Australian Journal of Basic and Applied Sciences, 9(1) January 2015, Pages: 183-193

AENSI Journals

Australian Journal of Basic and Applied Sciences

ISSN:1991-8178

Journal home page: www.ajbasweb.com

A Study of Fuzzy Based Block Selection Stratagem in Bittorrent Like P2P Network

[1]M. Padmavathi and [2]R.M. Suresh

[1]R.M.D Engineering College , Department of Information Technology, M.Padmavathi , PO Box 601 206, Kavaraipettai, Chennai
[2]SMIT Engineering College, Principal , R.M.Suresh , Near Mangadu ,Chikarraiapuram ,PO Box 600 069 ,Chennai

ARTICLE INFO

Article history:
Received 19 September 2014
Received in revised form
19 November 2014
Accepted 22 December 2014
Available online 2 January 2015

Keywords:
Peer-to-Peer (P2P) network, BitTorrent, Download time, Performance

ABSTRACT

As there is no distinction between client and server nodes in a peer-to-peer (P2P) network, nodes (or peers) shares resources with each others. Such setting guarantees that more resources like CPU, storage, and bandwidth are available for other peers when more nodes enter the network. BitTorrent is a predominantly usedfor distributing large files quickly and efficiently.It is a scalable peer-to-peer file distribution mechanism. As per the measurement studies, BitTorrent achieves a remarkable and excellent uplink and downlink utilization, but several issues related tolink utilization in simulation settings, fairness, and choice of BitTorrent's peer and piece selection mechanismscan be enhanced. Piece selection plays a significant role in the Bittorrent protocol, because the Bittorrent system can be viewed as a market where peers exchange their pieces with one another.This paper proposes a comparative study of various piece selection strategies in terms of fairness, link utilization based on fuzzy logic reasonings.The proposed technique is evaluated based on the link utilization and fairness through various scenarios.

To Cite This Article: M. Padmavathi and R.M. Suresh., A Study of Fuzzy Based Block Selection Stratagem in Bittorrent Like P2p Network. ***Aust. J. Basic & Appl. Sci.,*** *9(1): 183-193, 2015*

INTRODUCTION

Peer-to-Peer (P2P) networks are a type of computer network which creates decentralized,dynamic networks and are characterised by self-organizing, symmetric communication and distributed control (Risson, J., Moors, 2006). The P2P systems differs from the traditional client-server networks in a way that the server of traditional client-server systems requires large link capacity to be provided to its clients, whereas in P2P networks resources of each peer are shared with each other. The innate characteristics of P2P applications makes it useful to distribute the content to multiple downloaders over the internet in a scalable way as compared to client-server architecture. In P2P networks, as peers can serve as client and server simultaneously it takes lesser amount of time to distribute the content when compared to a traditional client-server model. The provision of powerful infrastructure for large-scale distributed applications like file sharing results in the popularity of Peer-to-peer (P2P) systems.The studies shows that 43% of Internet traffic is due to P2P traffic. BitTorrent is considered to be highly significant and predominant as it is the major cause for creating more than 50% of all P2P traffic among P2P systems (Parker, 2007), able in delivering large collection of data from original servers to clients. Even though, BitTorrent creates majority of traffic it varies from the traditional P2P application by means of content sharing by peers (Schulz, 2009).

While establishing connection with neighbour, each node downloads and uploads chunksamong neighbours. When a peer completes downloading a file it can become a seed maintaining file spreading like flood among peers. When a new peers join the swarm,the chances of a successful download gets increased. Some of the benefits of Bittorrent are reduced hardware and bandwidthcost, redundancy in case of system problems and reduced dependency on the original source. BitTorrent protocol employsthe local rarest first (LRF) algorithm to pick out the blocks for downloading from neighbours. The blocks which are replicatedin least among neighbours are chosen by LRF. The block diagram of a P2P network is seen in Figure 1.

BitTorrent is successful due to the following features. Firstly, it divides an entire file into blocks and disseminate that blocks instead of the complete file. Secondly, using LRF chunk selection algorithm, users will download the rare block among neighbors. It has been proved that LRF algorithm can maximize service capacity and aid in prevention of last block problems (Bharambe, A.R., 2005). Thirdly , it uses rate-based Tit-for-Tat (TFT) unchoking scheme to discourage freeriders, as freeriders will be choked again and again when uploads to other users failed. Given the significance of BitTorrent, it is important to study its overall efficiency

Corresponding Author: M. Padmavathi, R.M.D Engineering College, Department of Information Technology, M.Padmavathi , PO Box 601 206, Kavaraipettai, Chennai
E-mail: mpadmavathi1979@gmail.com

and the effectiveness of its various piece selection mechanisms. The main goal of this paper is to provide a comprehensive analysis ofBitTorrent piece selectionusing a simulation approach.Simulation-based approaches are useful in understanding and improving BitTorrent performance. Live internet computations like tracker logs (Pouwelse, J.A., 2004), or live Torrent participation are not sufficient to study performance metrics of Bittorrent. As configuration parameters of BitTorrent mechanisms can't be controlled or alternate methods cannot be included, simulators are used for studying BitTorrent performance. The simulator helps to study and model peer activity including peer joining/leaving a torrent and block exchanges. All mechanism like LRF, TFT can be incorporated in detail. By means of associating a downlink/uplink bandwidth for every node, asymmetric network access is modelled. Simulations can be carried out on real data or else it is possible to model scenarios based on hypothesis leading to better understanding of peer to peer networks.

