Journal of Zhejiang University SCIENCE A
ISSN 1673-565X (Print); ISSN 1862-1775 (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

Content subscribing mechanism in P2P streaming based on gamma distribution prediction[*]

GUO Tong-qiang[†], WENG Jian-guang[†‡], ZHUANG Yue-ting
(*School of Computer Science, Zhejiang University, Hangzhou 310027, China*)
[†]E-mail: guotq@cs.zju.edu.cn; wengjg@cs.zju.edu.cn

Received Feb. 25, 2007; revision accepted Sept. 13, 2007

Abstract: P2P systems are categorized into tree-based and mesh-based systems according to their topologies. Mesh-based systems are considered more suitable for large-scale Internet applications, but require optimization on latency issue. This paper proposes a content subscribing mechanism (CSM) to eliminate unnecessary time delays during data relaying. A node can send content data to its neighbors as soon as it receives the data segment. No additional time is taken during the interactive stages prior to data segment transmission of streaming content.

CSM consists of three steps. First, every node records its historical segments latency, and adopts gamma distribution, which possesses powerful expression ability, to express latency statistics. Second, a node predicts subscribing success ratio of every neighbor by comparing the gamma distribution parameters of the node and its neighbors before selecting a neighbor node to subscribe a data segment. The above steps would not increase latency as they are executed before the data segments are ready at the neighbor nodes. Finally, the node, which was subscribed to, sends the subscribed data segment to the subscriber immediately when it has the data segment. Experiments show that CSM significantly reduces the content data transmission latency.

Key words: P2P streaming, Gamma distribution, Content subscribing mechanism (CSM)
doi:10.1631/jzus.2007.A1983 **Document code:** A **CLC number:** TP393.4

INTRODUCTION

Large-scale real-time multimedia streaming has been referred to as one of the most vital applications in current and next generation Internet. There have always been bottlenecks at the server side in traditional client-server streaming service as every client consumes the server's bandwidth. IP multicast was designed to overcome such disadvantage of client-server mode, but it has not been widely deployed. Alternatively, P2P streaming is a feasible way to provide multimedia services by utilizing the upload bandwidth of each end-user.

Scalability and latency are key challenges to be addressed in the implementation of P2P streaming systems. Tree-based systems, e.g. Narada (Chu *et al*., 2000), CAN-multicast (Ratnasamy *et al*., 2001), SCRIBE (Castro *et al*., 2002), Nice (Banerjee *et al*., 2002), Zigzag (Tran *et al*., 2003), Peercast (Bawa *et al*., 2003), ACTIVE (Liu and Zimmermann, 2006) are proposed to construct spanning trees rooted at the streaming sources and push data from the roots to all other nodes along the tree branches. Hence the latency will be as low as possible for a stable tree-based overlap. However, tree-based overlay cannot utilize all resources of the participant nodes and lacks scalable capacity for heterogeneous and asymmetric networks such as ADSL networks. Furthermore, many peers join and leave the group frequently, which will make it difficult to maintain a stable multicast tree.

[‡] Corresponding author

[*] Project supported by the National Natural Science Foundation of China (Nos. 60533090 and 60525108), the National Basic research Program (973) of China (No. 2002CB312101), the China-US Million Book Digital Library Project (www.cadal.zju.edu.cn), the Science and Technology Project of Zhejiang Province (Nos. 2005C13032 and 2005C11001-05), and the Research Project of Education Department of Zhejiang Province (No. 20061352), China

Unlike tree-based overlap, mesh-based systems, e.g. Donet (Zhang X.Y. *et al*., 2005) , PRM (Banerjee *et al*., 2003), Chainsaw (Pai *et al*., 2003), GridMedia (Zhang M. *et al*., 2005), never construct a single source multicast tree. Instead, each node takes a small set of other nodes as neighbors. A node can get streaming content from all its neighbors and find substitutions quickly once a neighbor leaves. Every node manages its neighbors set independently and employs schedule algorithms to arrange data requests. No particular structured topology is necessary to construct. Such autonomous manner enables mesh-based method to support a large number of nodes.

A major challenge of mesh-based method is latency reduction with a significantly growing number of nodes. From the view of a single node, streaming data is received from multiple neighbors. This requires a node to determine from which neighbor to ask for each data segment according to the data availability reported by the neighbors. Conventionally, there is a serial interactive stage before content data is transmitted as illustrated in Fig.1. Such interactive stages result in a hop latency of mesh-based overlap, which is much longer than that of a tree-based one.

