

Received April 22, 2020, accepted May 11, 2020, date of publication May 18, 2020, date of current version June 3, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2995274

A Social-Aware P2P Video Transmission Strategy for Multimedia IoT Devices

TZ-HENG HSU¹ AND YAO-MIN TUNG

Department of Computer Science and Information Engineering, Southern Taiwan University of Science and Technology, Tainan 710, Taiwan

Corresponding author: Tz-Heng Hsu (hsuth@stust.edu.tw)

This work was supported in part by the Ministry of Science and Technology of the Taiwan under Grant MOST 103-2221-E-218 -018 -.

ABSTRACT With the growing popularity of social network and video sharing services, video sharing behaviors have become increasingly socializing. People can produce and share high-quality videos at any time and place using high-speed mobile networks and multimedia IoT devices, e.g., Android Things and smart TV Boxes. The multimedia IoT devices require higher bandwidth, memory, and computing resources to process and transmit multimedia contents such as audio, video, and image. In this paper, we propose a social-aware P2P video transmission strategy for multimedia IoT devices. In the proposed architecture, multimedia IoT devices are communicated using a peer-to-peer network. The peers in our work are represented by multimedia IoT devices with user interactions. Since users have their own friend list, the social links can be classified into different priority classes in accordance with their social relationships, e. g., family, friends, and others. The proposed strategy adopts weighted fair queue (WFQ) for P2P video transmission according to different queuing priority classes. Each priority class is given a weighted factor according to social relationship, current download progress, and mutual resource sharing contributions. By leveraging the inherent trust associated with social links, the proposed strategy can reduce the impact of free riders and give users good video sharing and watching experiences with multimedia IoT devices.

INDEX TERMS Multimedia IoT, P2P network, social network, video streaming.

I. INTRODUCTION

Multimedia IoT environments consists of heterogeneous devices, e.g., smartphones, tablets, and Tv Boxes. Wireless mobile networks and multimedia IoT devices are becoming more and more popular; people can produce high-quality movies and share video at any time and any place. Users share user-generated media contents (UGCs), such as travel videos, through social network services (SNSs) with multimedia IoT devices. Video sharing has become more and more socialized due to the popularity of social networking services (SNSs) such as Facebook and Twitter. Through the relationship of friends, social media disseminates information more widely and quickly than traditional portal sites and news services.

The Internet of Things (IoT) consists of network-connected multimedia IoT devices, which enables devices to collect and exchange data with each other. Fig. 1 shows the flow chart of sharing user-generated media contents (UGCs) to social network services and video sharing websites with multimedia IoT devices. First, the source video provider, such as a

The associate editor coordinating the review of this manuscript and approving it for publication was Chun-Wei Tsai¹.

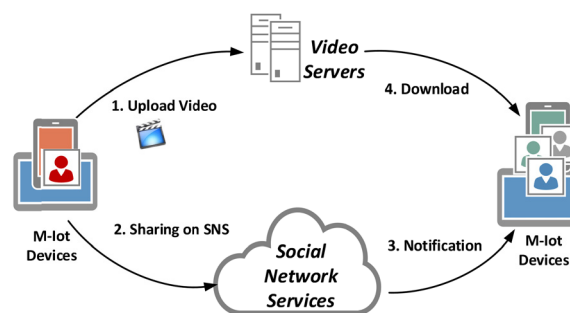


FIGURE 1. The flow chart of sharing user-generated media contents (UGCs) to social network services and video sharing websites with multimedia IoT devices.

multimedia IoT device user, uploads user-generated media content (UGC) to the video server. Then, the user utilizes social network services to share the video information. At the same time, the social network service (SNS) server actively notifies members of the user's community to share the video information. If the user's community members are interested in the video, then after clicking, the video transmission and the viewing is started on the multimedia IoT devices.

In a Peer-to-peer (P2P) network, users can share network resources such as storage space and bandwidth to other users with multimedia IoT devices. Most P2P networks have greedy peers who obtain resources as much as possible without contributing anything back, which impacts other contributors' sharing willingness and resulting in low resource sharing efficiency. Free riders often drain the network resources, i.e., only download but not upload resources to other users, which affects the sharing willingness of other contributors, resulting in the low efficiency of resource sharing in a P2P network [1].

In this paper, we propose a social aware peer-to-peer video transmission strategy for multimedia IoT devices. The peers are represented by multimedia IoT devices with user interactions. Each peer has its own community friend list. When a peer submits a video watching request, a list of community relations will be queried first in order to obtain the parameters like social weight (SW), receive contribution (RC), providing contribution (PC), and the estimated contribution value (CV). Then, the proposed video transmission strategy will calculate the social weighted contribution value (SWCV) of each participant peer. After obtaining the download progress (DP) of participant peers, the proposed algorithm then adopts the weighted fair queue (WFQ) and transfer finishing first (TFF) algorithm to perform video streaming transmission. Through the social connection and trust relationship between community users, the video streaming packets can be effectively transmitted to the community users in the heterogeneous network environment, which can reduce the impact of the free riders, lower the backbone load and network transmission cost, and provide good transmission quality to the community members with multimedia IoT devices.

II. RELATED WORKS

A. P2P AND SOCIAL NETWORKS

A peer-to-peer network refers to an Internet system without a central server and relying on peers to exchange information. Each peer is acted as a client and a server. In a P2P network, any peer can join and leave the network at any time without limitation. P2P network technologies can be used for file sharing, but P2P file sharing systems are prone to greedy behavior, i.e., free riders that take advantages instead of sharing resources. In [2], J. Altmann *et al.* conducted research on the behavior of the Gnutella network, and the results showed that 70% of peers did not share any files at all, and 85% of users were the free riders who enjoyed it. In [3], Liu *et al.* proposed a P2P network control system with the upload/download ratio of each peer, which observes the transmission variations and finally regulates a threshold to the system to adjust peers' transmission rate.

A social network is an online community established by a group of people who share the same interests and activities, which provides users with various contact, communication and interaction services. In [4], a study conducted by

Cheng *et al.* on Renren Network, the largest social networking site in China, shows that most users of social networking services (SNSs) are willing to share network resources and bandwidth to close friends to help them download videos. In [5], Pouwelse *et al.* proposed the Tribler file sharing mechanism to improve the file search, recommendation and download of multimedia contents by considering the social relationships and trust among users. In [6], Cheng *et al.* proposed a peer-assisted system for users to easily share personal recorded videos and to establish a video sharing website similar to YouTube by making use of the characteristics of the social community, such as the characteristics of the small-world network model. In [7], Liu *et al.* introduced a tit-for-tat mechanism for peer-to-peer content sharing. Peers exchanged resources through social relationship links. Each peer can limit the size of resources provided to other peers. To exchange of resources, the proposed mechanism establishes a connection according to the community relationship between the peers, and the content and resources will be transmitted through the path of the community link. In [8], Wang *et al.* proposed a social media sharing system, which allocates video streaming resources and adjusts the transmission bandwidth to reduce the server loads in accordance with the user's mutual contribution in the social community.

