Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 06, June-2019

A Distributed Approach to Improve the Fairness in P2P File Sharing

Deepesh Varshney
Lecturer
M. G. Polytechnic
Hathras, UP, India

Abstract— **Peer-to-Peer System is a file-sharing system in which all nodes act as both server as well as client. It has a fundamental problem of unfairness i.e. how to improve the performance of system. Fairness defined as the ratio of the download rate to upload rate. Free-riders and failure-nodes cause slower download times for others by contributing little or no upload bandwidth while consuming much download bandwidth. Previous attempts to address this fair bandwidth allocation problem suffer from slow peer discovery, inaccurate predictions of neighboring peers' bandwidth allocations, underutilization of bandwidth, and complex parameter tuning. We present Fair-New-Torrent, a new distributed approach that give better and accurately response time in accordance with their simulation.**

Keywords— Peer-to-Peer Network, Bit-Torrent, Fair-Torrent

I. INTRODUCTION

In the world of network we share the information (images, audios, videos, files) among different users or nodes which are connected to each other. In which there is a particular server and so many clients and clients are communicating by the server [1]. The network of personal computers, each of which acts as both client and server, so that can exchange the information, files, emails extras directly with every other user on the network [1]. The network system classify into two things. Figure 1 shows the classification of network system [2].

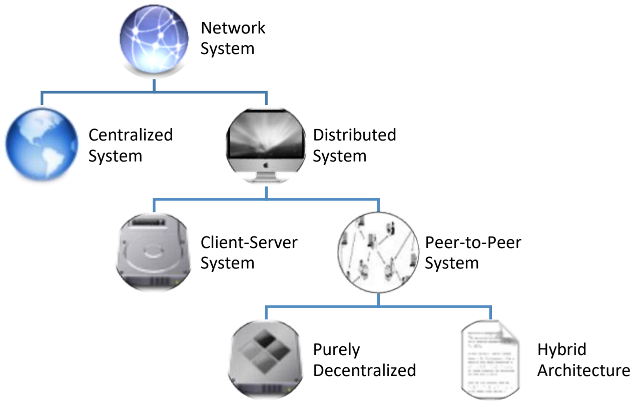

Figure 1: Classification of Network System

The main components are Client-Server System and Peer-to-Peer System have own distinguishes. These distinguish is defined in the table 1.1 [6]. Both systems are decentralized and also have distributed the application among the nodes of system. The comparison based on their application, working and parameters. Table 1 shows the comparison of both kind of system viz. Client-Server System and Peer-to-Peer System [7]. There are many kind of P2P architecture, which is categorized on the basis of their parameters.

Parameters	**Client-Server System**	**Peer-to-Peer System**
Server	One particular Central Server (one server and handles all the clients)	Act as both Client and Server (Sharing the information among them)
Resources	Central server handles all security and file transaction.	Each node share its own resources and own security
Implement	More expensive to implement (require a central file server)	Not so expensive (need a dedicated machine, Server software)

Table 1: Comparison of Client-Server System and Peer-to-Peer System

P2P systems are growing by attracting millions of users and expanding into many application areas. Although P2P files haring is now an inherent part of our computing network. File sharing applications are suffered by a fundamental problem of unfairness in the network system. How bandwidth among peers is used and allocated. Peers do not receive service equally with what they contribute to the system. Unfairness happens by many performance problems. The peers who do not contribute the services to others in the network. These peers are known as free-riders. The numbers of free rider who have upload bandwidth to zero or a small value; take as much as possible from the system while contributing little resources. And high-contributing peers often discover slow download times in the presence of free-riders. Thus, a peer cannot guarantee their own performance by increasing share among the nodes. The Performance is determined by the ratio of upload bandwidth to download bandwidth.

Many nodes does not contributing without taking participating the role in the simulation. These nodes are known as failure node or neutral node. By taking an unfair share of information, failure-node cause less download times for contributing peers. If a P2P system could guarantee fair bandwidth allocation, where by fairness we mean that a peer having a ratio of uploading to downloading approximately same, the network system would be able to guarantee a certain level of performance for contributing peers [6][7].