Some optimization methods have been proposed to reduce latency within mesh-based overlap. Data redundancy and pushing data are two typical strategies. PRM (Banerjee *et al*., 2003) is a typical system, adopting data redundancy strategy. It enables nodes to send streaming content to some redundant nodes randomly. Many bandwidth resources are wasted even though latency is reduced by shortening relaying paths. GridMedia (Zhang M. *et al*., 2005) is a system which employs pushing data strategy to reduce latency. It uses a push-pull method to simulate push effects in tree-based method and to obtain failed data by pull method. GridMedia needs a global synchronization mechanism which is impractical for applications consisting of a large number of Internet users.

This paper proposes a new method to reduce the latency of mesh-based P2P streaming systems referred to as content subscribing mechanism (CSM). CSM tries to eliminate unnecessary time delays during data relaying. A node can send content data to its neighbors as soon as it receives the data segment. No redundant time is taken by interactive stages before data segment transmission of streaming content. CSM consists of three steps. First, every node records its historical segments latency, and adopts gamma distribution, which possesses powerful expression ability, to express latency statistics. Second, a node predicts subscribing success ratio of every neighbor by comparing its gamma distribution parameters with its neighbors' before selecting a neighbor node to subscribe a data segment. The above steps would not increase latency as they are executed before the data segments are ready at neighbor nodes. Finally, the node, which was subscribed to, sends the subscribed data segment to the subscriber immediately when it has the data segment.

The remaining sections of this paper are organized as follows. Section 2 analyses the latency issue in traditional tree-based P2P streaming approaches. Section 3 describes the gamma distribution of segment latency in mesh-based P2P streaming systems. Section 4 describes the content subscribing mechanism and Section 5 evaluates the overall latency with content subscribing mechanism. Experimental comparison is given in Section 6. Finally Section 7 concludes the paper.

LATENCY ISSUE

The data driven method is a typical mesh-based P2P streaming approach. We will first analyze its latency issue. In data driven method, data segments are relayed among the peers hop by hop. The typical flow from the time a content segment is ready at the sender to the time it is ready at the receiver is depicted in Fig.1. The milestones are listed in Table 1.

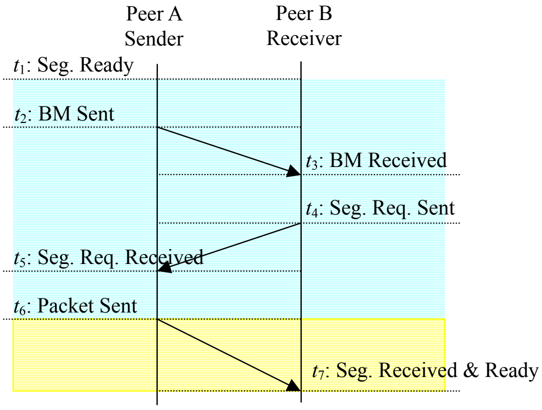

Fig.1 Time flow of segment relaying process in data driven method

Table 1 Milestones of segment relaying process in data driven method

Parameter	Description
t_1	A Segment is ready at Peer A
t_2	Peer A sends Buffer Map (BM) to Peer B
t_3	Peer B receives BM from Peer A
t_4	Peer B sends Segment Request to Peer A
t_5	Peer A receives Segment Request from Peer B
t_6	Peer A sends Segment to Peer B
t_7	Peer B receives Segment from Peer B, i.e., Segment is ready at Peer B

Assuming the time delay in transferring a single packet between the sender and the receiver is τ, the interval of sending buffer map is ε, and the interval of scheduling incoming requests and outgoing acknowledgement is σ. Then the expected hop latency δ_{hop} is as follows:

$$\delta_{\text{hop}} = \frac{1}{2}\varepsilon + \tau + \frac{1}{2}\sigma + \tau + \frac{1}{2}\sigma + \tau = \frac{1}{2}\varepsilon + 3\tau + \sigma. \quad (1)$$

The hop latency is clearly much longer than the time needed to transfer a single packet. We combine the first five parts as preparative latency δ_{prep} and denote the last part used in transferring the segment as transferring latency δ_{tran}. From both theoretical and practical views, δ_{prep} is the major cause of the high latency. The goal of this paper is to eliminate this latency using a data subscribing mechanism.