In [9], Li *et al.* examined the behavior of users on social networks by monitoring the behavior of more than 1 million users and crawled the data of 2,500 videos on Facebook. The results are as follows: most of the videos watched by users are from close friends of the users themselves, and most of the video watching behaviors are promoted through social interaction, and the rest are driven by interests. Based on the observation results, the author introduced the SocialTube system that systematically explores the social relationships and common interests of Online Social Networks (OSNs) in order to enhance the transmission efficiency of video sharing. SocialTube is a social network-based P2P architecture based on a social video acquisition algorithm to increase the accuracy of video prefetching and reduce the initial playback delay of the video. In [10], Kang Chen *et al.* proposed a reputation system named SocialTrust to incentivize peer collaboration in P2P networks. The reputation of a peer is based on feedback from other peers, and the reputation threshold is used to distinguish selfless or selfish peers. Due to the high service cost of reputation inquiry, frequent reputation inquiry can cause the system to overload and reduce service quality. SocialTrust combines traditional reputation systems and social networks to overcome shortcomings. SocialTrust has the following advantages over previous reputation systems: (1) Integrate social networks and reputation systems to save reputation query costs. (2) SocialTrust considers the social grade and reputation of a peer to measure its trust level, making the reputation evaluation more accurate. (3) SocialTrust encourages node cooperation, and the system uses service provision values and rating evaluation values to calculate trust reputation.

In [11], Rajapaksha *et al.* presented a SocialiVideo approach that allows users to share their produced video content among existing social connections. SocialiVideo stores video content in users' networking devices and serves others using a P2P connection. The authors implement a prototype based on Facebook/Akamai content delivery approach and evaluates the performance. The results show that SocialiVideo provides benefits for multiple participants, including CDNs and ISPs, as well as better QoE for end users. In [12], Shahriar *et al.* proposed a time-based user grouping and replication protocol that guarantees content availability for decentralized sharing of online social media. The protocol discovers cyclic diurnal patterns in user uptime behaviors to make sure the content persistence with minimal replication cost. The authors present a mathematical model for peer-uptime duration and replication group size. Simulation results show that the proposed protocol reaches high content persistence without suffering substantial network and storage overheads. In [13], Cui *et al.* proposed an evolutionary game theory based framework to analyze the effect of resource allocation mechanisms on peers' contribution behaviors such as donors. The authors have analyzed and found that allocation mechanisms can encourage contribution behaviors of peers when the benefit of resources is larger than specific thresholds. The results show that the impact of user behaviors is limited when the benefit of resources is large enough.

B. PACKET SCHEDULING MECHANISMS

In order to provide more efficient and real-time streaming services, many scholars have developed various packet scheduling mechanisms based on the different service requirements of each data stream. From the earliest first-in-first-out (FIFO) policy, followed by the concepts of priority and queuing class, the priority queue (PQ) is developed. However, priority queues may cause starvation of low-priority data streams. In order to improve this unfair mechanism, Bennett and Zhang [14] proposed the concept of fair queue (FQ). Weighted round robin (WRR) is evolved from fair queue. It is mainly used for a fixed size packet system. Each cycle WRR sorts the packet transmission sequences proportional to the queue weight [15], [16]. Weighted fair queue (WFQ) is a scheduling mechanism proposed by Demers *et al.* [17]. Compared to weighted round robin (WRR) that uses packet as the unit, WFQ uses byte as the adjustment unit. First, packets are classified according to different bandwidth requirements. After that, the estimated transmission completion time of each packet is calculated according to the bandwidth usage, packet length, and arrival time of the packet. Then, the packet transmission is determined according to the estimated delivery completion time.

Although WFQ improves the unfairness of WRR priorities, it also increases the computational complexity. Therefore, Shreedhar and Varghese [18] proposed a deficit round robin (DRR) scheduling mechanism that mainly adopts the method of round robin (RR). DRR mechanism adds Deficit Counter to each queue to represent the increment

of weight and counter each time. The scheduler will first give each queue the amount of bandwidth that can be used (Quantum size). At the end of each cycle, each queue will use the remaining Deficit Counter plus the Quantum size as the next cycle Deficit Counter. Each time DRR sends a packet, it will first check the Deficit Counter. If there is a remaining Deficit Counter available and the Deficit Counter value is greater than the packet length, the packet can be sent out. The remaining Deficit Counter is subtracted with the same packet length and then continue to compare each queue in sequence [15], [16]. In [19], Proskochylo *et al.* studies queue management mechanisms on the routers. The effect of queue management mechanisms, e.g., FIFO, PQ, CQ, WFQ, LLQ, to QoS for real-time traffic on the router in an IP network was examined. Except FIFO and CQ, the simulation results revealed that most of the queuing mechanisms can provide high QoS for real-time network traffic.

In this paper, through the trust relationship between people, we developed a social-aware P2P network video transmission strategy to stimulate the sharing willingness of users using multimedia IoT devices. Finally, the download progress (DP) value combined with weighted fair queuing (WFQ) and transfer finishing first (TFF) algorithm is used.

III. ALGORITHMS

This section describes the proposed social-Aware P2P video transmission strategy in detail.

A. SOCIAL RELATIONSHIP MANAGEMENT

In the proposed mechanism, the peers are represented by multimedia IoT devices with user interactions. Each peer has its own friend list and the corresponding social relationships. A Peer can set designated types of the relationship between friends. In the proposed mechanism, the friend relationships are divided into five different levels in descending order: family (5), close friends (4), acquaintances (3), friends with similar interests (2), and online friends (1), with the higher number (weight) the higher priority. The five levels of friend relationships are defined and influenced by [20], [21]. An example of the social relationship and resource contribution table is shown in the table I, containing five fields: friend name, relationship, social weight (SW), receiving contribution (RC) in terms of uploaded volume, and providing contribution (PC) in terms of downloaded volume.

TABLE 1. An example of social relationship and resource contribution table.

Name	Relationship	Social Weight	RC	PC
John	family	5	200MB	100MB
Bob	close friends	4	0MB	90MB
Peter	acquaintances	3	90MB	150MB
Jerry	friends with similar interests	2	100MB	900MB
...

Each peer can upload and download data from other peers at the same time. When a peer requests for videos with other peers, after the transmission is successful, the requesting peer records the amount of data shared and transmitted by the sharing peer as the receiving contribution (RC). Meanwhile, the sharing peer records the amount of data provided and transmitted by the requesting peer is regarded as the providing contribution (PC).

B. TRANSMISSION QUEUE MANAGEMENT

In the proposed mechanism, the weights in WFQ are defined as bandwidth reservation values proportional to different friend relationships, current transmission status, and resource sharing contributions.

Fig. 2 illustrates an example of peer requests in the proposed social-aware P2P video transmission strategy. In Fig. 2, peer A, B, C, D sends requests to sharing peer E for downloading the desired video contents. Peer A has two data flows A1 and A2; the corresponding data flow of Peer B, C, D is denoted as B1, C1, and D1. The social weight (SW) value of peer A is 5; the SW value of peer B is 2; the SW value of peer C is 1, and the SW value of peer D is 1.

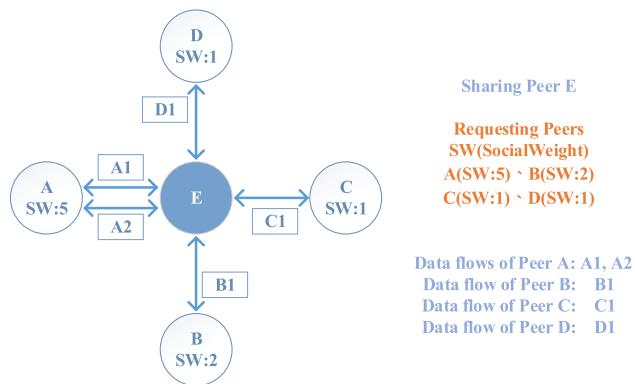


FIGURE 2. An example of peers that send requests for retrieving videos.

