Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

A Passive Network Measurement-based Traffic Control Algorithm in Gateway of P2P Systems

Yibo Jiang, Weijie Chen, Jianwei Zheng, Wanliang Wang*, Yanwei Zhao

Collage of Information Engineering, Zhejiang University of Technology, P.R.China
*(*Corresponding author, e-mail: wangwanliang@zjut.edu.cn)*

Abstract: With the progress of peer-to-peer technology, P2P applications have evolved and established themselves as the leading 'growth app' of Internet traffic workload. In order to reserve enough bandwidth for other applications, gateway of P2P systems must control P2P's traffic. In this paper, passive network measurement of P2P traffic is introduced, and network behaviors of some P2P applications are discussed. Then, one model of receiving and forwarding P2P packets is constructed. One algorithm of traffic control in P2P is designed, which includes two important factors: rejection probability of P2P request and disconnection probability of P2P channel. The results of two parallel experiments show that control algorithm of P2P traffic based on network measurement can reserve enough network resource without disabling all P2P packets.

1. INTRODUCTION

From the end of 19th century to the beginning of 20th century, P2P (peer-to-peer) technology has been greatly improved. Applications of P2P systems include file sharing systems such as Napster (Ben et al., 2000), bit-torrent (Bram Cohen, 2007), eDonkey (Oliver et al., 2002), Kazaa, Gnutella, BitComet as well as VoIP systems such as Skype and GoogleTalk. Recently, some traditional distributed applications begin to use P2P technology, such as GRID, IPTV, networked games, etc. P2P systems are so popular that they contribute more than fifty percent to the overall network traffic. Thus, P2P systems have gained a lot of attention from the Internet users, ISPs (Internet Service providers) and research communities.

P2P technology is one of distributed applications, which does not depend on network server. In other words, there is no center node in P2P systems, and P2P's network is organized by its self freely. In such situation, P2P technology can solve the problem of bottleneck of server in data transmission of client/server mode (Andrew *et al.*, 2001).

P2P technology makes resource sharing conveniently. However, the wide-spread use of P2P systems has brought some serious drawbacks to Internet. To maximize speed of resource sharing and transmittability of data, many P2P softwares try to use as much bandwidth as they can. In practice, P2P software often creates more than ten threads simultaneity to send and receive data, and search data source in other nodes continually at the same time. P2P software's actions make network much more congested than before (Savage *et al.*, 1999). ISPs have been put in a dilemma. On the one hand, P2P system applications have resulted in an increase in revenue for ISPs, as they are one of the major reasons cited by Internet users for upgrading their Internet access to broadband. On the other hand, ISPs find that P2P traffic poses a significant traffic engineering challenge (Pablo *et al.*, 2006) (Vinay *et al.*, 2007). P2P traffic always occupies other applications' bandwidth, such as Web, FTP and Email.

In local area networks, people are accustomed to use P2P softwares to share files, play games and exchange movies. Not only P2P systems will acquire much network resource in Internet, but also it will lead to congestion in intranet. Because P2P network is a random overlay network which makes intranet as a part of Internet, P2P systems' network usually through gateway between Internet and intranet. In recent years, some ISPs have prepared to control traffic of P2P applications and many network managers of intranet disabled P2P application directly.

Obviously, data sharing is the basic demand of network users, and P2P systems can make the speed of data transmitting faster than any other traditional applications. Although P2P systems have some drawbacks in traffic, it is especially irrational that we disable P2P applications for no reason other than this simple fact. Furthermore, some P2P systems have modified their protocols and softwares to hide ports of TCP protocol and payload of IP packet, today.

In our opinion, P2P technology can be greatly developed and became the first applications of Internet reveals that it is a very efficient technology in many areas. It is not a good idea to avoid network congestion by simply disable P2P applications. So, we need to find an easy way to control P2P traffic. We will introduce network measurement and some important network parameters in P2P systems first. Then, some different methods of measurement will be classified. And we analyze common behaviors of P2P softwares based on network measurement and find a relationship between P2P traffic and behavior. At last, we will construct a model of P2P traffic and design an algorithm for traffic control. Aim of control algorithm in gateway is to fairly assign bandwidth for P2P applications and other applications.