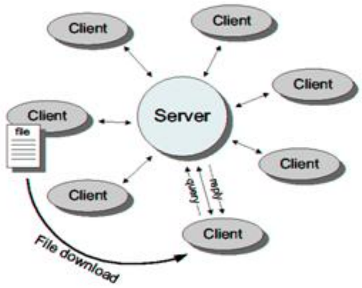

Fig. 1: A P2P network.

This paper aims to investigate the impact of Bit-Torrent's mechanisms and parameters on overall performance. The foresmost key determinant lies in knowing whether BitTorrent can keep all system uplinks active and use it fully. Full use means, optimal mean download time and also optimal load on the origin server. There are many reasons why uplink use might be less. One is that nodes take independent downloading decisions with regard to file blocks,hence neighbours might get a similar set of blocks decreasing and degrading their utility. How the blocks get replicated is governed by the peers using block-selection policy. It is found that the default policy used by BitTorrent namely LRF, may not be optimal in all secnarios. It is a key factor to find out if other policies for selecting pieces work equally well and under what workloads policy choice becomes crucial. TFT confinespeer connections resulting in a scenario where a node decides not to serve its peer, in spite of having useful blocks to serve. Also key BitTorrent mechanisms are dependent on parameters like peers number which interacts with each node and maximum permitted uploads. These mechanisms interact differently and performance could be influenced by specific parameter settings. In this paper, the BitTorrent is simulated for evaluating the effect of core mechanisms on the system under different workloads and different piece selection mechanisms. These piece selection mechanisms are compared with the proposed fuzzy approach to verify the fairness factor by means of utilizing mean upload utilization and mean download utilization under a network of nodes .

BITTORRENT

BitTorrent (BitTorrent, 2000) is a dominant P2Pfile sharing protocol guarantees for fast and efficient replication of single large file to a large set of clients by means of controlling upload bandwidth of downloading peers. BitTorrent breaks thethefiles into equal sized blocks and allow the nodes to join in a 'swarm' in order to upload and download the files at the same time. The BitTorrentprotocol organizespeers in an overlaynetwork called*Torrent*.Seperate 'torrent' is maintained for each file being distributedwhich are available in blocks ranging in size from 32 to 256 KB. Furthermore tracker and a web serverplays a vital rolein content distribution of BitTorrent besides a peer. A tracker is a special central node responsible for recording information periodically about nodes joining / leaving a torrent. The tracker does not participate in the original file distribution but it keeps meta-information about the peers which are active currently and acts as a common communication point for all the peers of the torrent. Part of a torrent are peers which can be either seeds or leechers. A seed is a peerwhichdedicates itself in serving other peers by means of remaining in the torrent and has an entire copy of a file. Aleecher is a peer which has only part of file meansthat it is still involved in downloading activity of a file. There must be mimimum one initial seed in order for the torrent to get started. A special file called .torrent extension is readily available on a web server which incorporates information about the file including piece length, name, hash value, and the tracker URL. As an initial step, a peerdesiring to download a file gets the related .torrent file from the web server.

As illustrated in studies ,a BitTorrent peer operates at the following 3 levels namely Peer selection at the macroscopic level, Choking / Unchoking at the mid level, and *Piece selection* atthe microscopic level. During peer selection (or) neighbor selection process the peer communicates with the tracker to obtain a random list of IP address and port number pairs of peers that have already joined and contributing in the torrent. The tracker will send the requesting peer , the list of around 50 adjoining peers that are active in a peer set.Among the neighbours list, the peer contacts around 25 to 40 in order to join them as its neighbors and establishes connection with them. As the neighbors countof a peer falls below the specified range roughly around 20, the peer will once again contact the tracker to obtain a peer list. Once the connection got established, peers can share file pieces among them. Fig 2.shows the various steps of file sharing . When complete piece is downloaded, its hash value is calculated using SHA and compared with the original value in the metafile(.torrent). If there is a match in the values, the peer will broadcast theexistence of the complete block orpiece to its neighbours.The peer choking/unchoking is the resolution process created bya peer to decide aboutthe willingness of data uploading to an interested neighbor (peer). First it sends choke(block)message to most of its peers in the neighbor list, means that it refuses to send the data. It then sends unchokemessage to a minimum numberof neighbours (leechers) defaultly 4 which provides best downloading rate and also from whom it downloaded several chunks.The above behavior is encouraged by an algorithm called Choking Algorithmwhich is a periodical operation where once in every 10 second a leecher selects other leechers from its neighbor list to upload chunks. In addition to the normal unchoking operation BitTorrent also performs the operation called optimistic unchoking once in every 30 seconds by means of sending unchoke message to a random peer, which is needed for the evolution of new peers during bootstrapping process when there is nothing to shareyet.The piece selection policy comes to an effect only when a peer unchokes the leecher. BitTorrent employs the rarest-first policy, when it needs rare piece among peers (i.e) the piece with the minimum interest value . The above steps are repeated at regular time intervals.The renewal to a peer selection is carried out if the neighbor count drops below a certain threshold value (20) due to peer leaving. The choking/unchoking process is continued at every 10 s and optimistic unchokingis repeated at every 30 s. Thepiece selection is done once an unchoked connection isavailable again to download a fresh piece.