Fig. 3 depicts the proposed social-aware P2P video transmission architecture with the WFQ queuing policy. The social-aware classifier will calculate the contribution value (CV); each data flow will be allocated to a different weighted

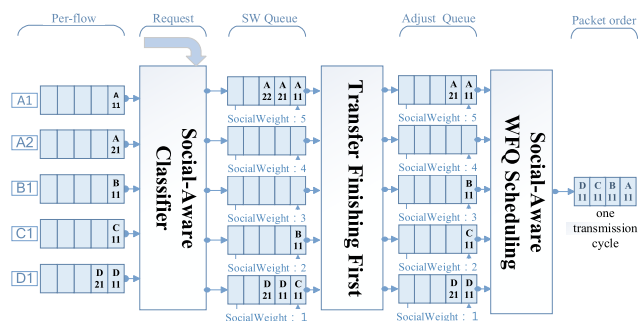


FIGURE 3. The proposed social-aware P2P video transmission architecture with WFQ queuing policy.

queue. The queue adjustment module adapts each data flow’s position according to the current transmission status and resource sharing contributions, and finally the data is sent to the social-aware WFQ for waiting packet scheduling.

When the sharing peer E receives peer A, B, C, and D’s requests, peer E queries the social relationship and resource contribution table for getting the corresponding value of social weight (SW), friend data contribution in terms of receiving contribution (RC), and friend data consumption in terms of providing contribution (PC). After calculating the contribution value (CV), then the social-aware classifier (SAC) will calculate the social weighted contribution value (SWCV) of each peer. After calculating the SWCV value, the transfer finishing priority value (TFFV) is estimated. Then, the algorithm determines the TFFV value falls in which classification range $Classifier_i$ and assigns the data flow to the queues with different weight, and finally sends the data to social-aware WFQ for packet scheduling.

Fig. 4 illustrates the flowchart of the proposed social-aware P2P video transmission architecture. The social-aware classifier determines and arranges the packet to corresponding priority queue. Then, the queue adjustment module assigns the packet to corresponding transmission queue according to the current transmission status and resource sharing contributions. Finally, the social-aware WFQ scheduler services the packet according to the scheduled starting and finishing time.

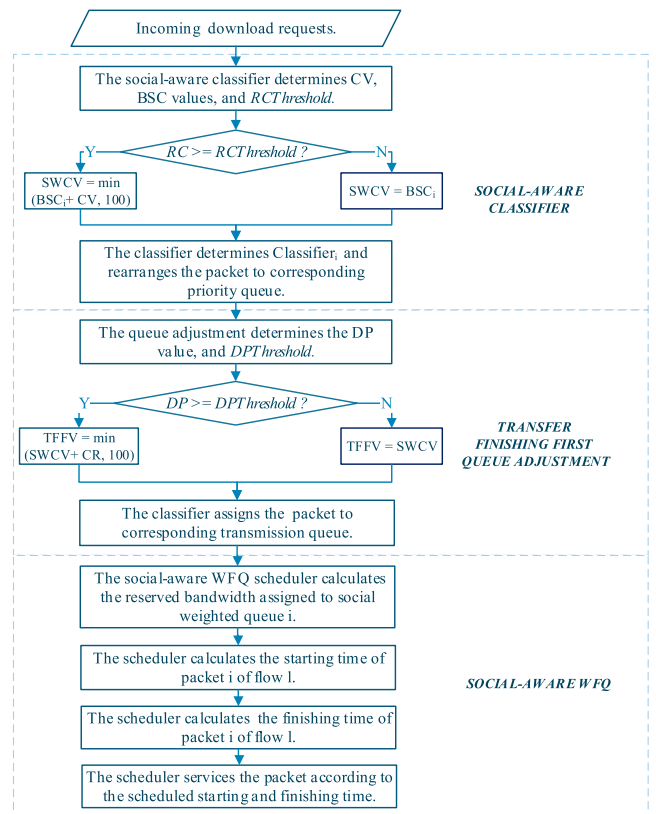


FIGURE 4. The flowchart of the proposed social-aware P2P video transmission architecture.

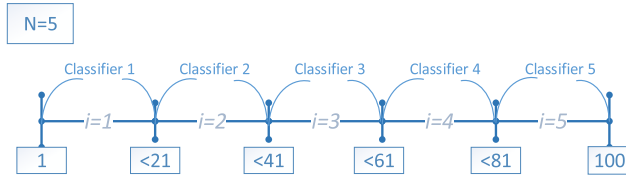


FIGURE 5. The $N = 5$ classification levels.

C. SOCIAL-AWARE CLASSIFIER

Assuming that there are N categories of distance ranges, where the classifier range (CR) is defened as the range value of each distance, which is calculated as shown in equation (3.1).

$$CR = \frac{100}{N} \tag{3.1}$$

Fig. 5 shows the $N = 5$ classification levels. For each class i , $ClassifierMax_i$ denotes the upper bound value of the i^{th} class, and $ClassifierMin_i$ denotes the lower bound value of the i^{th} class, as shown in equations (3.2) and (3.3).

$$ClassifierMax_i = \min(i \times \frac{100}{N}, 100) \tag{3.2}$$

$$ClassifierMin_i = (i - 1) \times \frac{100}{N} + 1 \tag{3.3}$$

When the social-aware classifier of the providing peer receives requests sent from the requesting peers, the classifier will query and obtain the social weights from the social relationship and resource contribution table. Then, the classifier calculates the contribution value (CV) of each request data flow based on equation (3.4),

$$CV = CR \times \frac{RC}{RC + PC} \tag{3.4}$$

where CR denotes the value of classifier range. Considering the social weight of friendships and the data contribution, a social weighted contribution value (SWCV) is defined. The social weighted contribution value (SWCV) is calculated from the basic social score (BSC) and data contribution value (CV). The basic social score (BSC) is defined as the intermediate value of i^{th} range classification level, as shown in equation (3.5)

$$BSC_i = CR \times (i - 1) + \frac{CR}{2} \tag{3.5}$$

The social weighted contribution value (SWCV) is defined in equation (3.6), where $RCThreshold$ denotes the threshold of a peer’s data provision value for leveling up the classified priority queue to $i+1^{th}$ level. For example, as shown in table I, only Jerry can have the opportunity level up to $i+1^{th}$ level if the $RCThreshold$ is set to 100MB. If the receiving contribution (RC) value of a peer is higher enough, then the peer will be leveled up. The best value of $RCThreshold$ can be set up based on empirical analysis, which may be vary in different networking environments. The social weighted contribution value (SWCV) is designed to encourage the

friends to share resources and contribute more data for getting higher transmission priority, hence eliminating the free-rider problem.

$$SWCV = \begin{cases} \min(BCS_i + CV, 100), & RC \geq RCThreshold \\ BCS_i, & RC < RCThreshold \end{cases} \tag{3.6}$$

After obtaining the social weighted contribution value (SWCV), the classifier next determines which $Classifier_i$ that the SWCV value falls into. Then, the requesting peer’s data flows are placed into the corresponding priority queues. When the source peer starts transmitting data streams to the requesting peers, the social-aware classifier sends packets in accordance with the weights assigned to the sending queues, and then performs transfer finishing first (TFF) queue adjustment algorithm.