GAMMA DISTRIBUTION OF SEGMENT LATENCY

We denote the delay of a segment as the period of time from producing it by the source until the time it is received by a peer, as segment latency (δ). For many reasons, δ is not constant for a peer, however it lies in a time region. The long term statistics for a large group of peers (e.g. Fig.2 and Fig.3) shows that the segment latency corresponds to gamma distribution, which is often used in reliability theory, waiting time and queue problems. In related analysis (Verma and Ooi, 2005), gamma distribution is also adapted to model hop delay in unstructured P2P circumstance.

The density function of the distribution function of segment latency is

$$p(t) = \begin{cases} 0, & t < \gamma, \\ \dfrac{\beta^\alpha}{\Gamma(\alpha)}(t-\gamma)^{\alpha-1}\mathrm{e}^{-\beta(t-\gamma)}, & t \ge \gamma, \end{cases} \quad (2)$$

where $\Gamma(\alpha) = \int_0^\infty t^{\alpha-1}\mathrm{e}^{-t}\mathrm{d}t$, γ is an offset parameter.

The distribution function of segment latency is

$$F(t) = \begin{cases} 0, & t < \gamma, \\ \displaystyle\int_{t_0}^{t} \frac{\beta^\alpha}{\Gamma(\alpha)}(t-\gamma)^{\alpha-1}\mathrm{e}^{-\beta(t-\gamma)}\mathrm{d}t, & t \ge \gamma, \end{cases} \quad (3)$$

In the above density and distribution functions, the three parameters α, β and γ can be estimated according to the historical statistics as described below.

For a series of segment latency $[\delta_1, \delta_2, \ldots, \delta_n]$, n is the number of the latency samples. The latency with maximum density is δ_{m}, which satisfies $\forall t$, $p(\delta_i \in (\delta_{\text{m}}-\varepsilon, \delta_{\text{m}}+\varepsilon]) \ge p(\delta_i \in (t-\varepsilon, t+\varepsilon])$, where precision threshold is ε. The mean of segment latency is $\delta_{\text{e}} = \sum_{i=1}^{n} \delta_i / n$, and the variance is $d^2 = \sum_{i=1}^{n} (\delta_i - \delta_{\text{e}})^2 / n$. γ is estimated by calculating the derivative of Eq.(2)

$$\begin{aligned} \frac{\mathrm{d}p(t)}{\mathrm{d}t} = \frac{\beta^\alpha}{\Gamma(\alpha)}[(\alpha-1)(t-t_0)^{\alpha-2}\mathrm{e}^{-\beta(t-\gamma)} - \\ (t-\gamma)^{\alpha-1}\mathrm{e}^{-\beta(t-\gamma)}\beta] = 0, \end{aligned} \quad (4)$$

and $\delta_{\text{m}}=\gamma+(\alpha-1)/\beta$. With the mean and variance of gamma distribution, $\mu=\gamma+\alpha/\beta$, $\sigma^2=\alpha/\beta^2$, we obtain the following equations set:

$$\begin{cases} \gamma = \delta_{\text{m}} - \dfrac{\alpha-1}{\beta}, \\ \gamma + \dfrac{\alpha}{\beta} = \displaystyle\sum_{i=1}^{n} l_i / n, \\ \dfrac{\alpha}{\beta^2} = \displaystyle\sum_{i=1}^{n} (\delta_i - \delta_{\text{e}})^2 / n. \end{cases} \quad (5)$$

By resolving Eq.(5), we get α, β and γ, which are adequate to represent the latency characteristic of a peer and are important guidelines for content subscribing mechanism.

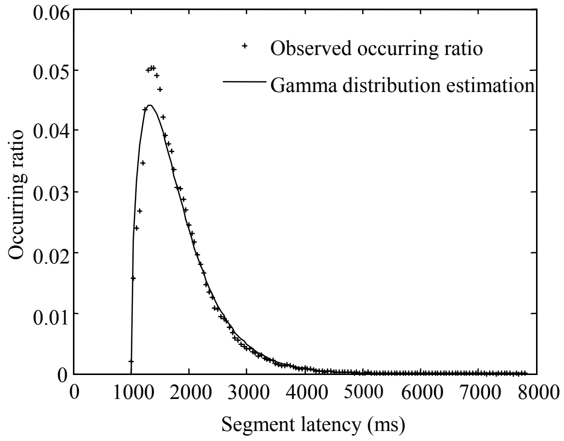

Fig.2 Occurring ratio vs. segment latency. The observed data is logged with a 3620-node group; the gamma parameters are α=1.6746, β=0.1024, γ=1000 ms