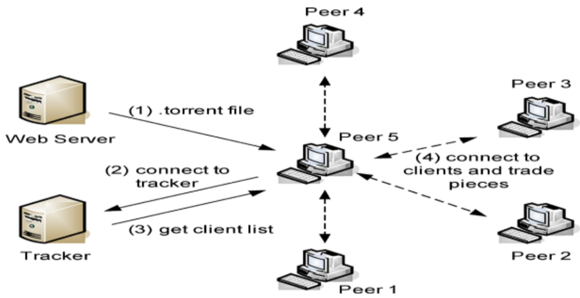

Fig. 2: File Sharing Steps of Bittorrent Protocol.

FUZZY LOGIC CONCEPT

Fuzzy logic is a part of fuzzy set theory which deals with the knowledge representation and inference from knowledge.Unlike other logical systems, fuzzy logic deals with vague or uncertain knowledge and developed using mathematical principles based on degrees of membership.In classical sets that there is no uncertainty and hence they have crispy boundaries, but in the case of a fuzzy set, certain level of uncertainty is allowed and hence the boundaries may be specified ambiguously. Hence we can conclude that the fuzzy logic has the ability to impersonate the human mind to efficiently employ the method of reasoning that are approximaterather than exact.Unlike double valued (0 or 1) Boolean logic, fuzzy logic is multi valued and contends with degrees of membership function and degrees of truth vales. It uses the range of logical values between 0 (completely false) and 1 (completely true). The basic building blocks of fuzzy logic are linguistic variables which represents the imprecision existing in the system . They may be defined as variables whose values are expressed as words or sentences.The various logical operations on fuzzy logic are union,conjunction and impliccations , where union of sets are denoted by ‘OR’ and conjunction of sets are denoted by ‘AND’ and implication operation is denoted by ‘IF .... THEN ‘. Consider 2 fuzzy sets A and B defined in space X, the union of the two sets are represented by a set C where C⊂X, then the corresponding membership function is defined as:

$$\mu_c\left(x\right)=Max\left[\mu_A\left(x\right),\mu_B\left(x\right)\right] \quad (1)$$

where $\mu_A\left(x\right)$ is membership function of set A and $\mu_B\left(x\right)$ is membership function of set B.

For the conjunction operations of sets A and B, the corresponding membership function is

$$\mu_c\left(x\right)=Min\left[\mu_A\left(x\right),\mu_B\left(x\right)\right] \quad (2)$$

For Implication operation the correponding membership function is defined as:

$$\mu_{A\rightarrow B}(x,y)=\mu_R(x,y)=min\left[\mu_A(x),\mu_B(x)\right] \quad (3)$$

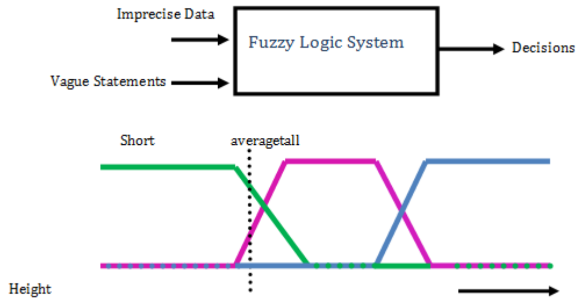

Fig. 3: Concept of Fuzy logic.

The major steps in fuzzy algorithms are: Definition of input / output fuzzy variables , Fuzzification , Inference, Rule specification and Defuzzification. The input / output fuzzy variable can be either a system variable or linguistic variable. In the above diagram the variables short,average,tall are called as the linguistic variables which represents the imprecisions existing in the system. Fuzzy logic system maps crisp intputs to crisp outputs. The central component of fuzzy logic are fuzzy inference system (FIS) which maps a nonlinear input data into a scalar output using fuzzy rules and is responsible for approximate the reasoning and deducing control actions. The mapping process includes input/output membership functions, Use of fuzzy logic operators, fuzzy if–then rule base, aggregation of output sets, and defuzzification. A general model of a fuzzy inference system (FIS) includes four components namely the fuzzifier, inference engine, rule base, and defuzzifier. The rule base includes linguistic rules that are designed by an experts. The rules can also be extracted fromnumeric data. Once the rules have been set up, the FIS can be deliberated as a system that mapsan input vector to an output vector. The fuzzifier maps real world input variables into corresponding fuzzy sets by determining the degree to which they belong to each of the fuzzy sets using membership functions. The result of the fuzzification is a fuzzified data which represents the level of truth of each of these linguistic variable. The inference engine maps input fuzzy sets into an output fuzzy sets. It involves determinationof the degree to which the antecedent is satisfied for eachrule.One or more rules may be triggered at the same time.During the aggregation process all the rules are combined into a single fuzzy set.The final step is a defuzzification which converts a output fuzzysets to a crisp number.