In P2P, the fair bandwidth can be difficult to achieve by following reasons: First is that there is no central entity which do not have the information of recourses. Second is that the size of bandwidth is not available in advance. Third is that Self contributing peer (free-riders) and failure-node may take advantages by no contribute the resources and consume the recourses. The last one is that which node does not contribute the role in share of data in the network. These nodes are known as failure node [8].

From the result of fairness of previous methods, it is noted that the free-riders have the important role in the network and cause the download the file in achieved time. We want to show that in a BT network data-exchange between nodes or peers is fair, that is at any given time the number of blocks (data packets) that a peer has uploaded is close to the number of blocks (data packets) that it has downloaded. So the main objective is that design a P2P system that comes close to ideal fairness. Ideal Fairness means the ratio of upload time to download time equal one approximately.

The objective of our proposed work is improved the performance time from downloading the file from the other in the network system. The proposed approach takes better time for the improved the performance by downloading the file instantaneously.

II. BACKGROUND AND RELATED WORK

Performance measure by terms i.e. time required for data travel from one node to another. That can be categorized into downloading time and uploading time. So these times depend upon the many kind of factor e.g. bandwidth, amount of data travel, connection make to others. So there are two type of strategy which measures the performance by different kind of factors [11][12][15]. The categorization of strategy is illustrated in the figure 2:

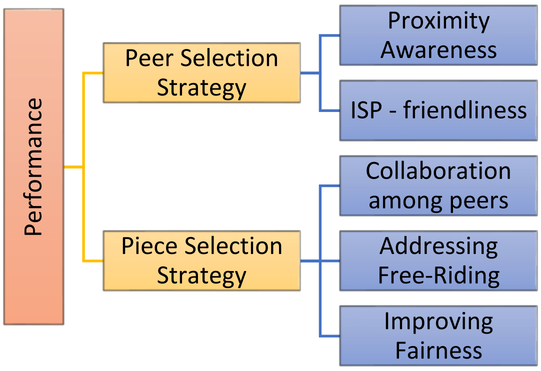

Figure 2: Categorization of strategy of performance in BT

The strategies are defined following as:

1. Peer Selection Strategy: In this strategy, improve the performance of BT by changing the way peers establish connections to others peers while joining and thereafter while maintaining the connections [12]. This strategy categorized into following:

1.1. Proximity Aware technique: BT builds its overlay network by selecting neighbors randomly. It explored the method of using proximity. Aim of exploiting proximity are achieving an efficient usage of network resource and reducing the individual downloading time for the peers [13].

1.2. ISP-Friendliness: ISP stands for Internet Service Provider. Peer chooses its neighbor mostly from peers within the same domain i.e. same service provider within the network [14].

2. Piece Selection Strategy: In this strategy, improve the piece exchange mechanism in BT to further improve download performance, punish free-riders. Free-riders are those peers that only download data from others but not contribute or upload any amount of data to others [15]. And also detect the failure node, which does not share in the networking. This strategy categorized into following:

2.1. Collaboration among Peers: In the network, peers have to promote to others using download rates. By these strategy peers collaborates the amount into network [16].

2.2. Addressing Free-riding: In the network, the previous strategy further, Collaboration among Peers, encourages by punishes free-riders. So find the role of free-rider and punish them or do not use the role of these peers [17].

2.3. Improving Fairness: In the network, find uploading rate and downloading rate for peer to peer [18]. And find out the fairness by using formula which is following defined as:

$$\text{Fairness} = \frac{\text{peer's uploading rate}}{\text{downloading rate of peer to connect with other peer}}$$

So, Performance measure by terms i.e. time required for data travel from one node to another. So we discussed different approaches to improve the fairness or response time to download a file from a server for a client. So the main objective is to improve the downloading time by detect the failure node in the simulation of network. By using the approach form peer selection strategy and take the previous approach to compare the downloading time and also detect the failure case.