2. NETWORK MEASUREMENT AND SOFTWARE BEHAVIOR IN P2P SYSTEMS

Main task of network measurement is to measure some parameters of real networks, such as bandwidth, packet delay, drop rate, etc. Then we can evaluate the runtime status of network by these statistic data (Stemm *et al.*, 2000) (Wijata *et al.*, 2000). P2P system's network is one type of overlay networks which based on Internet and local area networks. Thus, network measurement in P2P systems must collect parameters of P2P's network and infrastructure of overlay network.

There are two types of network measurement in P2P systems, active measurement and passive measurement. Active network measurement modifies a P2P software client manually, and connects this client as normal P2P software into P2P network to detect characteristics of entire networks. Thus, this client is seemed as a common node by other P2P softwares among neighbor nodes. Active network measurement often collects IP address, transportation port, request port and sharing resources of other nodes, as well as topology structure, packet delay, upload speed and download speed of P2P network. Passive network measurement usually deploys some detection points in infrastructure network. And it uses special softwares and hardwares to collect information of P2P traffic (Sen *et al.*, 2004). With the developing of P2P software and P2P transmitting protocol, P2P application's TCP port can be automatically changed and P2P packet's payload is encrypted. Passive network measurement's ability has been limited.

Actually fact, network measurement in P2P systems is to detect network behaviors of softwares which implement P2P protocol. Saroiu et al. analyzed two P2P systems, Napster and Gnutella (Saroiu *et al.*, 2002). Pouwelse et al. analyzed BitTorrent which is the most popular P2P system today (Pouwelse *et al.*, 2004). By their researches, we find such a common characteristic of behaviors that all types of P2P softwares will send many request packets to its neighbor nodes which are detected before, while a new node is connecting into P2P network. And neighbor nodes will send back response packets which contain their local information to new node after they receive a request packet.

Although, different types of P2P softwares may use different searching methods of neighbor node and collection algorithms of network information, they must send a request packet to other node for handshake before data sharing, as figure 1. If request of data sharing is failed, data transmitting between source node and destination node will not be constructed. At runtime of P2P softwares, they will send many request packets to other nodes, and search new neighbor nodes to increase transmitting. We have traced the bandwidth of P2P traffic in a local area network, as figure 2.

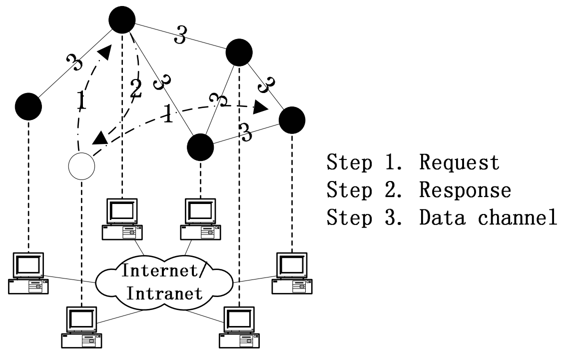

Fig 1. Request and response process of P2P softwares

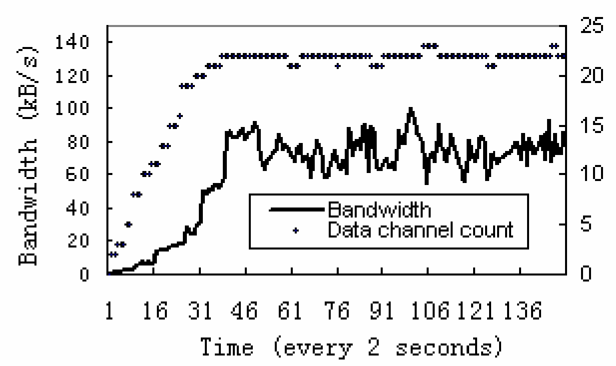

Fig 2. Bandwidth of P2P data sharing in LAN

Figure 2 shows that P2P software will search other neighbor nodes by flooding like method and find out as many resources as possible. Because of random disappearance and appearance of neighbor nodes in P2P network, P2P software would update neighbor information dynamically. If we can disable request packets (Laurent et al., 2006), gateway would prevent construction of P2P data transmitting channel and P2P software's data sharing process will be controlled. Hence, we should design a control algorithm to control P2P traffic and make it works without influence other applications.