RELATED WORKS

Bharambe and Herley (2005) independently developed an abstract simulator and conducted a simulation-based Bittorrent study which ruined the system's design and estimates the effect of its core mechanisms both separately and by grouping, under a diversity of workloads on the entire system performance. The most substantial metrics including link utilization, file download time and fairness among peers in terms of amount of content served are focused in this evaluation. The performance of Bittorrrent is validated from the results obtained which seems to be near-optimal with respect to download time, fairness and uplink bandwidth utilization excluding the extreme conditions. The experimentation exposes that the low bandwidth peers are capable to download much more than they upload to the network when the high bandwidth peers are present. It is observed that the rate-based TFT policy is not efficient in preventing unfairness,. The basic modification to the tracker and stricter is done in order to improve the fairness efficiently and named as block-based tit-for-tat policy .

Bharambe, *et al*., (2006) used a Microsoft research simulator for studing the performance of the BitTorrent protocol. The simulator allows variation of BitTorrent parameters and study of BitTorrent performance under various scenarios like bandwidth utilization, efficiency, and fairness.

Chiang, *et al*., (2011) proposed an algorithm called Interest-Intended Piece Selection (IIPS), in order to reduce the last piece problem when a sturdy cooperation among the peers retains. Every IIPS peer favours pieces that has the anticipated interest in it and will maximize the likelihood of being interesting to its cooperating peers when downloaded. It is shown from the simulation results that IIPS produces less occurrence of piece loss under rigorous condsitions and it slightly outperforms the BitTorrent's rarest-first algorithm in terms of piece diversity.

Liao, *et al*., (2007) proposed a mathematical model which is capable of predicting the average file download delay precisely in a heterogeneous BitTorrent-like system. The proposed model is derived with least assumptions, and requires minimal system information. A flexible token-based approach is proposed for

BitTorrent-like systems and applied to trade off among the entire system in order to improve the performance and fairness in case of excessive bandwidth users, by means of setting its parameters accurately. The proposed mathematical model is extended to predict the average file download delays in the token- based system, and illustrates the application of the model in order to determine the parameter factors which attains a target performance or fairness.

Deaconescu *et al*., (2009) implemented a framework for BitTorrent performance evaluation which is used to evaluate and compare the recent real world BitTorrent implementations. It focuses mainly on evaluation and comparison on different executions between variety of BitTorrent clients, each utilizing wide varying performance algorithms and optimization schemes.

Pouwelse, *et al*., (2005) concentrated on four major issues namely download performance, availability, flash-crowd handling, and integrity in order to perform a measurement study on BitTorrent. The objective of this work is to provide better understanding about a real P2P system which provides precise mechanisms to facilitate large community of users, for spotting out the design issues correctly and modeling P2P systems efficiently.

Flavio, *et al*., (2011) proposed BUTorrent for improving the downloading time tremendously. Because of the deficiency in global knowledge and the overlay dynamics the initial phase is reserved in a content distribution (file sharing) scenario. This phase causes delays in arriving at a steady state and hence maximizes file download times which is unclever piece distribution. A new class of seed scheduling algorithm proposed is based on a proportional fair scheme which is implemented using a real file sharing client . The simulation results reveals that the proposed file sharing client (BUTorrent) enhanced the the average downloading time of a standard file sharing protocol by more than 25%.

Han Seung *et al*., (2009) investigated the problem of selecting nodes for server for parallel download in overlay content-distribution networks. A node selection is done on a hypercube-like overlay network, that creates the optimal server set based on the worst-case link stress (WLS) criterion is proposed. The scaling is facilitated to a massive systems due to its efficiency and it does not needs routing information or topology collections or network measurements. The algorithm concentrates mainly on four performance criteria's . Firstly, the congestion level at the bottleneck link is reduced initially which means improving the throughput automatically. Secondly, the total number of links utilized and the total bandwidth of the network resources consumed is less in quantity. Thirdly, it assists in more data exchange over the neighbour nodes which is a main objective of content-distribution systems. Lastly, a low average round-trip time to selected servers is suggested.

Eger, *et al* (2008) recommended two pricing based bandwidth incentive schemes for fast file dissemination. The main parameters focused on the proposed algorithms are explicit price information and the download rates from other peers. The results of the proposed algorithms were compared with static and dynamic states of Bittorrent. The proposed pricing algorithms outperformsbittorrent in terms of fairness.

Ken *et al* (2011) focussed on the result of introducing BitTorrent nodes in different networks in OPNET .The simulation results revealed that the end-to-end delay increases due to the heavy P2P traffic on networks, and the level of congestion is reduced due to the presence of tracker and choke algorithms even though network performance is critical.

Xia, *et al*., (2010) conducted a detailed survey on the Bittorrent performance, and reviewed the findings of the study. Improvements to BitTorrent's mechanism were suggested and summarized in the literature reviews. Based on these reviews, further improvement to the bittorrent performance is suggested by authors.

Izhal,*et al*., (2004) conducted an analysis over BitTorrent based on measurements collected for the duration of five months long period with thousands of peers. The performance of the Bittorrent is assesed through several metrics. They gave a conclusion that bittorrent was realistic and inexpensive as compared to the classical client server based content distribution.

Legout,et.al (2006) provided a thorough measurement-based analysis of the rarest first and choke algorithms. Their studyalso includes a large variety of torrents, which allows us not to be focused towards a particular type of torrent.

EXISTING BITTORRENT PIECE SELECTION POLICIES

It is the process of determining which file pieces to select from which peers, based on the knowledge of the file pieces informations available in peers. The piece selection is done once a BitTorrent client establishes connections with its active potential peers.

