ICIC Express Letters
Part B: Applications
Volume **9**, Number **4**, April **2018**

ICIC International ©2018 ISSN 2185-2766
pp. **293–299**

AN INCENTIVE BASED PIECE DIFFUSION METHOD CONSIDERING PEER PERFORMANCE IN BITTORRENT

Yusaku Tsukuda[1] and Akihide Utani[2]

[1]School of Engineering
[2]Graduate School of Knowledge Engineering
Tokyo City University
1-28-1, Tamazutumi, Setagaya, Tokyo 158-8557, Japan
g1781809@tcu.ac.jp

Received October 2017; accepted January 2018

Abstract. *BitTorrent, which is one of file sharing softwares using peer-to-peer (p2p), has one problem that its users called "free-riders" are reluctant to upload data. An effective piece diffusion method considering Peer Join and Leave called "PDJL" has been proposed, which could help to solve the problem. However, PDJL does not make effectively utilize the broadband upload bandwidth of high performance peers. This paper proposes an incentive based piece diffusion method considering peer performance in BitTorrent. In the proposed scheme, high performance peers send many pieces and efficiently spread the pieces. The performance of the proposed scheme is evaluated through simulation experiments.*
Keywords: P2P, BitTorrent, PDJL, File sharing

1. **Introduction.** BitTorrent, which is one of file sharing softwares using peer-to-peer (p2p) [1], aims to share files between general users called "peers" by sending files in units called "piece" and provide major free software and open source software, music, movies, commercial applications. One problem of BitTorrent is that its users called "free-riders" [2] are reluctant to upload data in BitTorrent. Studies have been conducted on a reputation-based incentive mechanism (RBIM) [3,4], which could help to solve the problem. A reputation value is a numerical value of the degree of contribution to the network from the past transmission and reception amount of each peer. By allowing the peer having the higher reputation value to get the intended piece more preferentially, RBIM can give peers an incentive to upload much more pieces. However, in RBIM, pieces spread slowly since the download request from new peers will be rejected. There is a method called "PDJL" [5], which is a modified version of RBIM. In this study, we aim to improve the efficiency of downloading pieces using modified PDJL. In the simulation experiments, we demonstrated that downloading speed is faster in our proposed method than in PDJL. The rest of the paper is organized as follows. Section 2 outlines algorithms used in BitTorrent. In Section 3, we describe the proposed scheme based on PDJL and three download request transmission algorithms. In Section 4, the experimental results are reported in detail. Finally, this paper closes with conclusions and ideas for further study in Section 5.

2. **BitTorrent.** BitTorrent uses piece selection algorithms and choke algorithms that adopt ideas of Tit for Tat.

2.1. **Piece selection algorithms.** The order of the pieces to be downloaded affects the download efficiency. In BitTorrent, the download order is determined by four piece-selection strategies. The first piece selection strategy is the intensive strategy. This is the top priority strategy among piece selection algorithms, which means to concentrate on

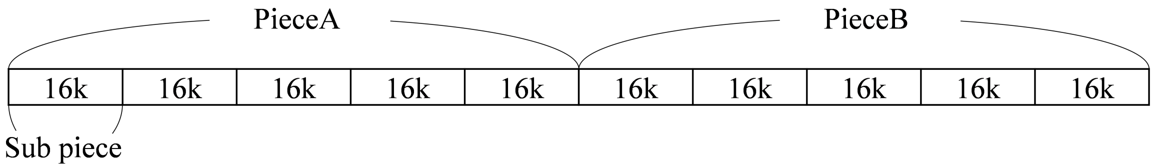
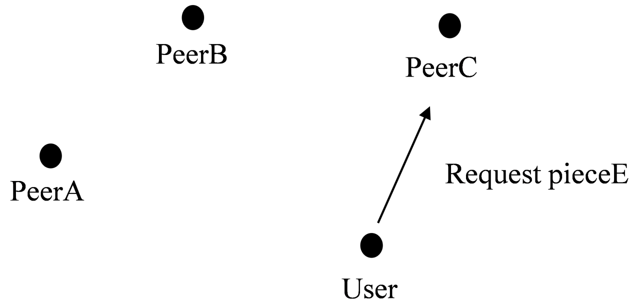