In Peer-to-Peer Network system, performance is measured by the bandwidth of peers which are participating in the network system. The bandwidth categorizes into two main components: altruistic and non-altruistic. Altruistic bandwidth is the bandwidth which denoted by certain peers for free to other peers, but there is no expectation of reverse from other peers. Non-altruistic bandwidth is defined as peers to others which are expecting the higher amount of exchange from other peers.

In simple words, the peer who has altruistic bandwidth and also expecting return higher rate. E.g. if a peer has finished downloading the file, then it may stay for uploading the file to other peers, but receiving nothing in return. Infect non-altruistic bandwidth is given to a peer with some amount of data as bandwidth in return.

The main issue of BT's approach is that the exchange of the non-altruistic bandwidth is not fair. For a high-uploading peer, e.g., it may take a long time to discover a subset of other like-uploading peers inside a BT network to exchange data. At the same time, free-riding peers and failure peers, peers that contribute little or no upload bandwidth, can often receive the same amount of download bandwidth as high-uploading peers. Finally, a BT peer is often willing to reciprocate to a neighbor at a higher rate.

Fair-Torrent has exactly the problem of allocation and sharing of the non-altruistic bandwidth and makes sure that the non-altruistic bandwidth is to be fair. By fair allocation, we mean that a peer receives the non-altruistic bandwidth equal to the rate at which it contributes non-altruistic bandwidth to other peers. Moreover, this fairness is achieved on a small time-scale, which allows a peer to achieve a very fast convergence, or a matching download rate, upon its entry into the system. Fair-Torrent avoids long discovery times, and even a high-uploading peer can begin receiving a download rate of non-altruistic bandwidth almost immediately, assuming that there is a subset of other peers, which in combination can match its high uploading rate.

III. PROPOSED IDEA

The new algorithm implements a distributed algorithm that provides better fairness and overcome the number of failure nodes (which are not participating in the network) and also give better service beyond what is already provided by previous algorithm. The proposed idea and algorithm works on the simulation panel. In the simulation panel, construct a network in which defined the number of nodes, their download capacity, download capacity and the amount of file size.

We want to show that data exchange between peers is fair i.e. the amount of data downloaded is close to the amount of data uploaded. There are different approaches to define fairness or performance in Torrent application. Figure 3.1 shows the block diagram of approach i.e. how the algorithm work based on different approach. The main objective is that the comparison between three kind of approach viz. BT approach, FT approach, FNT approach. So we discuss our three approach based on simulation result and determine the downloading time for each approach and compare them by also its average downloading time and also detect the free-riders and failure node in the network. For compatibility, the new approach based on the both BT and FT application. The new approach uses same method to communication. So the previous approaches are following as:

Bit-Torrent Algorithm

The Bit-Torrent (BT) application is based on random neighbor selection approach. In this approach the peers selects as randomly and based on the maximum upload rate of the peers.BT implements an algorithm that provide random period of time for communicating of peers to others. A BT client uploads the data to the other peers by randomly selection. How the peers work in the network and which node act as good peers and which act as different, so this can be define in behavior of node.

Node Behavior: So first describe that how the node behaves by taking different-different bandwidth. In the communication there can be a node or seed which do not contribute the data to others, this node known as failure node. To detect the failure nodes or peers which are not participating in the network.

The approach is defined following as:

i. Take a new variable which increment by the amount of seed’s data.
ii. Compare this variable by given input file size.
iii. Increment the time and again compare by a random variable which used for random peer selection in the network.
iv. Check the random node with other nodes which are participating in the network.
v. Increment the variable by upload amount of seeds which are participating in the network and increment the download amount with download data-rate-change.

Fair-Torrent Algorithm

In the second level of proposed approach, uses BT approaches and define the problem on which fairness can be improved. So the problem based on the free-riders, which nodes only download from others and do not upload i.e. do not contribute in share of data in the network. And another problem based on failure nodes, which do not participate in share i.e. which nodes have all information of data take a new deficit variable which is equal to difference of send data and received data. The difference of send data and received data is known as deficit variable. This deficit variable is defined for count for each iteration. For compatibility, FT approach based on same algorithm with a new variable, this is used for count the difference between send data and received data known as deficit counter. For improve the performance or fairness in BT uses the approach by using a new variable i.e. deficit variable. The approach is following defined as by its node behavior. The node behavior is main part in the simulation panel of the networking and shows the assignment of peers in the sharing of data in the network.