Strict Priority Piece Selection:

In this policy ,the pieces are downloaded in the strict order they need to be played back ,starting from the first piece.In a strict priority , when a single sub-piece has been requested, the lingering sub-pieces are requested from the specific piece before receiving sub-pieces from any other piece. Thisensureds getting complete pieces as quicklyas possible.

Table 1: Survey Summary.

Author	Developments & Outcomes	Findings	Improvements needed
Bharambe, *et al*., (2005)	Block-based tit-for-tat policy	Enhance Peer fairness	Poor Bandwidth Fairness.
Xia *et al*. (2006)	Mechanisms to improve the bit torrent performance	Performance improved	Peer Selection , Piece Selection
Chiang, *et al*., (2011)	Interest-Intended Piece Selection (IIPS) algorithm	Less occurrences of piece loss	Deprived Fairness
Yuh-JzerJoung , Hsiu-Lin Huang (2009)	Bandwidth based policy	Random policy yields better link utilization Bandwidth match policy yield better fairness	Poor link utilizationon bandwidth match policy Poor Fairnes on random policy.
Arnaud Legout et. al (2006)	Measurement based analysis of choke algorithms.	Distribute rare piece among peers	Link Utilization is poor
Esposito, *et al*., [13]	BUTorrent	Enhance downloading time	Low Peer fairness
SeungChul Han *et al*.,(2009)	Worst-case link stress (WLS)	Scaling to a huge system, Reduced congestion, less bandwidth usage	Peer fairness not achieved
Eger, *et al* (2008)	Two pricing based bandwidth trading schemes	Fast file distribution, better fairness	Frequent piece loss
Liao, *et al*., (2007)	Predict the average file download delay	Accomplishes a target performance/fairness	Bandwidth fairness is not accomplished

Local Rarest First Piece Selection:

It is considered to be a default piece selection strategy in which the local peer keeps the record of ownerscount of each file piece in its peer set and it always requires the piece that are rare among the neighbours as first in view of keeping the file segements available at all the time even in case of flashcrowd event. Bittorrent allows each and every peer to share their rare file segments in prior in order to improve the overall system throughput.Mostly all P2P systems employ rarest first algorithm with some minor alterationsdue to its distribution efficiency. The rarest first algorithm is preferred for three reasons. First, it increases the probability of usefulness of a BitTorrentnode to its peers. Next, it distributes the file pieces equally in the view of minimizing the risk during a file download period.Lastly, thealgorithmmotivates the Bittorrent peers in downloading variety of pieces from the seed which helpsin improving the rate at which new pieces are insertedinto the group.

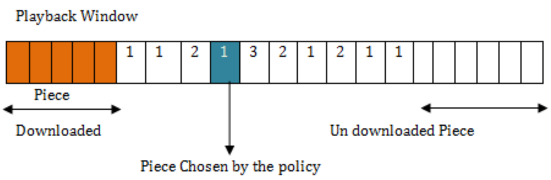

Fig. 4: Local Rarest First Policy.

The rarest first algorithm as seen in Figure 4 works as follows. Each and every peer keeps the record of the total number of replica of each piece in its peer set. This information is used for constructing a rarest pieces set. Let us assume n as the number of copies of the rarest piece, then the index of each piece in the peer set is added to the rarest pieces set. Whenever a piece is added or removed from peer set , the rarest piece set of a peer is updated. A peer selects the next piece to download at random from its rarest pieces set. The pseudocode of LRF is given below:

Local-rarest first segment selection:

UpdateBufferMap()
create array candidate[1..S]
num←0
for i←1 to S
ifBufferMap[i][0] < B and BufferMap[i][Up] > 0
ifnum = 0
num← 1
candidate [1] ← i
else
k← $\sum$ BufferMap[i][j] (j=0..P)

l← $\sum$ BufferMap[candidate[1]][j] (j=0..P)
if k < 1
num← 1
candidate [1] ← i
end if
if k = l
num←num + 1
candidate [num] ← i
endif
endif
endif
endfor
ifnum> 0
i← rand() mod num
request(candidate[i+1])
endif

whereS is the number of segements, B is the number of blocks in one segment and P is the number of local peers. *BuffferMap[i][j]* is the number of blocks in segment i having been received by peer j, peer 0 is the current node and *Up* is the upstream peer. Rarest first policy is better when the number of pieces is more than 4.

The performance of the rarest first algorithm is enhanced by the random first policy.

Random Piece Selection:

It is an another popular piece selection strategy inwhich the local peer always selects the randomly available piece from the remote peers until a complete piece is assembled, when the peer joins a torrent initially. BitTorrentuses thispolicy strictly for first four pieces in order to quickly fill up the initial file pieces and then switches to therarest firstpolicy ensuring downloading its first pieces quickly than the rarest first policy to get a complete piece as fast as possible , which is essential as some pieces are required to reciprocate withTFT Choke algorithm. Although it is used in only in the initial phase to quickly get a piece for sharing, it is considered in this paper as it do not require an extra information such as sorted list as required by LRF.

Bandwidth Based Piece Selection:

In bandwidth based piece selection, the users are classified as high bandwidth users and low bandwidth users. The average download delay of both the group are measured. To study the performance of the high bandwidth users and low bandwidth users , a token system is launched . The primary use of tokens are exchanging blocks between the users. To facilitate trading among users , token tables are maintained by them, which contain thepeer's token informations. When the peer uploads U_{up} bytes to a neighbour peer, the neighbour peer's token lessen by $T_{down}U_{up}$. Alternatively ,neighbor's token value increases by a factor of $T_{up}U_{down}$when a peer download U_{down} bytes by the user from their neighbor. The above method guarantees fairness among all the nodes participating in the networkactively.