D. TRANSFER FINISHING FIRST QUEUE ADJUSTMENT

While transmitting videos, the source providing peer obtains the information of downloading video file size and the downloading progress of the requesting peer. The queue adjustment module considers the downloading progress archived and then levels up the data stream in which the downloading progress is greater than the predefined threshold $DPTreshold$. The transfer finishing first policy can achieve early completion of data transmission and speed up the release of the connection resources.

After the social weighted contribution value (SWCV) is obtained, the file size and the data transmission progress of the current requesting peer can be known, and then the transfer finishing first value (TFFV) is calculated as defined in formula (3.7). We use the download progress (DP) of the data to adjust the transmission queue and level up the data stream whose download progress (DP) is greater than $DPTreshold$, and increase the queue weight by one step, up to queue level 5. The unit of Download Progress (DP) is expressed as percentage completed of the download. This kind of adjustment is beneficial to those peers who can complete the transmission early and release system resources. The best value of $DPTreshold$ can be set up based on empirical analysis, which may be vary in different networking environments. After obtaining the TFFV value, the algorithm determines which classification range $Classifier_i$ the TFFV value will fall in, and then assigns the requesting peer to the corresponding transmission queue, and wait for social-aware WFQ to be scheduled.

$$TFFV = \begin{cases} \min(SWCV + CR, 100), & DP \geq DPTreshold \\ SWCV, & DP < DPTreshold \end{cases} \tag{3.7}$$

Fig. 3 shows an example of queuing adjustment. In Fig. 3, after calculating each data flow’s SWCV value, the data flow B1, C1, and D1 are placed into corresponding social-weighted queue (SWQ) SWQ2, SWQ1, and SWQ1. Followed by adjusting the social-weighted queue with the transfer

TABLE 2. The symbols used in WFQ packet scheduling.

Symbol	Definition
N	Classification Levels
SWB_i	The reserved bandwidth assigned to social weighted queue i
$BW_{Provided}$	The total bandwidth providing by the source peer.
SW_i	Social weighted value of queue i

finishing first policy, where the downloading progresses of B1 and C1 are greater than $DThreshold$, data flows B1 and C1 will be leveled up, i.e., data flow B2 is upgraded to SWQ3 queue and data flow C1 is upgraded to SWQ2 queue, waiting for next packet scheduling.

E. SOCIAL-AWARE WFQ

We use the WFQ scheduler to dispatch data packets according to the reserved bandwidth assigned to each social weighted queue. The reserved bandwidth assigned to social weighted queue i is defined in equation (3.8). Table 2 defines the symbols used in WFQ packet scheduling.

$$SWB_i = \left(\frac{SW_i}{\sum_{j=1}^N SW_j} \right) \times BW_{Provided} \quad (3.8)$$

Let $S(l, i)$ denote the starting time of packet i of flow l , $F(l, i)$ denote the finishing time of packet i of flow l , $A(l, i)$ denote the arrival time of packet i of flow l . Considering the finishing time of packet $i-1$ and the arrival time of packet i of flow l in queue, the starting time of packet i of flow l is calculated as equation (3.9).

$$S(l, i) = \max\{A(l, i), F(l, i-1)\} \quad (3.9)$$

Let $L(l, i)$ denote the length of packet i of flow l . The finishing time of packet i of flow l is calculated as Equation (3.10).

$$F(l, i) = S(l, i) + L(l, i) / SWB_i \quad (3.10)$$

where SWB_i denotes the reserved bandwidth assigned to social weighted queue i .

Fig. 6 shows an example of social-aware WFQ packet scheduling. Assume there are five priority queues, the total bandwidth $BW_{provided}$ is 180 units, the allocated bandwidth for $SWB_5, SWB_3, SWB_2,$ and SWB_1 are denoted as 100, 40, 20, and 20. The estimated finishing time of each packet according to Equation (3.9) and (3.10) is denoted as $A11=0.6, B11=1.0, C11=2.0,$ and $D11=2.5$. As a result, the sending order in sequence is A11, B11, C11, and D11.

IV. PERFORMANCE EVALUATION

This session simulates and evaluates the performance of the proposed packet delivery strategy. The experiments are divided into five simulation environments (a), (b), (c), and (d) to explore and analyze the results.

- The main purpose of the simulation environment (a) is to compare the number of packets received by each peer when the social-aware WFQ weight is set by each peer under the same transmission time.

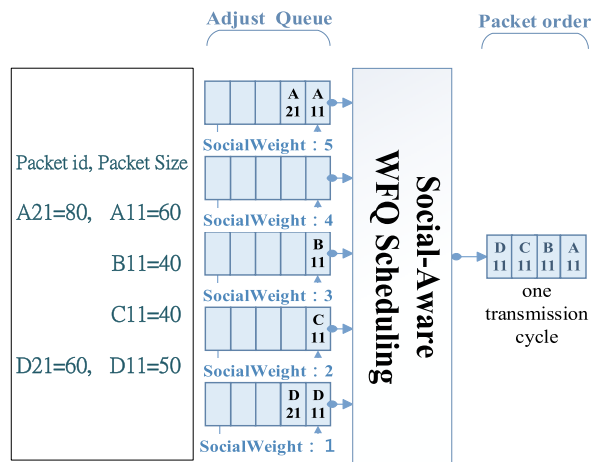


FIGURE 6. An example of Social-Aware WFQ packet scheduling.

- In the simulation environment (b), the request time of each requesting peer is separated to observe the change in data throughput.
- The main purpose of the simulation environment (c) is to observe the change in data throughput while the requesting peer’s receiving contribution RC) is greater than $RCThreshold$ and the social weighted contribution value (SWCV) has reached the upgrade level; the SW value will increase by one.
- The main purpose of the simulation environment (d) is to observe the change in data throughput while the requesting peer’s download progress (DP) is greater than 90% of the preconfigured $DPTthreshold$ and the transfer finishing first value (TFFV) has reached the upgrade level; the SW value will increase by one.

In the simulations, we use the network simulator NS-2 with the version 2.35 to simulate the network environments [22]. The proposed system refers to the methods introduced by Stadtfeld [23] to establish the peers in the simulation environment. In the simulated environment, when a sharing peer that owns the video receives a video request, the sharing peer then establishes a CBR connection with the requesting peer for data transmission. The resource sharing peer will use weighted fair queuing (WFQ) to schedule the data packet transmission.

The NS2 simulation environment is 1400 * 1400 meters; the MAC type is 802.11g, the routing protocol is AODV; the broadcast radius is 200 meters, and the bandwidth is 54Mbps. There are 10 peers in total, 5 are request peers, 3 are peers responsible for forwarding packets, and 2 are source peers providing the movies. The source peers use weighted fair queuing (WFQ) to schedule the data packets. Simulation environment settings are shown in Table 3.

A. SIMULATION ENVIRONMENT (A)

The simulated environment compares the CBR packet size to 1,000 bytes with the transmission time of 10, 20, 30, and 50 seconds. The social-Aware WFQ weight is set by each

TABLE 3. Simulation environment settings.