FIGURE 1. Splitting pieces



	PieceA	PieceB	PieceC	PieceD	PieceE
PeerA	○	○	○		
PeerB	○	○		○	
PeerC			○	○	○

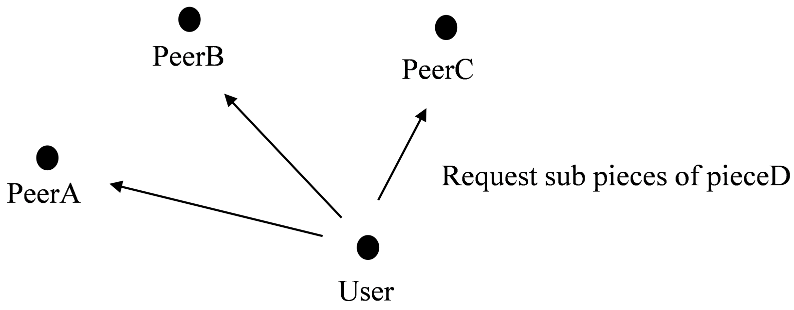
FIGURE 2. Rare piece strategy

one piece. In BitTorrent, one piece is divided into multiple sub pieces for every 16k bytes and then downloaded as shown in Figure 1.

Users download sub pieces from other peers and complete one piece. The second piece selection strategy is the rare piece strategy, in which users download the most rare piece among the pieces other peers have, as shown in Figure 2.

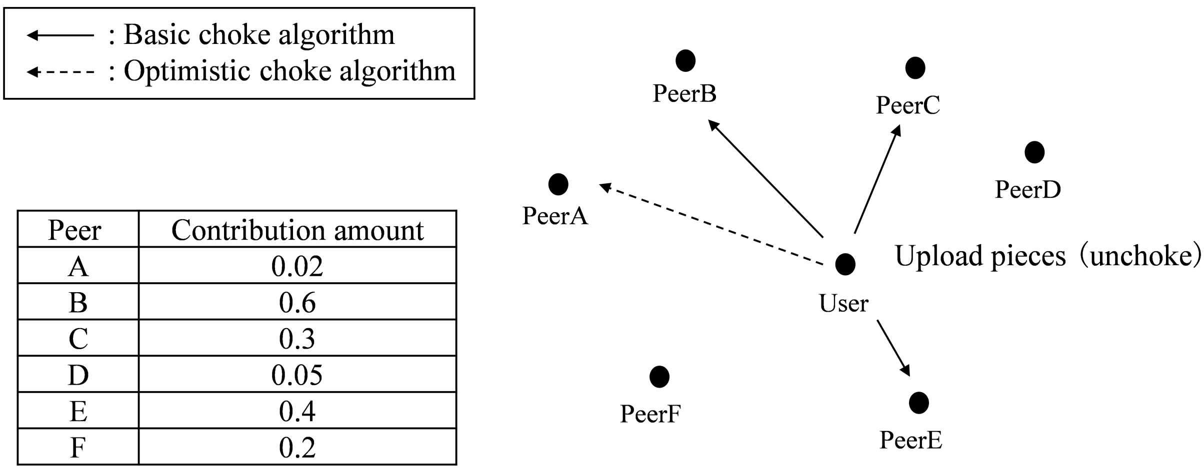
When a peer holding the rare piece leaves, there is a possibility that other peers cannot acquire the piece since other peers do not have the piece. Therefore, this is a strategy to disperse rare pieces. However, there is a case in which rare piece priority strategy cannot be applied. The third piece selection strategy is random strategy. That is right after the BitTorrent client is started. Immediately after starting the BitTorrent client, users have nothing to upload. Therefore, users cannot upload any pieces to other peers at all. In order to upload pieces to other pieces, users must get a complete piece so that it could be uploaded. However, in the rare piece priority strategy, because there are few peers with rare pieces, a new peer cannot download a rare piece from other peers at the same time and transfer efficiency gets worse. On the other hand, even if users download a piece that many peers have, users cannot upload the piece to other peers because every peer has the same piece. Therefore, a new peer uses the random strategy, which is the third piece strategy. The random strategy is used until the first piece is completed, and then the rare piece strategy is used. The fourth piece selection strategy is the End-Game model. If users try to download the last sub piece from a peer with narrow network bandwidth, there is a problem that downloading will be delayed. In order to avoid it, a user requests to download the last sub pieces to all the connected peers, as shown in Figure 3. When a user receives all sub pieces, a user sends a download cancel request to the peers to which the download request was sent, and ends the download.

2.2. **Choke algorithms.** Choke algorithms adjust the balance between supply and demand of files. BitTorrent uses two choke algorithms. The first choke algorithm is the



	PieceA	PieceB	PieceC	PieceD	PieceE
User	○	○	○		○

Figure 3. End-Game model



Peer	Contribution amount
A	0.02
B	0.6
C	0.3
D	0.05
E	0.4
F	0.2

Figure 4. Choke algorithm

basic choke algorithm. A peer uploads pieces to a certain number of peers (unchoked state). Consider the case that the number of unchoked peers is four. The choice of the peers to be put in the unchoked state is determined by the contribution amount. A peer uses the contribution amount for the past 30 seconds. A peer unchokes the three peers with a large contribution amount. The second choke algorithm is the optimistic choke algorithm. A peer randomly unchokes another peer. A peer changes peers to unchoke every 30 seconds. By choosing the peers randomly, the user can upload the rare pieces to the peers with low contribution amount, as shown in Figure 4.