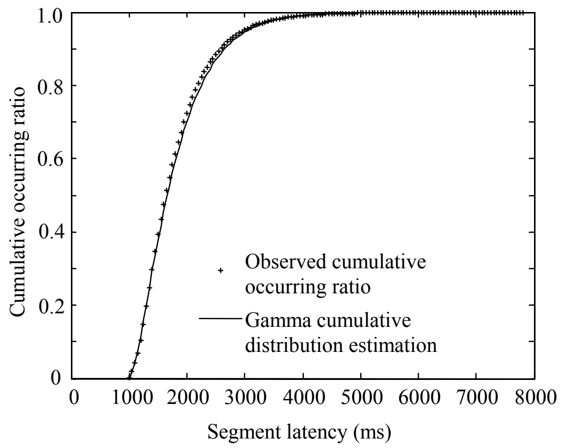

Fig.3 Accumulating occurring percentages vs. segment latency

CONTENT SUBSCRIBING MECHANISM

Content subscribing mechanism is based on the gamma distribution. A peer can predict the available time of each segment for its neighbors. The peer selects capable neighbors to send data subscribing request to, depending on the probability and other factors, such as the upload capacity and inter-peer priority. The subscribed data can be instantly transferred to the peer once it is available. Such mechanism dramatically reduces the latency close to that in tree-based method because it eliminates the preparative latency δ_{prep}. The revised time flow of relaying a segment is demonstrated in Fig.4. The corresponding milestones are listed in Table 2. It should be emphasized that the optimized hop latency consists of nothing more than the gray part, i.e. the transfer latency.

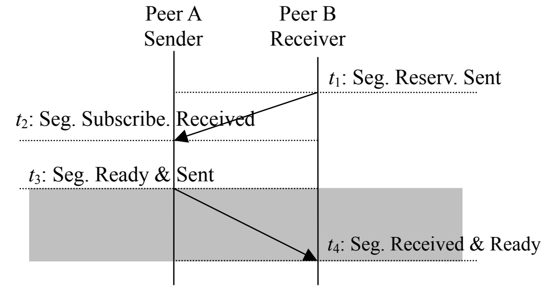

Fig.4 Time flow of segment relaying process in data driven method with content subscribing mechanism

Table 2 Milestones of segment relaying process in data driven method with content subscribing mechanism

Parameter	Description
t_1	Peer B sends Segment Subscribing to Peer A
t_2	Peer A receives Segment Subscribing from Peer B
t_3	The Segment is ready at Peer A and is sent to Peer B immediately
t_4	Peer B receives Segment from Peer B, i.e., Segment is ready at Peer B

The major step of CSM is selecting capable neighbors to send the subscribing requests to. The latency distribution function is adapted to compare the possibility of the success of data subscribing among different neighbors. Suppose the neighbors of a peer X are Nh_i, i=1, …, n. The latency parameters for peer X are α_X, β_X, γ_X. The latency parameters for neighbor Nh_i are α_i, β_i, γ_i. The transferring latency for neighbor Nh_i is $\delta_{\text{tr},i}$. Assuming peer X is requesting a data segment from a neighbor Nh_i with the expected latency of no more than $t_X+\alpha_X/\beta_X$, the probability is

$$F(Nh_i)=\begin{cases} 0, & \dfrac{\alpha_X}{\beta_X}+t_X-\Delta_{\text{tr},i}<t_i, \\ \displaystyle\int_{t_i}^{\frac{\alpha_X}{\beta_X}+t_X-\Delta_{\text{tr},i}} \frac{\beta_i^{\alpha_i}}{\Gamma(\alpha_i)}(t-t_i)^{\alpha_i-1}\mathrm{e}^{-\beta_i(t-t_i)}\mathrm{d}t, & \\ & \dfrac{\alpha_X}{\beta_X}+t_X-\Delta_{\text{tr},i}\geq t_i. \end{cases} \quad (6)$$

Given the minimum tolerable probability $p_{\min}$, the candidate neighbor set S_{C} is

$$S_{\mathrm{C}} = \{Nh_i \mid p(Nh_i) > p_{\min} \wedge \frac{\alpha_X}{\beta_X} + \gamma_X \ge \frac{\alpha_i}{\beta_i} + \gamma_i + \delta_{\mathrm{tr},i}\}$$
$$\text{for } i = 1, \ldots, n \tag{7}$$

where the condition $\gamma_X + \alpha_X/\beta_X \ge \alpha_i/\beta_i + \gamma_i + \delta_{\mathrm{tr},i}$ helps the peer to avoid the deadlock of subscribing request among peers as well as to optimize the overlay by subscribing data from neighbors with lower latency.