Network Environment Settings	
Environment Area	1,400*1,400m
Simulation Peers	10
Broadcast Radius	250m
Wireless Bandwidth	54Mbps
Transport protocol	802.11g
Routing protocol	AODV
Video Transmission Mode	CBR
CBR Transmission Rate	32Mbps
CBR Packet Size	1,000 bytes
Video Size	1GB
Wireless 802.11g Parameter Settings	
freq / Pt	2.4e+9 / 0.28183815
RXThresh / CStresh	3.652e-10 / 1.559e-11
SlotTime / SIFS	0.000009 / 0.000010
PreambleLength	96
PLCPHeaderLength	40
PLCPDataRate	6.0e6
dataRate	54.0e6
basicRate	1.0e6

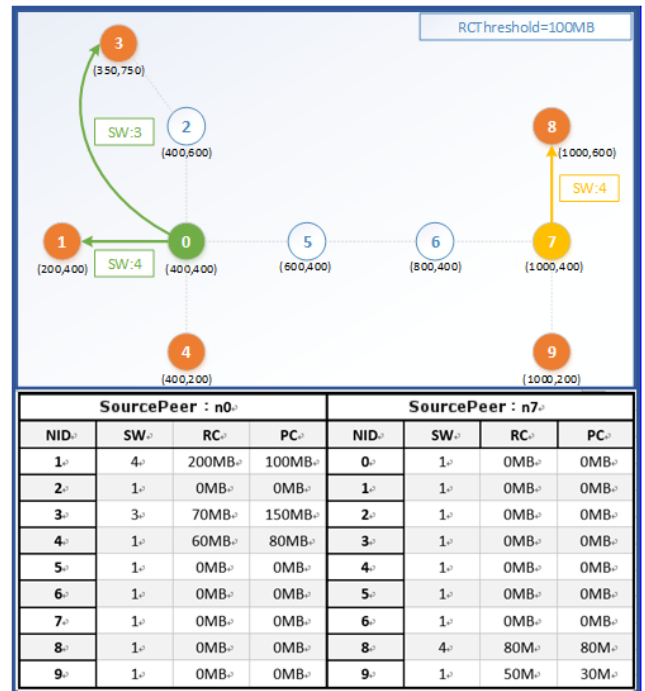


FIGURE 8. The simulation environment (a) and the social relationship list.

peer under the same transmission time, using and not using the social weight (SW). The simulation compares the packet reception and data throughput of the requesting peer. This simulation does not consider download progress (DP).

In Fig. 7, the video source peers are n0 and n7; the source peer n0 has Video_a, Video_c, Video_d, Video_e files; peer n7 has Video_b, Video_c, Video_d, Video_e files; the requesting peer is n1, n3, n4, n8, and n9. Peer n1, n3, and n4 send video requests to peer n0 for watching Video_a; Peer n8 and n9 send video requests to peer n7 for watching Video_b. The interval between transmission sequences is set to 0.2 second.

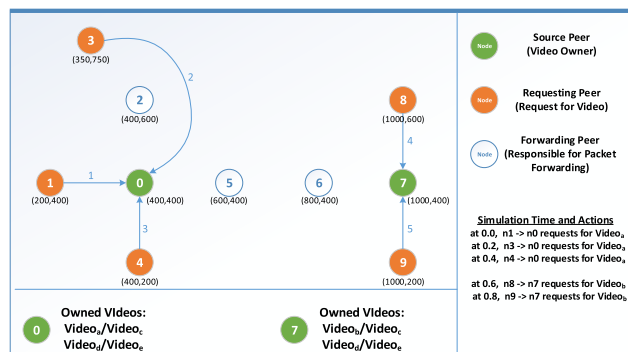


FIGURE 7. The network configurations of simulation environment (a).

After peer n0 and n7 receiving the video requests from other peers, the source peers examine the social relationship list among the request peers and categorize with the social-aware classifier (SAC). The *RCThreshold* is set to 100MB, with 5 weight levels. Fig. 8 shows the simulation environment and the corresponding social relationship list.

Fig. 9 shows the receiving data throughput of peers with source peer n0, where SAWFQ denotes the simulation results of social-aware WFQ algorithm and Original denotes the simulation results without the queue adjustment algorithm. The requesting peers are n1, n3, n4, and SW are 5: 3: 1, respectively. The social aware WFQ (SAWFQ) receives 44,345K bytes in peer n1, 26,174K bytes in n3, and 8,759Kbytes in

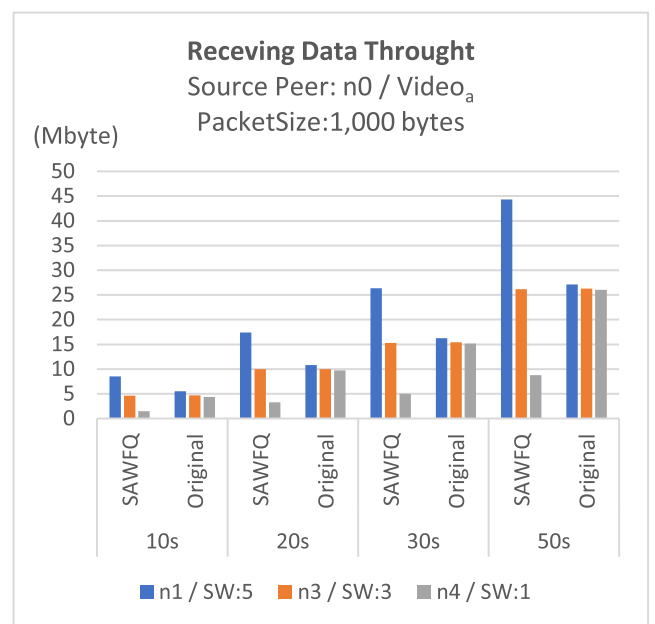


FIGURE 9. The receiving data throughput of peers with source peer n0.

n4 in 50 seconds. Comparing the original SW with 1, the peer n1 using SAWFQ has a larger bandwidth usage because SW is 5, and the amount of received packets is 17,218K bytes more than the original (SW=1), which increases 63.5% transmission data volume. The peer n3 using SAWFQ is decreased by 17,257K bytes and 66% because the SW is 1 and the allocated bandwidth is smaller than peer n1.

Fig. 10 shows the receiving data throughput of peers with source peer n7. The requesting peers are n8 and n9, and the SW are 4: 1, respectively. The SAWFQ receives 82,040K bytes in peer n8 and 20,488K bytes in n9 in 50 seconds. Comparing the original SW with 1, the peer n8 using SAWFQ has a larger bandwidth because SW is 4, and the amount of received packets is 30,364K bytes more than the original (SW=1), which increases 58.7% transmission data volume. The peer n9 using SAWFQ is reduced by 30,360K bytes and 59.7% because the SW is 1 and the allocated bandwidth is smaller than peer n8.

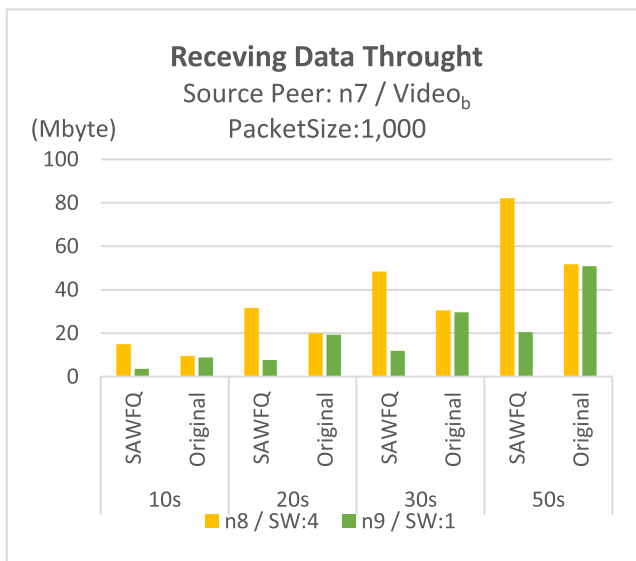


FIGURE 10. The receiving data throughput of peers with source peer n7.