Node Behavior: So first describe the how the node behaves by taking different bandwidth. In the communication find the deficit counter for each iteration or each peer. And for every peer check which have low deficit variable make communication with this.

The approach is defined following as:

i. The approach work as BT approach.
ii. Compare this variable by given input file size.
iii. Increment the time and again pick a random node by selecting lowest deficit variable in the simulation of network.
iv. Check the random node with other nodes which are participating in the network.
v. Increment the variable by upload amount of selected node determining by simulation network.

Fair-New-Torrent Algorithm

In the next level of proposed approach, uses previous approach of BT and FT and then compare the result of downloading time and average downloading time. By using BT approaches and FT approaches define the problem on which fairness can be improved. So the problem based on the failure case, our motive is that how to fair or improve the performance of P2P network. The failure nodes does not contribute in the network i.e. do not contribute in share of data or information in the network. And another problem based on free-rider, which do not upload to the others in the network.

As we know that the difference of send data and received data deficit variable. This deficit variable is defined for count for each iteration. And for each iteration we also count the failure node and take its uploading rate and then subtract from our deficit variable. For compatibility, FNT approach based on same algorithm with a new variable i.e. the amount of data-rate of failure node and then subtract from deficit variable. The approach is following defined as by its node behavior.

The Node behavior based defined the case of failure node of the network and gets the amount of data-rate of this failure case. The approach for the find the failure case defined in the node behavior.

Node Behavior: So first describe the how the node behaves by taking different bandwidth. In the communication find the deficit counter for each iteration or each peer. And for every peer check which have low deficit variable make communication with this.

The approach is defined following as:

i. The approach work as BT and FT approach.
ii. Compare deficit variable and a new variable i.e. the amount of data-rate of failure node and check subtraction of these variables.
iii. Increment the time and again compare by a random variable which used for random peer selection in the network.
iv. Check the random node with other nodes which are participating in the network.
v. Increment the variable by upload amount of seeds which are participating in the network.

IV. PROPOSED ALGORITHM

The objective of our proposed work is to make simulation panel. It covers the number of nodes and its rate of amount of data change. The proposed approach gives more accurate result as compared to previous approach.

Steps of New Approach: There are the following steps which taken for proposed algorithm:

i. Consider the number of nodes.
ii. Assign its amount of data-change-rate.
iii. Assign the amount of file-size.
iv. Increment the amount of upload_size by upload_rate$_i$
v. Increment the amount of upload_data$_i$ by upload_rate$_i$
vi. Increment the amount of download_data$_i$ by download_rate$_i$
vii. Using the algorithm, compare amount of file_size by upload_size.

Detect the failure node as selected_node by using function which is defined in the algorithm.

The block diagram of the approach is shown in the figure 3. In this approach our main motive is that detect the number of failure node. Then take the amount of it and subtract by deficit variable, which is used in the previous approach.