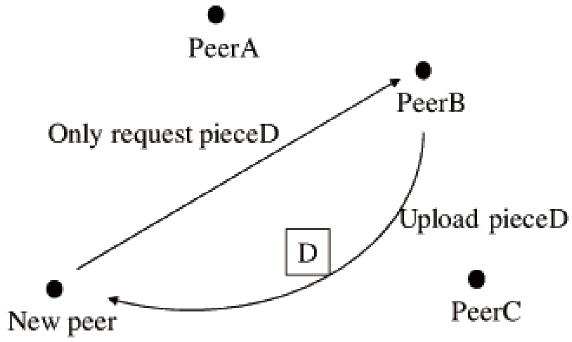
3. **Proposed Scheme.** In the proposed scheme, download efficiency is increased using modified PDJL. In this section, we detail the proposed scheme.

3.1. **Piece diffusion method considering peer join and leave.** In PDJL, a new peer downloads a rare piece from the adjacent peers, which enables the new peer to efficiently exchange piece with other peers from the beginning. The rarity $N_p(t)$ of piece p at time t is expressed as follows

$$N_p(t) = num_p(\Delta t)/\Delta t \tag{1}$$

$num_p(\Delta t)$ is the total number of requests from the neighboring peers to piece p at Δt. The last peer preferentially uploads a rare piece to new peers, as shown in Figure 5.

If the last peers leave, a rare piece can be dispersed.



Peer	The number of owned pieces
A	5
B	**6**
C	1

Pieces of peerB	$N_p(t)$
PieceA	1.2
PieceB	2.5
PieceC	1.5
PieceD	**4.0**

FIGURE 5. Exchange piece of a new peer

R_{ij}	T_{ij}
$R_{AB} = 0.05$	$T_{AB} = -0.1$
$R_{AC} = 0.1$	$T_{AC} = 0.3$

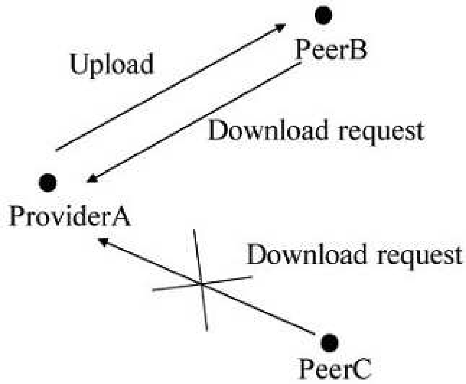


FIGURE 6. A choke algorithm considering the number of owned pieces

PDJL uses a threshold and reputation value when requesting the download of pieces.

$$R_{ij} = \frac{\arctan(\mathit{Maxflow}(j,i)) - \arctan(\mathit{Maxflow}(i,j))}{2\pi} \tag{2}$$

$$T_{ij} = X_j^2 - \alpha \tag{3}$$

where R_{ij} is the reputation value for piece requester j seen from the piece provider i. T_{ij} is the threshold of piece requester j. Download request of piece requester j is sent to piece provider i when R_{ij} is greater than T_{ij} as shown in Figure 6.

Peers are divided into two groups: normal performance peers and high performance peers. The thresholds of all peers are unified. Therefore, high performance peers are faster in getting all pieces and leaving the network than normal peers. So PDJL does not make effectively utilize the broadband upload bandwidth of high performance peers.

3.2. **Modified PDJL.** We aim to improve the efficiency of downloading pieces throughout the network. Our suggestions are setting the threshold by peer performance. In order to effectively utilize the upload bandwidth and the download bandwidth of high performance peers, set the threshold for high performance peers higher than normal performance peers, which could enable high performance peers to spread more pieces throughout the network and eliminate the difference in piece acquisition completion time with normal performance peers. In the previous study, what is not clear is how a peer picks out other peers he needs at random on purpose. So, we created three methods in the study. The first method is a random selection, which randomly selects peers who have piece he wants. The second method is an unchoke selection, which selects peers he contributed to. The third method is a request-based number selection, which selects peers with few requests from other peers.

4. Experiment.

4.1. Conditions of simulation and parameter setting.

Network in simulation of this study consists of normal peers and free-riders according to [3-8], in which 800 new peers join 200 existing peers. The simulator of this study generates detachment only after the participation of peers has ended while repeating the joining and leaving of peers. The simulation time is called round. One round is two seconds. Table 1 shows the parameter settings.