Table 2: Comparison of various piece selection methods.

Piece Selection strategy	Findings	Improvements Needed
Local Rarest Policy	Enhanced Upload Utilization	Last Block Problem , Fairness criteria
Random Policy	Better Link Utilization	Fairness
Bandwidth Based Policy	Enhance Fairness	Link Utilization

PROPOSED FUZZY BASED PIECE SELECTION

The Proposed model combines the benefits of various piece selection strategies and improves utilization factor and fairness factor. To utilize the above benefits , it is planned to study the P2P network through a probability based means of uplinks or downlinks bandwidth allocation whenever the nodes join the network. At the time of modelling the connection is presumed to be an ADSL with download bandwidth more than upload bandwidth. A file size of 100 Mb with block size of 256 Kb is considered for simulation, with an initial seed owning a 1024 Kbps bandwidth. At any one time, 1000 nodes are available in an active manner. Each node has a limitations of upto 5 neighbour for uploading files. Efficiency of a Bittorrent is assesed through a formulae , mean utilization of peers uplinks/downlinks over time. Utilization at any point is calculated using the formula , Ratio of an aggregate traffic flow on all uplinks/downlinks to the aggregate capacity of all system uplinks/downlinks. The service capacity of the system is maximum when all network uplinks are flooded. An access link asymmetry treats uplink as the key determinant of performance, eventhough downlink

Australian Journal of Basic and Applied Sciences, 9(1) January 2015, Pages: 183-193

utilization seems to be vital .Theoretical illustration shows that the nodes have boundless capacity, but in reality it demands a huge size of local buffer in order to store the data.

The performance of the proposed architecture is tested against existing block selection policies by increasing the nodes linearly by a factor of 200 in each experimentation, under an assumption that the nodes that join in a network will unrlelate themselves when the total file download gets completed. The performance of the proposed idea is compared against piece selection scenarios like Random policy, Local Rarest First (LRF) and Bandwidth based policy in order to choose the blocks from neighbours.

The proposed technique, employs selection of LRF , Random and Bandwidth based technique based on the availability of blocks and the cooperation of nodes during downloads. The input fuzzy variable selected for this purpose is the available bandwidth of the node and the availability of a specific block across the entire network. The input variables are fed into the fuzzy logic system, and the rules are generated. Based on the rules generated, the proposed method chooses LRF , random or bandwidth based block selection during downloads. The following Figure 5 shows the flowchart of the proposed method.

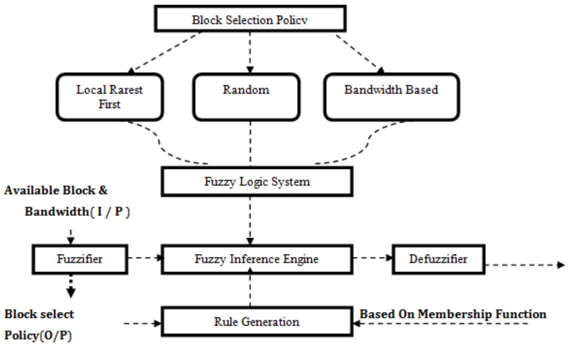

Fig. 5: Proposed Fuzzy Based Piece Selection.

Algorithm:
INPUT: Available Bandwidth, Available block.
OUTPUT: Piece selection technique
STEP 1: Find the degree to which the input belongs in the membership function.
STEP 2:Fuzzify the input
STEP 3 : Apply fuzzy Operators
STEP 4 : Apply fuzzy implicaiton based on rule generation
STEP 3:Aggregate the outputs
STEP 4:Defuzzify

RESULTS AND DISCUSSIONS

At the time of modelling, most of the connections are assumed to be an ADSL with download bandwidth more than the upload bandwidth. The simulation parameters are listed in Table 3, and the usage of uplink and downlink bandwidth is illustrated in Table 4.

Table 3: Simulation Parameters.

Parameter	Value
Connection	ADSL
Available Bandwidth	1024 Kbps
File Size	100 Mb
Block Size	256 Kb
Neighbour count of a node	5
Nodes considered for simulation	100 to 1000 nodes

Table 4: Bandwidth Availability of Each Node.

Download Bandwidth (Kbps)	Upload Bandwidth (Kbps)	Active Nodes Count
64	64	30
384	128	30
896	128	20
1920	128	20

Australian Journal of Basic and Applied Sciences, 9(1) January 2015, Pages: 183-193

The inputvariables choosenfor simulation is, the available bandwidth of the node and the availability of a certain block across the whole network. The Membership Function (MF) uses a bell curve in the midrange as shown in Figure 6 for both the input linguistic variable.

The control strategy of the fuzzy logic system is contained in the rule blocks. Each rule block restricts all rules forsimilar context. The 'if' part of the rules, define the condition for which the rules are designed and the 'then' part gives the output of the fuzzy system for that condition. The importance of the rule is weighed using degree of support (DoS). The processing of the rules starts with calculating the 'if' part. The operator type of the rule block determines which method is used.Therules generated are shown in Table 7.