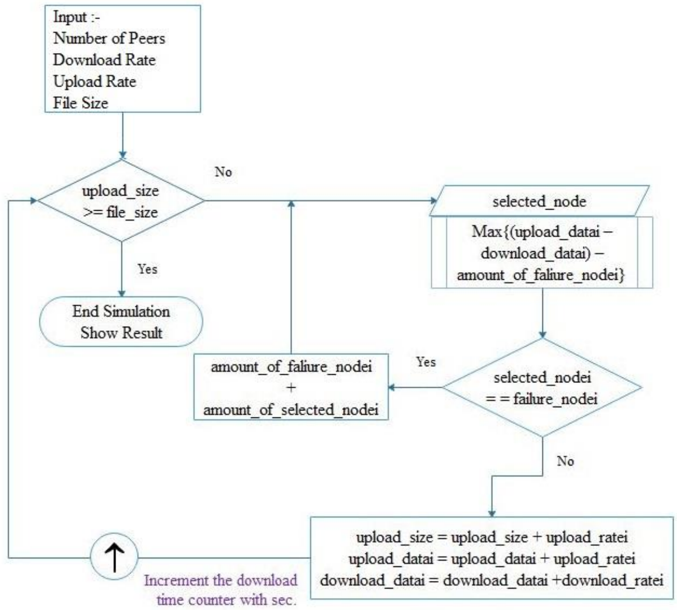

Figure 3: Block-diagram of New Approach

How Algorithm Work: In this approach the main objective is that to detect the failure node, take deficit variable and subtract the amount of data-rate of selected_node (which detect by approach viz. failure node). Take a random failure_node and check selected_node equal to failure_node. If not then increment the amount of upload_size, upload_data, and download_data.

Using this following approach to determine the download time and detect the failure node:

Input: Number of peers, amount of data-rate, amount of file_size

Find the download time and failure node from given peers by using following algorithm:

```
begin while(upload_size == file_size)
        selected_node by Max {(upload_data$_i$ – download_data$_i$) – amount_of_faliure_node$_i$}

begin if(selected_node == failure_node)
        amount_of_faliure_node$_{i+1}$=amount_of_faliure_node$_i$ + amount_of_selected_node$_{i+1}$
        else
                upload_size = upload_size + upload_rate$_i$
                upload_data$_i$ = upload_data$_i$ + upload_rate$_i$
                download_data$_i$ = download_data$_i$ + download_rate$_i$
        end if
        increment the downloading_time
end while
```

V. RESULT

The proposed approach is implemented using MyEclipse7.5 at Intel core i2 processor CPU 2.26 GHz PC with 2 GB of RAM. The results are explained with illustrative examples. The experimental results of existing approaches of downloading time and proposed approach are depicted in the form of tables and graphs.

The proposed approach is tested on simulation Panel. The simulation panel describes the number of nodes which are taken from input and also its data-rate and amount of file size. The values are stored in simulation panel. We have used values as proposed approach for determine the downloading time. So we describe the preliminary simulation setup and result analysis of obtained result. The preliminary simulation setup or we can say environment was created using the Java with the small simulation framework. Initially assign the value for the number of seeds and its data-rate change amount, and take number of leecher as constant i.e. 1, and also assign its data-rate change amount. Considering that Seeds have all the information about data, and then simulation panel uses as multiple leechers and assign the data-rate change amount. So there are two tasks first is that find the downloading time with respect to single leecher. And another is that find the downloading time and also detect the number of failure node with respect to multiple leecher.

Simulation panel likely take following points or parameters shows in table 2.

1	Number of Seeds (participating in the network)
2	Amount of data-rate change
3	Number of leechers (Initially take one and then multiple)
4	Amount of data-rate change
5	Size of File

Table 2: Parameters likely take for detect the failure node

From various numbers of nodes can affect the result for its downloading time. And number of failure node also affect. So the first simulation result compares two approaches with our new one. There are the following graphs which show the various results for different numbers of nodes. And also shows the number of failure node which are not participating in the network.

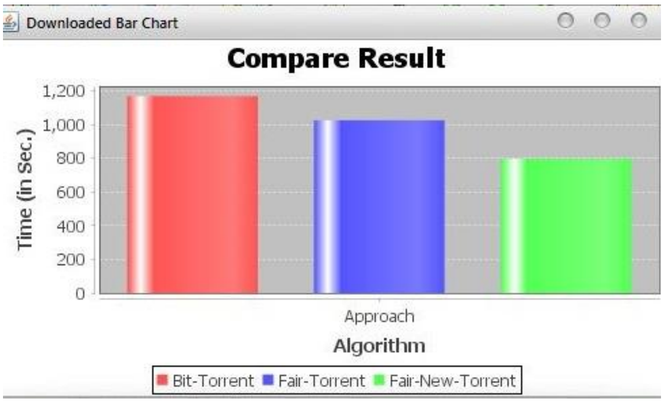

Figure 4: Downloaded Time Graph
(Red- BT Algo, Blue- FT Algo, Green- FNT Algo)