From the above results, it shows that peers with higher SW have better data reception than other peers. It means that in social relationships, people who are relatively close, such as close friends, will be given higher weight, so that close friends can have more bandwidth.

B. SIMULATION ENVIRONMENT (B)

The simulation environments (b) and (a) are basically the same. The main distinction is in the demand time of each requesting peer. In the simulation environment (b), the request time is different at each requesting peer, which is mainly used to observe the change of the data throughput when peers join at different time.

Fig. 11 shows that Video_a of peer n0 is requested by peer n1 at 1.0 second, and Video_a of peer n0 is requested by peer n3 at 11.0 second; Video_a of peer n0 is requested by peer n4 at

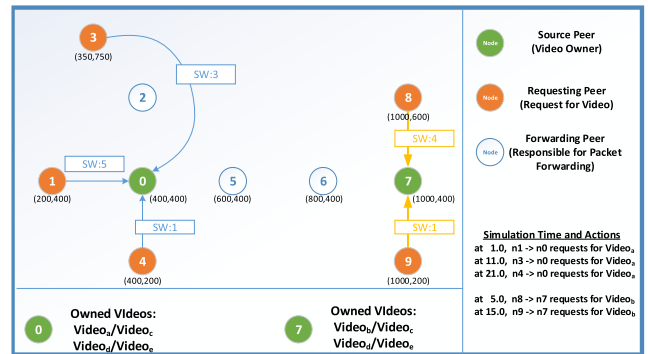


FIGURE 11. The network configuration of simulation environment (b).

21.0 second. Video_b of peer n7 is requested by peer n8 at 5.0 second, and Video_b of peer n7 is requested by peer n9 at 15.0 second. All social weight (SW) values are unchanged, as shown in Fig. 8.

In Fig. 12, the x-axis denotes the simulation time in seconds; the y-axis denotes the data throughput in Mbps; the CBR packet size is 1,000 bytes, and the request transmission time of each peer is 50 seconds. The SW of peer n1 = 5, peer n3 = 3, and peer n4 = 1. Fig. 12 shows that peer n1 has the highest data throughput of about 16.8Mbps in 0-11 seconds, and peer n3 in the 11th second sends a request to join the transmission. When peer n1 sends a request to join the transmission at 11 second, it shows that the data throughput of peer n1 begins to drop at this time. At the 21 second, peer n4 sends a request to join the transmission; at the 51 second, peer n1 finishes the video transmission. At the same time, the data throughput of peer n3 and n4 are increasing. At 41 second, the peer n3 also finishes the video transmission. At the same time, all the bandwidth was given to peer n4 and the data throughput went straight up.

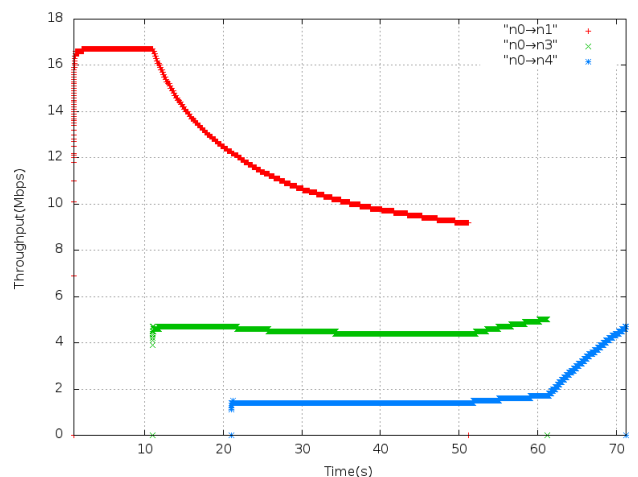


FIGURE 12. The changes in data throughput of peers n1, n3, and n4 in simulation Environment (b).

In Fig. 13, the SW of peer n8 is 4 and peer n9 is 1. It shows that peer n8 has the highest data throughput of about

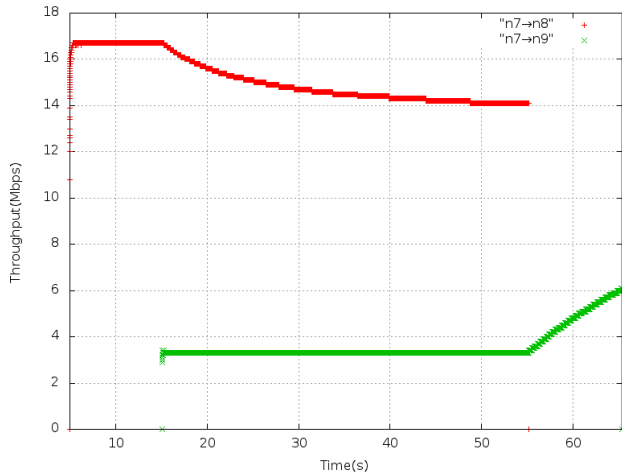


FIGURE 13. The changes in data throughput of peers n8 and n9 in simulation environment (b).

810Mbps in 0-15 seconds, and peer n9 sends a request to join the transmission at 15 second. Fig. 13 shows that the data throughput of peer n8 starts to drop at this time. After the peer n9 joins the transmission, the peer n8 still keeps a certain level of bandwidth. At 55 second, peer n8 finishes the transmission. At same time, peer n9 has all the bandwidth and the data throughput went up in a straight line. It shows that the data throughput of peer n8 slowly decreases to the proportion of weight allocated by the SW. Finally, the WFQ allocates the available bandwidth proportional to the SW value.

C. SIMULATION ENVIRONMENT (C)

The simulation environment (c) mainly observes the changes in data throughput when the request peer's receiving contribution (RC) is greater than *RCThreshold* and the SW value is increased by one.

In Fig. 14, at 1.0 second, the requesting peer n1 sends a request to the source peer n0 and the corresponding SW is 3;

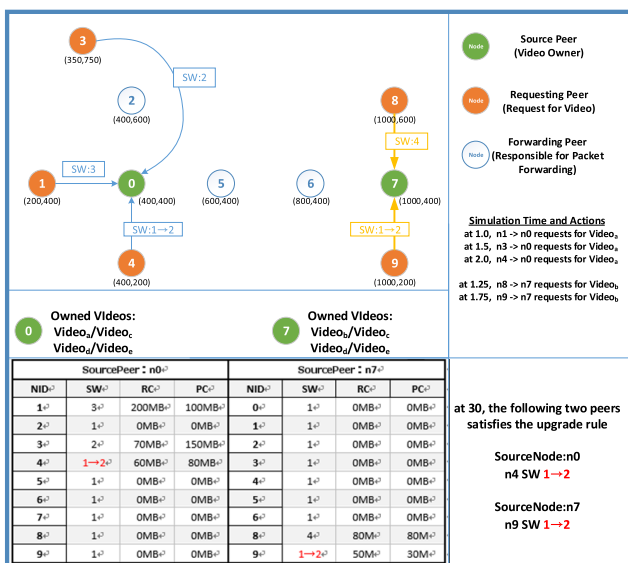


FIGURE 14. The network configurations of simulation environment (c).

at 1.5 second, the requesting peer n3 sends a request to the source peer n0 and the corresponding SW is 2; at 2.0 second, the requesting peer n4 sends a request to the source peer n0 and the corresponding SW is 1, the SW is raised to 2 at 30 second. At 1.25 second, the requesting peer n8 sends a request to the source peer n7 and the corresponding SW is 4; at 1.75 second, the requesting peer n9 sends a request to the source peer n7 and the corresponding SW is 4, the SW is raised to 2 at 30 second

In Fig. 15, the SW of peer n1 is 3, peer n3 is 2, and peer n4 is 2. At 30 second, peer n4's receiving contribution (RC) is greater than the *RCThreshold* value, and the corresponding SW of peer n4 is increased from 1 to 2. Meanwhile, the data throughput of peer n4 is increased to 2.9Mbps. At the end, each transmission will continue to transmit data packets according to the bandwidth allocated by the SW.