3. CONTROL ALGORITHM OF P2P TRAFFIC

Control algorithm is based on network measurement, and it will control P2P traffic in gateway's packet filter. The main task of control algorithm is to control and limit P2P traffic that guarantee other applications would assign enough network resource fairly. There are two control methods. The first method is limitation of P2P request packet depending on gateway's dynamic load in time. The second method is disconnection of some P2P data transmitting channels which have been occupied more network resource when gateway is overloaded.

Control algorithm needs some parameters of network measurement, such as type identification of every workstation's packet, statistic of traffic in delta time, sending and receiving rate of packet, request packet identification. Number of workstations in a local area network is n, and speed of data transmitting of every workstation is $v_i(i=1\cdots n)$. v_i is calculated by network measurement module, which is implemented in gateway. Network measurement module can record every packet through gateway, and identify which one is P2P data packet. A packet can be identified by its TCP port and payload information. By every speed v_i, current total speed of P2P traffic in gateway is V, as follows:

$$V=\sum_{i=1}^{n} v_i \tag{1}$$

Network manager could custom P2P's workload of gateway depending on real network's status, and range of load is $S \in [0, B]$. So bandwidth of P2P traffic should be controlled in S. Moreover, this range is an estimated range. Real bandwidth of P2P traffic may beyond the upper limit in a short time.

Gateway should decide which request packet is allowed to forward while a packet of P2P traffic is received from workstations. If a request of P2P system is allowed, it will be forwarded immediately. Otherwise, the request packet will be dropped directly. All types of P2P softwares have fault-recovery mechanism of timeout. If a request is timeout, the data transmitting channel will not be constructed. Control algorithm in gateway can limit P2P traffic by dropping request packet. Control algorithm use a random method to reject request packet by rejection probability of request α, as follows:

$$\alpha(V)=\frac{\left(\frac{V}{B}e\right)^t}{\exp\left(\frac{tV}{B}\right)} \tag{2}$$

t is a constant. Network manager can change t to modify rejection probability of request.

Every workstation's P2P traffic in previous span Δt is accumulated in $D_i(i=1\cdots n)$, and proportions of total bandwidth is $N_i(i=1\cdots n)$, as follows:

$$N_i=\frac{D_i}{\sum_{j=1}^{n} D_j} \tag{3}$$

Every node's percentage of current total bandwidth of P2P traffic is $M_i(i=1\cdots n)$, and $M_i=v_i/V$. Every node's disconnection probability of P2P data transmitting channel is $P_i(i=1\cdots n)$, as follows:

$$P_i(M_i,N_i)=\frac{\exp(t-tN_i)}{1+w}N_i^t+\frac{\exp(t-tM_i)}{1+w}wM_i^t \tag{4}$$

w is the weight ratio of M_i and N_i, and it is adjusted by network manager.

Modified disconnection probability of P2P data transmitting channel for every node is $\beta_i(i=1\cdots n)$, as follows:

$$\beta_i=\frac{P_i}{\sum_{j=1}^{n} P_j} \tag{5}$$

In this model, there are two probabilities, rejection probability of request α and disconnection probability of P2P data transmitting channel β_i. Rejection probability of request α influences packet switching process when P2P software begins to connect into overlay network. By current traffic parameters, gateway randomly drops every request packet depending on rejection probability. Using random drop method can prevent later nodes have no chance to start a P2P connection. Disconnection probability of P2P channel β_i will be used when gateway is overloaded. Gateway detects total bandwidth of P2P traffic based on network measurement module in runtime. If total bandwidth of P2P traffic beyond the upper limit of estimated range S, gateway will randomly interrupt more than one P2P data channels depending on disconnection probability. At the same time, gateway will reject all requests packet, because rejection probability is one. After that, new P2P data channel will not be constructed and old P2P data channels which using more network resource have higher probability to be interrupted. Thus, gateway will force P2P systems release bandwidth until gateway's load reduce to S.