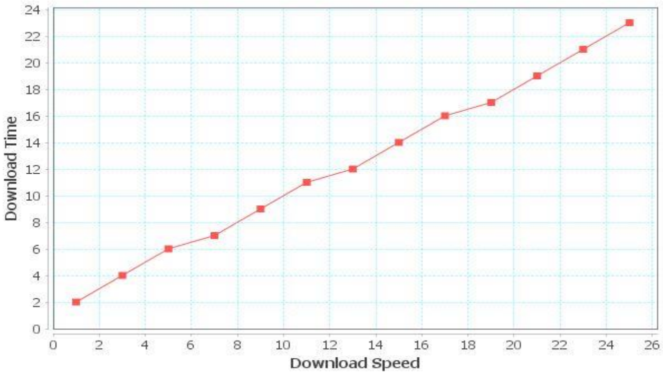

Figure 5: Downloaded Time Graph for FT Approach

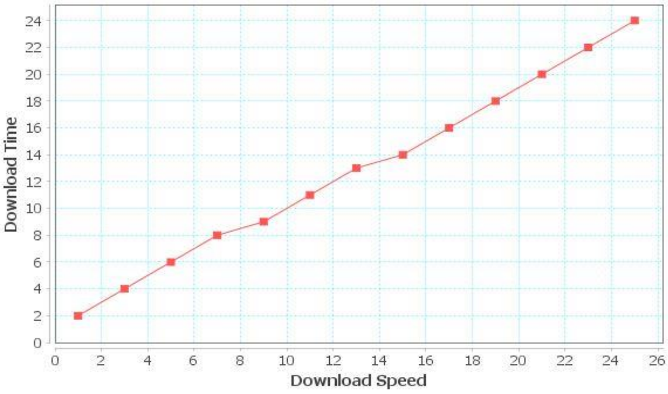

Figure 6: Downloaded Time Graph for FNT Approach

Figure 4 shows the Bar-Chart representation, defines the average downloading time for each approach for rough work of single leecher (data-rate = 5 kbps), 3 Seed (data-rate = 3, 5, 7 kbps), size of file is 5 mb.

Figure 5 and 6 shows the Downloaded-Time Curve, defines the curve download time with respect to varies download speed. The simulation rough work is taken above as defined for bar-chart representation.

VI. CONCLUSION AND FUTURE WORK

In this work, the problem of failure node detection for improved the download time. It addressed for early downloading time for previous approach. Find out the downloading time for different approach is still a difficult and a complex problem in computer science because various issues and challenges like detection of free-riders, failure nodes etc. Most of the researchers have tried to address this problem in various ways by introducing various techniques & approaches and achieved the improvement result and detection of free-riders or free-riders.

For quantitative results, the proposed approach is tested on simulation panel. We calculated two parameters: download time and number of failure node. Download time is the main task to compare approaches by different input. Number of failure case is the next task to improve the download time for proposed work.

The proposed framework has achieved accurate result from previous approach. We have also presented the comparative result analysis with other methods of measure the download time. The calculation complexity of the proposed approach is also easier and accurate than previous approaches.

Detect the number of free-riders with the help of deficit variable using approach of fair torrent and fair new torrent. For various inputs, measure the maximum download time and average download time may be future work for the given proposed approach.