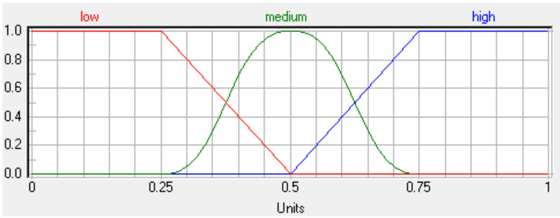

Fig. 6: Input membership function for Block available and Bandwidth.

The definition points of the input membership function are shown in Table 5.

Table 5: Definition Point of The Input Membership Function.

Term Name	Shape/Par.	Definition Points (x, y)		
low	linear	(0, 1)	(0.25, 1)	(0.5, 0)
		(1, 0)		
medium	S-Shape/0.50	(0, 0)	(0.25, 0)	(0.5, 1)
		(0.75, 0)	(1, 0)	
high	linear	(0, 0)	(0.5, 0)	(0.75, 1)
		(1, 1)		

The output membership function for the policy to be chosen is shown in Figure 7 and table 6 shows the output definition point.

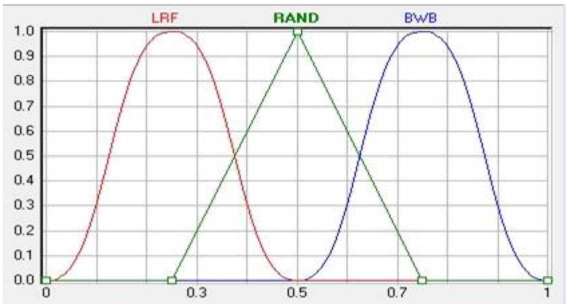

Fig. 7: The policy selected based in input MF.

Table 6: Definition Point of Output Membership Function.

Term Name	Shape/Par.	Definition Points (x, y)		
LRF	S-Shape/0.50	(0, 0)	(0.25, 1)	(0.5, 0)
		(1, 0)		
Rand	linear	(0, 0)	(0.25, 0)	(0.5, 1)
		(0.75, 0)	(1, 0)	
BWB	S-Shape/0.50	(0, 0)	(0.5, 0)	(0.75, 1)
		(1, 0)		

Table 7: Rules Generated.

IF		THEN	
Bandwidth Available	BlockAvailable	DOS	Policy
low	low	0.25	LRF
low	low	0.5	Random
low	low	0.75	Bandwidth Based
low	medium	0.25	LRF
medium	low	0.5	LRF
medium	low	0.5	Random

medium	low	0.75	Bandwidth Based
medium	medium	0.5	LRF
high	low	0.25	LRF
high	low	0.5	Random
high	medium	0.75	LRF

Simulations were carried out after modifying the OCTOSIM simulator. Simulations were carried out with 200,400,600, 800 and 1000 nodes in the network. 10 runs were simulated. Table 8 shows the mean upload and download utilization of the network for the best run.

Table 8: Utilization within the network for best run.

Number of nodes	Mean upload utilization over time (%)			Mean Download utilization over time (%)		
	LRF	Random	Bandwidth Based	LRF	Random	Bandwidth Based
100	95.2	97.2	97.62	37	37.3	42.54
200	97.3	98.24	99.01	42	40.83	41.75
300	97.4	98.24	99.01	43.5	42.5	42.61
400	97.8	98.26	99.05	43.93	43.78	44.01
500	97.9	98.28	99.07	44.02	43.91	44.52
600	98	98.3	99.12	44.21	43.98	44.82
700	98.1	98.32	99.12	44.2	44.14	44.85
800	98.1	98.37	99.12	44.76	44.36	45.08
900	98.4	98.35	99.1	44.81	44.8	45.13
1000	98.4	98.37	99.1	44.82	44.8	45.34

From the simulation results of table 7, it can be seen that the performance of the proposed system gets improved by using the best of the three techniques in terms of improved utilization of the bandwidth for both uploads and downloads. It can be seen that the result is consistent with the linearly increasing number of nodes in the network. Figure 5 and figure 6shows graphical representation of the mean upload and download utilization in the network.

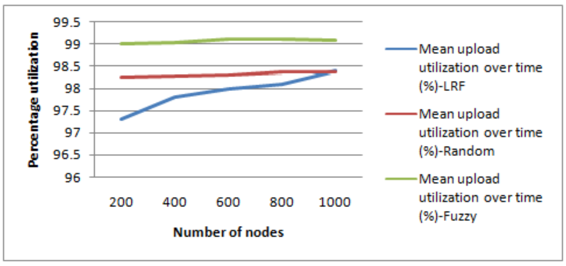

Fig. 5: Mean upload utilization for the best run.

The improvement in the download utilization time is not very significant ranging from 0.71% to 1.76% in the proposed system; however the mean download utilization is higher than other systems tested.

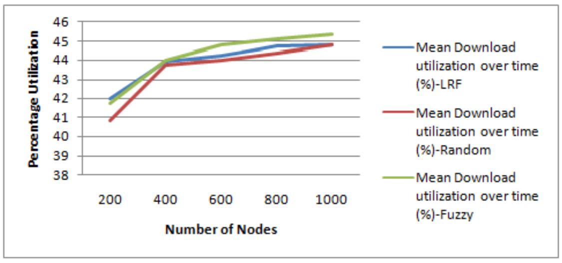

Fig. 6: Mean download utilization for the best run.