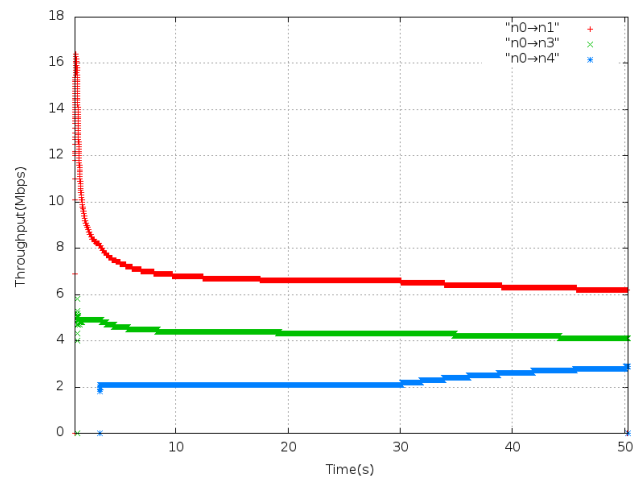


FIGURE 15. The changes in data throughput of peers n1, n3, and n4 in simulation Environment (c).

In Fig. 16, the SW of peer n8 is 4, and peer n9 is 2. At 30 second, peer n9's receiving contribution (RC) is greater than the *RCThreshold* value, and the corresponding SW of

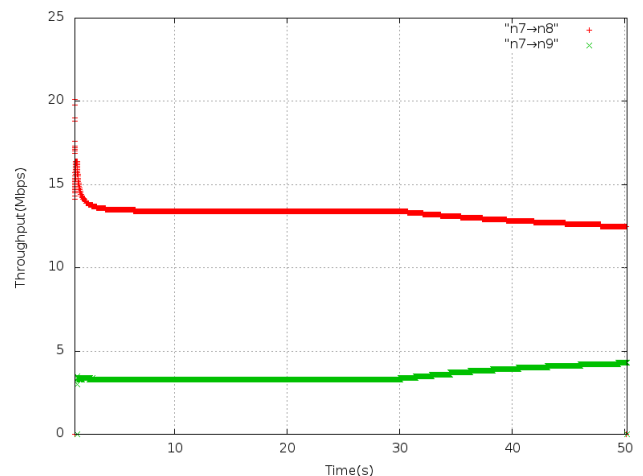


FIGURE 16. The changes in data throughput of peers n8 and n9 in simulation Environment (c).

peer n9 is increased from 1 to 2. At the same time, the data throughput of peer n9 is slowly increased from 1.7Mbps and the data throughput of peer n8 is decreased from 13.6Mbps. At the end, each transmission will continue to transmit data packets according to the bandwidth allocated by the SW.

D. SIMULATION ENVIRONMENT (d)

The simulation environment (d) mainly observes the changes in data throughput when the request peer’s download progress (DP) is greater than $DPThreshold$ and the SW value is increased by one. If the peer n4’s and peer n9’s receiving contribution (RC) are greater than $RCThreshold$, and the corresponding SW values are increased after calculating the social weighted contribution value (SWCV). At 90 second, the download progress (DP) of peer n4 and peer n9 is greater than 90% of the $DPThreshold$ setting, and the SW value is increased by one after calculating the TFFV value. The social weight (SW) changes are shown in Table 4.

TABLE 4. Changes of Social Weight (SW) in Simulation Environment (d).

Source Peer : n0				Source Peer : n7			
Social Weight (SW)				Social Weight (SW)			
NID	Initial	RC(40s)	DT(90s)	NID	Initial	RC(40s)	DT(90s)
1	3	3	3	8	4	4	4
3	2	2	2	9	1	↗2	↗3
4	1	↗2	↗3				

In Fig. 17, at 40 second, peer n4 reaches the SWCV level up condition, and the SW of peer n4 is increased from 1 to 2. The data throughput of peer n4 is increased. At 90 second, peer n4 reaches TFFV level up condition, and the SW of peer n4 is increased from 2 to 3, which speeding up the transmission of peer n4. At the end, each transmission will continue to transmit data packets according to the bandwidth allocated by the SW.

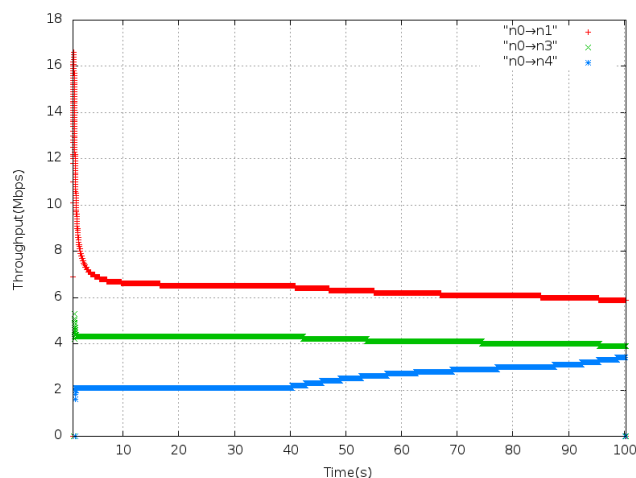


FIGURE 17. The changes in data throughput of peers n1, n3, and n4 in simulation Environment (d).

In Fig. 18, at 40 second, peer n9 satisfies the SWCV level up condition, and the SW of peer n9 is increased from 1 to 2. The data throughput of peer n9 is increased. At 90 second, peer n9 satisfies the TFFV level up condition, and the SW of peer n9 is increased from 2 to 3, which speeding up the transmission of peer n9. At the end, each transmission will continue to transmit data packets according to the bandwidth allocated by the SW.

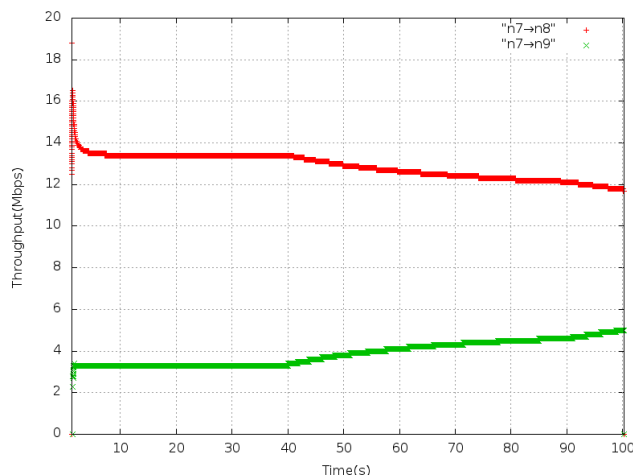


FIGURE 18. The changes in data throughput of peers n8 and n9 in simulation Environment (d).