REFERENCES

[1] http://en.wikipedia.org/wiki/Computer_network
[2] http://en.wikipedia.org/wiki/Peer-to-peer
[3] http://en.wikipedia.org/wiki/Peer-to-peer_file_sharing
[4] http://en.wikipedia.org/wiki/BitTorrent
[5] http://en.wikipedia.org/wiki/Glossary_of_BitTorrent_terms
[6] http://mg8.org/processing/bt.html
[7] BitTorrent: http://www.bittorrent.com
[8] L.M.Bertels, Pourebrahimi and Vassiliadis, “A Survey of Peer-to-Peer Networks,” 16th Annual Workshop on Circuits, Systems and Signal Processing, pp.1-8, 2005.
[9] Dongyu, Qiu and Srikant, “Modeling and performance analysis of BitTorrent-like peer-to-peer networks,” ACM SIGCOMM Computer Communication Review, vol.34, no.4, pp.367-378, 2004.
[10] A.P, Bandara, and Jayasumana, “Community-Based Caching for Enhanced Lookup Performance in P2P Systems,” IEEE Transactions on Parallel and Distributed Systems, vol.24, no.9, pp.1752-1762, Sept. 2013.
[11] Crowcroft, Lua, Keong and Pias, “A survey and comparison of peer-to-peer overlay network schemes,” IEEE Communications Surveys and Tutorials, vol.7, no.1-4, pp.72-93, 2005.
[12] Bandara, Dilum, and P. Jayasumana, “Exploiting communities for enhancing lookup performance in structured p2p systems,” IEEE International Conference on Communications (ICC), vol.1, no.6, 2011.
[13] J. K. Muppala, R. Lei and Xia, “A survey of BitTorrent Performance,” IEEE Communications Surveys & Tutorials, vol.12, no.2, pp.140-158, 2010.
[14] C. G. Lee, K. Kannan and S. G. Koo, “On neighbor-selection strategy in hybrid peer-to-peer networks,” Future Generation Computer Systems, vol. 22, no. 7, pp.732 -741, 2006.
[15] J. K. Muppala, L. Zhang and W. Tu, “Exploiting proximity in cooperative download of large files in peer-to-Peer Networks,” International Conference on Internet and Web Applications and Services, vol.0, pp.1, 2007.
[16] A. Zhang, G. Suwala, J. Medved, P. Cao, R. Bindal, T. Bates and W. Chan, “Improving traffic locality in BitTorrent via biased neighbor selection,” 26th IEEE International Conference on Distributed Computing Systems, pp.66, 2006.
[17] M. Coates and R. Thommes “BitTorrent fairness: analysis and improvements,” Workshop Internet, Telecom, and Signal Proc., 2005.
[18] A. R. Bharambe, C. Herley and V. N. Padmanabhan, “Analyzing and improving a BitTorrent networks performance mechanisms,” 25th IEEE International Conference on Computer Communications. Proceedings, pp.1 -12, 2006.
[19] M. Ahamad and S. Jun, “Incentives in BitTorrent induce free riding,” ACM SIGCOMM workshop on Economics of peer-to-peer systems, pp.116-121, 2005.
[20] D. Wallach, P. Druschel and T. Ngan, “Enforcing Fair Sharing of Peer-to-Peer Resources,” In proceeding of 2nd International Workshop on Peer-to-Peer Systems (IPTPS), February 2003.
[21] D. Epema, M. V. Steen and P. Garbacki, “An amortized tit-for-tat protocol for exchanging bandwidth instead of content in p2p networks,” Self-Adaptive and Self-Organizing Systems, First International Conference, pp.119-128, 2007.
[22] M. Kryczka, R. Cuevas, and A. Cuevas, “Measuring BitTorrent Ecosystem: Techniques, Tips and Tricks, “ IEEE Communications Magazine, vol.49, no.9, pp.144-152, 2011.
[23] D. Pouwelse, J. Epema, M. Hales, Rahman, R. Meulpolder and Sips, “Improving efficiency and fairness in p2p systems with effort-based incentives,” IEEE International Conference on Communication, pp.1-5, 2010.
[24] Alex Sherman, Jason Nieh and Clifford Stein, “FairTorrent: a deficit-based distributed algorithm to ensure fairness in peer-to-peer systems,” IEEE Transactions on Networking, vol.20, no.5, pp.1361-1374, 2012.
[25] A. Sherman, A. Stavrou, C. Stein and J. Nieh, “Mitigating the Effect of Free-Riders in BitTorrent using Trusted Agents,” IEEE Transactions on Networking, pp.1-7, 2008.