H (2017) Efficient schemes for playout latency reduction in P2P-VoD systems. In: *Advances in soft computing and machine learning in image processing* 477–495
21. Tauhiduzzaman M, Wang M (2015) Fighting pollution attacks in P2P streaming. *Comput Netw* 79: 39–52
22. Weijie W, Ma R, Lui J (2014) Distributed caching via rewarding: an incentive scheme design in P2P-VoD systems. *IEEE Trans on Paral and Distrib Sys* 25(3):612–621
23. Wojcik R, Domzal J (2016) Performance evaluation of traffic management mechanisms for peer-to-peer networks. *AEU Int J Electron Commun* 70(2):162–171
24. Yang X, Cores F, Hernandez P, Ripoll A, Luque E (2010) Designing an effective P2P system for a VoD system to exploit the multicast communication. *J. of Paral & Distrib Compu* 70(12):1175–1192
25. Yu Q, Ye B, Lu S, Chen D (2012) Optimal data scheduling for P2P video-on-demand streaming systems. *IET Commun* 6(12):1625
26. Yu C, Li J, Li X, Ren X, Gupta B (2017) Four-image encryption scheme based on quaternion Fresnel transform, chaos and computer generated hologram. *Mult vide Tool & Appl* 77(4):4585–4608
27. Zhou Y, Fu T, Chiu D (2013) On replication algorithm in P2P VoD. *IEEE/ACM Trans Networking* 21(1): 233–243
28. Zhou Y, Fu T, Chiu D (2015) A unifying model and analysis of P2P VoD replication and scheduling. *IEEE/ ACM Trans Networking* 23(4):1163–1175

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





**K. P. AjithaGladis** (Kochan Nadar Packianathan Ajitha Gladis) obtained her Bachelor degree in Computer Science and Engineering from Institute of Road and Transport Technology, Erode, Bharathiyar University in 1995. She obtained her Master degree in Computer Science and Engineering from Govt. College of Technology, Coimbatore, Bharathiyar University in 2002. She obtained Ph. D in Computer Science and Engineering from Vinayaka Missions University, Salem in 2012. Currently she is a Professor in IT department at C.S.I. Institute of Technology, Thovalai. Her areas of interest are image processing, and pattern recognition.