The number of nodes in the network is increased from 100 to 1000. The graph depicts the performance of the network with the linearly increasing number of nodes. The nodes jointhe network during 10thsecond and stay there itself till the downloading process gets over. Figure 5 and 6 shows the graphical representation of mean upload utilization and mean download utilization of various block selection policies across various nodes. It shows that the upload utilization of a bandwidth based policy achieved ahigh percentage, above 99% for varying number (100 to 1000) of nodes in the network. These high values of upload indicates that the network performance is optimum with respect to mean download time.

CONCLUSION

In this paper,a bit torrent like network is evaluated for different block selection and studies the same under various scenarios. For evaluation, the LRF policies, random policy and the proposed fuzzy based policy for picking up blocks for download from its neighbor were considered. Simulations arecarried out with network containing 200 nodes to 1000 nodes for various bandwidth considerations.The result obtained shows the upload utilization by the proposed technique is slightly better than random policy and LRF policy. During simulation, it is assumed that bandwidth remains constant during the entire process which may not be true in real time system. Since the network parameters are additive in nature, the proposed solution is NP-Complete. Further work needs to be done to address the NP complete problem.

REFERENCES

Risson, J., Moors, 2006. Survey of research towards robust peer-to-peer networks: search methods. Computer Networks, 50(17): 3485-3521.

Parker, 2007. The true picture of peer-to-peer filesharing. URL: http://www.cachelogic.com/, Accessed February.

Schulz, Hendrik, Klaus Mochalski, 2009. Internet Study. Leipzig, Germany.

BitTorrent, 2000. http://bittorrent.com.

Cohen, B., 2000. Incentives build robustness in BitTorrent. In Workshop on Economics of Peer-to-Peer systems, 6: 68-72.

Bharambe, A.R., C. Herley, V.N. Padmanabhan, 2005. Analyzing and improving BitTorrent performance. Microsoft Research, Microsoft Corporation One Microsoft Way Redmond, WA, 98052, 2005-2003.

Pouwelse, J.A., P. Garbacki, D. Epema, H.J. Sips, 2004. A measurement study of the BitTorrent peer-to-peer file-sharing system. Technical Report PDS-2004-003, Delft University of Technology, The Netherlands.

Liao, W. C., F. Papadopoulos, K. Psounis, 2007. Performance analysis of BitTorrent-like systems with heterogeneous users. Performance Evaluation, 64(9): 876-891.

Deaconescu, R., R. Rughinis, N. Tapus, 2009. A bittorrent performance evaluation framework. In Networking and Services, 2009. ICNS'09. Fifth International Conference on, 354-358. IEEE.

Bharambe, A.R., C. Herley, V.N. Padmanabhan, 2005. Analyzing and improving BitTorrent performance. Microsoft Research, Microsoft Corporation One Microsoft Way Redmond, WA, 98052: 2005-2003.

Pouwelse, J., P. Garbacki, D. Epema, H. Sips, 2005. The bittorrent p2p file-sharing system: Measurements and analysis. Peer-to-Peer Systems IV, 205-216.

Chiang, J. L., Y.Y. Tseng, W.T. Chen, 2011. Interest-Intended Piece Selection in BitTorrent-like peer-to-peer file sharing systems. Journal of parallel and distributed computing, 71(6): 879-888.

Esposito, F., I. Matta, D. Bera, P. Michiardi, 2011. On the impact of seed scheduling in peer-to-peer networks. Computer Networks, 55(15): 3303-3317.

Han, S.C., Y. Xia, 2009. Optimal node-selection algorithm for parallel download in overlay content-distribution networks. Computer Networks, 53(9): 1480-1496.

Eger, K., U. Killat, 2008. Bandwidth trading in bittorrent-like p2p networks for content distribution. Computer Communications, 31(2): 201-211.

Nam, K.K., Y.J. Xu, 2011. Analysis of BitTorrent Protocol and Its Effect on the Network.

Xia, R.L., J.K. Muppala, 2010. A survey of BitTorrent performance. Communications Surveys & Tutorials, IEEE, 12(2): 140-158.

Joung, Y.J., W.T. Wong, H.M. Huang, Y.F. Chou, 2012. Building a network-aware and load-balanced peer-to-peer system for range queries. Computer Networks.

Yuh-JzerJoung, Hsiu-Lin Huang, 2012. A detailed examination of the overlay construction and maintenance mechanism in BitTorrent, Computer Communications 35 (2012) 682–695

Bharambe, A., C. Herley and V. Padmanabhan, 2006. Microsoftresearch simulator for the BitTorrent protocol, http://research.microsoft.com/projects/btsim/.

Izal, M., G. Urvoy-keller, E.W. Biersack, P.A. Felber, A.A. Hamra and L. Garces-Erice, 2004. Dissecting BitTorrent: Five months in a torrentslifetime, in Passive and Active Measurements (PAM), LNCS, 3014: 1-11.

Legout, A., G. Urvoy-Keller and P. Michiardi, 2006. Rarest first and chokealgorithms are enough, in IMC ’06: Proc. 6th ACM SIGCOMM conference on Internet measurement. New York, NY, USA: ACM, 203-216.