Considering the accepted ratio ξ_i of data subscribing requests for neighbor Nh_i, the best neighbor Nh_{best} is chosen from non-empty candidate neighbor set S_{C}. Nh_{best} satisfies

$$Nh_i \in S_{\mathrm{C}} \wedge Nh_{\mathrm{best}} \ne Nh_i \wedge F(Nh_{\mathrm{best}}) \cdot \xi_{\mathrm{best}} > F(Nh_i) \cdot \xi_i \quad \forall Nh_i. \tag{8}$$

In most cases, the best neighbor to subscribe data segment exists. Otherwise the peer waits for the next schedule loop. Finally a traditional data driven method is used.

OVERALL LATENCY ESTIMATION

Many factors lead to failed subscribing requests and increase the segment latency. Let η denote the ratio of successfully subscribing requests among all requests. If a subscription fails, the data driven method is adopted as default. Let δ_{w} denote the extra waiting time for failed subscription compared with pure data driven method. The overall hop latency is

$$\begin{aligned}\delta_{\mathrm{hop}}(\eta) &= \delta_{\mathrm{tr}} \cdot \eta + (\delta_{\mathrm{prep}} + \delta_{\mathrm{trans}} + \delta_{\mathrm{w}}) \cdot (1-\eta) \\ &= \delta_{\mathrm{trans}} + (\delta_{\mathrm{trans}} + \delta_{\mathrm{w}}) \cdot (1-\eta).\end{aligned} \tag{9}$$

We estimate δ_{w} by analyzing the density function of the latency distribution. Given a maximum probability threshold $p_{\max}$, we get $t_{\max}$ by resolving the following equation

$$\int_{\gamma_i}^{t_{\max}-\delta_{\mathrm{tr},i}} \frac{\beta_i^{\alpha_i}}{\Gamma(\alpha_i)} (t - t_i)^{\alpha_i - 1} \mathrm{e}^{-\beta_i (t-\gamma_i)} \mathrm{d}t = p_{\max}. \tag{10}$$

The root $t_{\max}$ indicates that it is wasteful for the peer to continue waiting for the subscribed data due to the low density of the latency distribution. Suppose the timestamp of requested segment is t_{SG} and the time of receipt of response denial from the neighbor is t_{deny}, then

$$\delta_{\mathrm{w}} = \begin{cases} 0, & \min(t_{\mathrm{deny}}, t_{\mathrm{SG}} + t_{\max}) < t_4, \\ \min(t_{\mathrm{deny}}, t_{\mathrm{SG}} + t_{\max}) - t_4, & \min(t_{\mathrm{deny}}, t_{\mathrm{SG}} + t_{\max}) \ge t_4, \end{cases} \tag{11}$$

where t_4 is the time when the receiver has sent segment request to the sender, as shown in Fig.1.

Eq.(11) indicates that if the neighbor sends the response denial in time, the extra waiting time is expected to be zero, i.e.,

$$\begin{aligned}\delta_{\mathrm{hop}}(\eta)_{\mathrm{opt}} &= \delta_{\mathrm{tr}} \cdot \eta + (\delta_{\mathrm{pr}} + \delta_{\mathrm{tr}} + 0) \cdot (1-\eta) \\ &= \delta_{\mathrm{tr}} + \delta_{\mathrm{pr}} \cdot (1-\eta).\end{aligned} \tag{12}$$

Eq.(12) explains the principle of selecting capable neighbors in content subscribing mechanism described in the previous section.

Now we analyze the overall segment latency. There is a corresponding relaying tree for each segment which may be different for every data segment. Assuming the group size is N and the average degree for the inner node is d. Layer 0 contains the source peer and Layers L and L+1 contain all leaf nodes. We can obtain

$$\frac{L^d - 1}{L-1} \le N < \frac{L^{d+1} - 1}{L-1}. \tag{13}$$

The average segment latency is

$$\begin{aligned}\delta_{\mathrm{node}}(\mathrm{avg.}) &= \frac{1}{N}\left[\sum_{k=1}^{L} d^k (k-1)\delta_{\mathrm{hop}} + \left(N - \frac{L^d - 1}{L-1}\right)\delta_{\mathrm{hop}}\right] \\ &\approx \lambda \cdot (\log_d N) \cdot \delta_{\mathrm{hop}},\end{aligned} \tag{14}$$

where λ is a coefficient near 1.