4. IMPLEMENT AND EXPERIMENTS

According to previous sections, we constructed a model for P2P software and designed an algorithm to control P2P traffic. Then, we implemented this algorithm as a computer function in gateway. Control function is embedded into packet filter system. And packet filter software of gateway collects every node's parameters, such as packet sending rate, packet receiving rate and TCP port. Network measurement module identifies every packet by TCP port and record packet information of P2P traffic and other applications' traffic, respectively. Control function which contains control algorithm will calculate rejection probability of request and disconnection probability of P2P data transmitting channel to determine how to control P2P traffic in real-time.

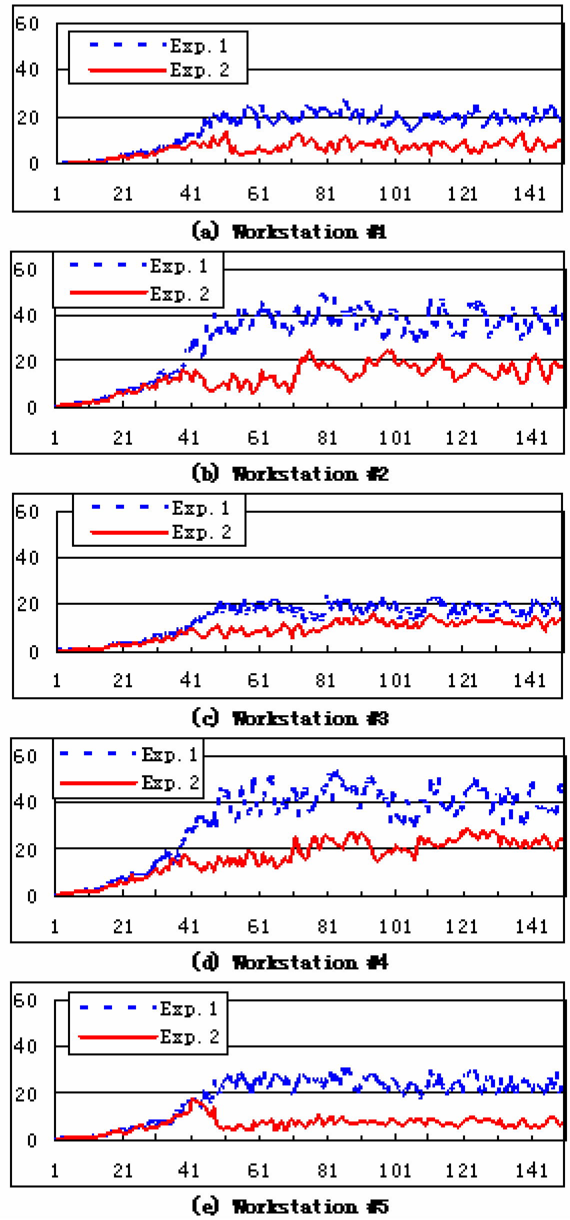

Fig 3. Contrast charts of five workstations' bandwidth. (kB/s)

Packet filter was deployed in a gateway, and the local area network had five workstations. Every workstation had some different P2P softwares, such as BitComet, Emule and BitTorrent. To accurately identify every P2P packet, we bound P2P softwares to fixed TCP port. In control algorithm, constant t was 10 and w was 4. P2P estimated bandwidth was $S \in [0kB/s, 80kB/s]$. All P2P softwares started together and shared data with other neighbor nodes in Internet. Experiments last 300 seconds and record bandwidths every 2 seconds. Two parallel experiments are run for contrast. The first experiment (Exp.1) was the common packet filter model which was not implement control algorithm. The second experiment (Exp.2) used packet filter with control algorithm. After experiments, contrast charts of bandwidth of five workstations as figure 3 and contrast chart of total bandwidth of gateway as figure 4.