V. CONCLUSION

In this paper, a social-aware P2P video transmission strategy for multimedia IoT devices is proposed. We introduce a social relationship first video transmission policy that employs weighted fair queue (WFQ) for P2P video transmission according to distinctive queuing priority classes of different social relationships among friends. Each peer has its own friend list, and the social links are classified into different priority classes according to their social relationships, e.g., family, friends, and others. Meanwhile, the proposed strategy calculated the transmission priority according to social relationship, current download progress, and mutual resource sharing contributions. By leveraging the inherent trust associated with social links, the proposed strategy can reduce the impact of free riders and give users good video sharing and watching experiences with multimedia IoT devices.

REFERENCES

- [1] D. Hughes, G. Coulson, and J. Walkerdine, “Free riding on Gnutella revisited: The bell tolls?” *IEEE Distrib. Syst. Online*, vol. 6, no. 6, pp. 1–18, Jun. 2005.
- [2] J. Altmann and Z. B. Bedane, “A P2P file sharing network topology formation algorithm based on social network information,” in *Proc. IEEE INFOCOM Workshops*, Apr. 2009, pp. 1–6.
- [3] Z. Liu, P. Dhungel, D. Wu, C. Zhang, and K. W. Ross, “Understanding and improving ratio incentives in private communities,” in *Proc. IEEE 30th Int. Conf. Distrib. Comput. Syst.*, Jun. 2010, pp. 610–621.
- [4] X. Cheng, J. Liu, H. Wang, and C. Wang, “Coordinate live streaming and storage sharing for social media content distribution,” *IEEE Trans. Multimedia*, vol. 14, no. 6, pp. 1558–1565, Dec. 2012.

- [5] J. A. Pouwelse, P. Garbacki, J. Wang, A. Bakker, J. Yang, A. Iosup, D. H. J. Epema, M. Reinders, M. R. V. Steen, and H. J. Sips, "Tribler: A social based peer-to-peer system," in *Proc. 5th Int. Workshop Peer-to-Peer Syst. (IPTPS)*, 2007, pp. 127–138.
- [6] X. Cheng and J. Liu, "NetTube: Exploring social networks for Peer-to-Peer short video sharing," in *Proc. 28th Conf. Comput. Commun. (IEEE INFOCOM)*, Apr. 2009, pp. 1152–1160.
- [7] Z. Liu, H. Hu, Y. Liu, K. W. Ross, Y. Wang, and M. Mobius, "P2P trading in social networks: The value of staying connected," in *Proc. IEEE INFOCOM*, Mar. 2010, pp. 1–9.
- [8] Z. Wang, C. Wu, L. Sun, and S. Yang, "Peer-assisted social media streaming with social reciprocity," *IEEE Trans. Netw. Service Manage.*, vol. 10, no. 1, pp. 84–94, Mar. 2013.
- [9] Z. Li, H. Shen, H. Wang, G. Liu, and J. Li, "SocialTube: P2P-assisted video sharing in online social networks," in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 2886–2890.
- [10] K. Chen, H. Shen, K. Sapra, and G. Liu, "A social network integrated reputation system for cooperative P2P file sharing," in *Proc. 22nd Int. Conf. Comput. Commun. Netw. (ICCCN)*, Jul. 2013, pp. 1–7.
- [11] P. Rajapaksha, R. Farahbakhsh, S. Mohammadi, M. N. Dailey, and N. Crespi, "Video content delivery enhancement in CDNs based on users' social information," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Washington, DC, USA, Dec. 2016, pp. 1–6.
- [12] N. Shahriar, S. R. Chowdhury, R. Ahmed, M. Sharmin, R. Boutaba, and B. Mathieu, "Availability in P2P based online social networks," in *Proc. 4th Int. Conf. Netw., Syst. Secur. (NSysS)*, Dhaka, Bangladesh, Dec. 2017, pp. 1–9.
- [13] G. Cui, Y. Ren, S. Tian, T. Wu, and K.-K.-R. Choo, "Evolution of resource sharing cooperation based on reciprocity in social networks," in *Proc. 27th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Hangzhou, China, Jul. 2018, pp. 1–9.
- [14] J. C. R. Bennett and H. Zhang, "WF²Q: Worst-case fair weighted fair queueing," in *Proc. IEEE INFOCOM*, vol. 1, Mar. 1996, pp. 120–128.
- [15] T.-L. Sheu and Y.-M. Shiu, "A jitter minimization mechanism with credit/deficit adjustment in IPv6-based DiffServ networks," M.S. thesis, Dept. Elect. Eng., Nat. Sun Yat-Sen Univ., Kaohsiung, Taiwan, 2003.
- [16] W.-S. Hwang and Y.-H. Jan, "The scheduling mechanism for improving WiMAX real-time streaming QoS," M.S. thesis, Dept. Comput. Sci. Inf. Eng., Nat. Kaohsiung Univ. Appl. Sci., Kaohsiung, Taiwan, 2009.
- [17] A. Demers, S. Keshav, and S. Shenker, "Analysis and simulation of a fair queueing algorithm," in *Proc. SIGCOMM Symp. Commun. Archit. Protocols*, New York, NY, USA, 1989, pp. 1–12.
- [18] M. Shreedhar and G. Varghese, "Efficient fair queueing using deficit round-robin," *IEEE/ACM Trans. Netw.*, vol. 4, no. 3, pp. 375–385, Jun. 1996.
- [19] A. Proskochylo, M. Zriakhov, and A. Akulynichev, "The effects of queueing algorithms on QoS for real-time traffic in process of load balancing," in *Proc. Int. Sci.-Practical Conf. Problems Infocommun. Sci. Technol. (PIC S T)*, Kharkiv, Ukraine, Oct. 2018, pp. 575–580.
- [20] S. J. Fusco, K. Michael, A. Aloudat, and R. Abbas, "Monitoring people using location-based social networking and its negative impact on trust," in *Proc. IEEE Int. Symp. Technol. Soc. (ISTAS)*, Chicago, IL, USA, May 2011, pp. 1–11.
- [21] T. Hernandez, A. Levin, M. Foster, B. West, M. J. Nicholson, and W. Cukier, "The next digital divide: Online social network privacy," Ted Rogers School Manage., Privacy Cyber Crime Inst., Ryerson Univ., 2008. Accessed: May 10, 2019. [Online]. Available: <https://cscs.ryerson.ca/products/the-next-digital-divide-online-social-network-privacy>
- [22] *The Network Simulator—NS-2*. Accessed: Oct. 10, 2019. [Online]. Available: <http://www.isi.edu/nsnam/ns/index.html>
- [23] C. Stadtfeld. *NetSim: A Social Networks Simulation Tool in R*. Accessed: May 19, 2020. [Online]. Available: https://sn.ethz.ch/content/dam/ethz/special-interest/gess/social-networks-dam/documents/jss_netsim.pdf



TZ-HENG HSU received the M.S. and Ph.D. degrees from the Department of Computer Science and Information Engineering, National Cheng Kung University, Taiwan, in July 1998 and June 2005, respectively. He is currently an Associate Professor with the Department of Computer Science and Information Engineering, Southern Taiwan University of Science and Technology. His research interests are in the Internet of Things (IoT), artificial intelligence, wireless and mobile network protocols, peer-to-peer computing and communications, and multimedia streaming.



YAO-MIN TUNG received the B.S. degree from the Department of Computer Science and Information Engineering, Southern Taiwan University of Science and Technology, in July 2016. His research interests are in the Internet of Things (IoT), wireless and mobile network protocols, social networks, peer-to-peer computing and communications, and multimedia streaming.