Table 1. Parameter settings

The number of existing peers	200
The number of new peers	800
The number of seeders	200
The number of pieces	4000
Simulation time: round	2 seconds
Refreshing cycle of connection	150 rounds
Maximum number of connections	50
Maximum number of download requests	10
Maximum upload bandwidth of high performance peers	5 pieces/round
Maximum download bandwidth of high performance peers	10 pieces/round
Maximum upload bandwidth of normal performance peers	1 piece/round
Maximum download bandwidth of normal performance peers	3 pieces/round
The number of high performance peers: The number of normal performance peers	3 : 7
Limiting factor of normal performance peers	0.6
Limiting factor of high performance peers	0.4
Proportion of free rider	30%
Proportion of upload rejection by free rider	80%
The average arrival rate	0.25

4.2. Experimental result.

Based on the simulation environment described above and parameter settings, the results of the average completion round of normal peers, average completion round of high performance peers, average completion round of normal performance peers, a complete round of normal peers by the simulation experiment are shown Table 2, Table 3, and Table 4.

TABLE 2. Random selection

	PDJL	Proposed method
Average completion round of normal peers	2550	2388
Average completion round of high performance peers	1863	2266
Average completion round of normal performance peers	2550	2358
A complete round of normal peers	5707	5641

TABLE 3. Unchoke selection

	PDJL	Proposed method
Average completion round of normal peers	2628	2399
Average completion round of high performance peers	1942	2391
Average completion round of normal performance peers	2335	2311
A complete round of normal peers	5665	5471

TABLE 4. Request-based number selection

	PDJL	Proposed method
Average completion round of normal peers	1637	1980
Average completion round of high performance peers	2194	2033
Average completion round of normal performance peers	2005	1928
A complete round of normal peers	7120	6057

As shown in Tables 2 and 3, average completion round of high performance peers gets worse. However, average completion round of normal peers and a complete round of normal peers and average completion round of normal performance peers are reduced. As shown in Table 4, average completion round of normal peers gets worse. However, average completion round of high performance peers and a complete round of normal peers and average completion round of normal performance peers are reduced. Therefore, these results show our proposed method improves the efficiency of downloading pieces throughout the network.

5. **Conclusions.** In this paper, we have proposed a scheme based on PDJL to improve the efficiency of downloading pieces and have confirmed its effectiveness. The proposed scheme can provide pieces to many normal peers and speed up a complete round of normal peers in each download request transmission algorithm. In addition, we show the superiority of the modified PDJL by simulation experiments of thousands of peers. Future work includes a detailed reputation of the proposed scheme in largescale p2p networks.

Acknowledgment. This work is partially supported by the Grant-in-Aid for Scientific Research (Grant No. 17K00349) from the Japan Society for the Promotion of Science. The authors also gratefully acknowledge the helpful comments and suggestions of the associate editor and reviewers.

REFERENCES

[1] W. Mazurczyk and P. Kopiczko, Understanding BitTorrent throught real measurements, *Communications*, vol.10, pp.107-118, 2013.
[2] M. Karakaya, I. Korpeoglu and O. Ulusoy, Free riding in peer-to-peer networks, *IEEE Internet Computing*, vol.13, pp.92-98, 2009.

[3] M. Meulpolder, J. A. Pouwelse, D. H. J. Epema and H. J. Sips, BarterCast: A practical approach to prevent lazy freeriding in P2P networks, *IEEE International Symposium on Parallel & Distributed Processing*, pp.1-8, 2009.

[4] R. Delaviz, N. Andrade and J. A. Pouwelse, Improving accuracy and coverage in an Internet-deployed reputation mechanism, *IEEE the 10th International Conference on Peer-to-Peer Computing (P2P)*, pp.1-9, 2010.

[5] S. Takeda, S. Umeda and H. Shigeno, An incentive-based piece diffusion method considering peer join and leave, *Trans. of Information Processing Society*, vol.56, no.2, pp.421-429, 2015.

[6] A. R. Bharambe, C. Herley and V. N. Padmanabhan, Analyzing and improving a BitTorrent networks performance mechanisms, *The 25th IEEE International Conference on Computer Communications*, pp.1-12, 2006.

[7] M. Sirivianos, X. Yang and S. Jarecki, Robust and efficient incentives for cooperative content distribution, *IEEE/ACM Trans. Networking*, vol.17, no.6, pp.1766-1779, 2009.

[8] A. Jafari, M. Talebi, A. Khonsari and G. Sepidnam, Maximizing download bandwidth for file sharing in BitTorrent-like peer-to-peer networks, *The 14th IEEE International Conference on Parallel and Distributed Systems*, pp.344-350, 2008.