Results of two experiments show that if gateway's packet filter did not use control algorithm, every P2P software in workstations would search other neighbor nodes and constructed many data channels for transmitting. Total bandwidth of P2P traffic has been increased quickly until gateway overloaded. Using control algorithm, however, bandwidth of P2P traffic have been controlled well in estimated range. Thus, gateway has enough bandwidth to serve other applications. By our algorithm, LAN can control P2P systems' traffic, and gateway may assign network resource for every application fairly.

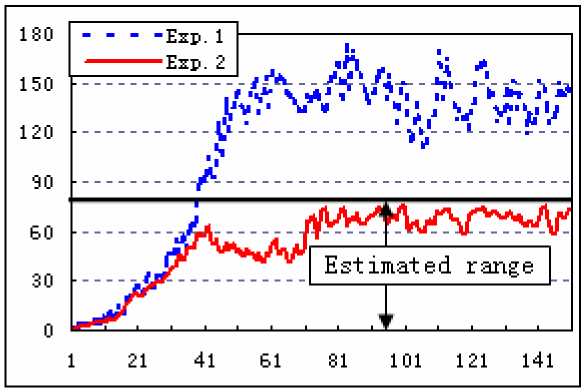

Fig 4. Contrast chart of total bandwidth. (kB/s)

5. SUMMARY

In order to reserve enough bandwidth for other applications, gateway must control P2P's traffic. In this paper, we introduced network measurement and some important network parameters in P2P system. Some different methods of measurement were classified. And we analyzed common behaviors of P2P softwares based on network measurement and found a relationship between P2P traffic and behavior. Then, we constructed a model of P2P traffic and designed an algorithm for traffic control. At last, two contrast experiments had been performed. The results of two parallel experiments show that control algorithm of P2P traffic based on network measurement can reserve enough network resource without disabling all P2P packets.

6. ACKNOWLEDGEMENT

This work was supported by the National Science Foundation of China (60573123).

REFERENCES

Andrew Oram, Andy Oram (2001), *Peer-to-Peer: Harnessing the Power of Disruptive Technologies* [M], O'Reilly & Associates, Inc., Sebastopol, CA.

Ben Byer, Evan Martin, Colten Edwards (2000), *Napster messages*, http://opennap.sourceforge.net/napster.txt.

Bram Cohen (2007), *Bittorrent protocol specification*, http://wiki.theory.org/BitTorrentSpecification

J.A. Pouwelse, P. Garbacki, D.H.J. 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.

Laurent Bernaille, Renata Teixeira (2006), *Traffic classification on the fly* [J], ACM SIGCOMM Computer Communication Review, **Vol36(2)**, pp23-26.

Oliver Heckmann, Axel Bock (2002), *The eDonkey 2000 Protocol*, ftp://ftp.kom.e-technik.tu-darmstadt.de/pub/papers/HB02-1-paper.pdf.

Pablo Rodriguez, See-Mong Tan, Christos Gkantsidis (2006), *On the feasibility of commercial, legal P2P content distribution* [J], ACM SIGCOMM Computer Communication Review, **Vol36(1)**, pp75-78.

Savage S., Collins A., Hoffman E. (1999), *The End-to-End Effects of Internet Path Selection* [C], Proceedings of ACM SIGCOMM, pp289-299.

Saroiu S, Gummadi P K, Gribble S D (2002), *A measurement study of peer-to-peer file sharing systems* [C], Proc. of the Multimedia Computing and Networking 2002, pp156–170.

Stemm M., Katz, R., Seshan S. (2000), *A network measurement architecture for adaptive applications* [C], INFOCOM 2000, **Vol(1)**, pp285-294.

Sen S, Wang J (2004), *Analyzing peer-to-peer traffic across large networks* [C], IEEE/ACM Transactions on Networking, **Vol(12)**:2, pp219-232.

Vinay Aggarwal, Anja Feldmann, Christian Scheideler (2007), *Can ISPS and P2P users cooperate for improved performance?* [J], ACM SIGCOMM Computer Communication Review, **Vol37(3)**, pp29-40.

Wijata Y.I., Niehaus D., Frost V.S. (2000), *A scalable agent-based network measurement infrastructure* [J], Communications Magazine, IEEE, **Vol(38)**:9, pp174-183.