We can conclude that the average segment latency is linear to the hop latency. This illustrates that content subscribing mechanism can reduce the hop latency significantly by eliminating the preparative latency. The latency is optimized to be similar as that of tree-based method. Furthermore no particular structure needs to be maintained as tree-based method.

EVALUATIONS

In this section, we first use Ns simulations to examine various design issues and tradeoffs in mesh-based P2P streaming with the content subscribing mechanism. We focus on the relationship between segment latency, group size, node degree, startup time, churns and success ratio of data subscribing requests. A packet level simulator is more suitable than session level simulators.

We have also included the proposal in this paper and deployed it on the Internet (www.tvants.com). Some experimental results are also obtained by measurement of actual Internet applications.

Simulation setup

In our simulations, the physical topology is generated with Brite, using the following configuration parameters: 16 AS with 16 routers per AS in top-down mode and RED queue management at all routers. The delay on each access link is randomly selected between 5 ms and 100 ms. Core links have high bandwidth and thus all connections experience bottleneck only on the access links. To form a randomly connected and directed overlay, each peer contacts a bootstrapping node to learn about a random subset of participating peers until it identifies the specified number of neighbors. The segment scheduling mechanism and content subscribing mechanism in individual peers are implemented. Each simulation runs for 600 s. The presented results illustrate the behavior of the system during the steady state after all peers have identified their neighbors and can obtain qualified streaming data from neighbors.

Average segment latency according to number of peers

To explore the average segment latency according to the number of peers, we examine scenarios with different numbers of peers with homogeneous and symmetric access links. Every peer reports its average segment latency to a log server. Fig.5 depicts the average segment latency and average hop latency according to the number of peers. The latency is evidently shortened when content subscribing mechanism is adapted, which is also confirmed by Internet application measurements.

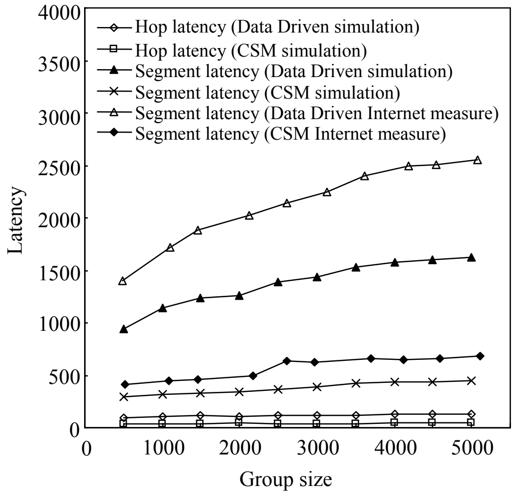

Fig.5 Average hop latency and segment latency vs. the number of peers

Average hop latency according to subscribing success ratio

The effect of content subscribing mechanism is influenced by the success ratio of subscribing request. To examine this issue, we simulate broadcasting groups with 1000 and 5000 peers with different success ratios. Fig.6 depicts the average hop latency according to the subscribing success ratio. We obtain the shortest hop latency when the success ratio is 100%. However, when the success ratio is below a threshold (30% with 1000 peers; 36% with 5000 peers), the hop latency is longer than that with a traditional data driven method.

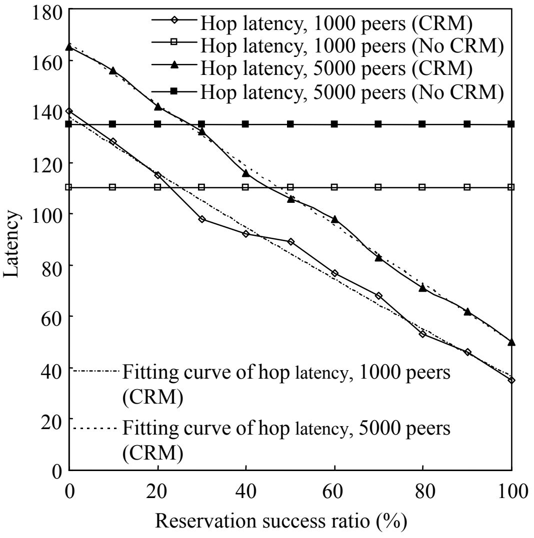

Fig.6 Average hop latency vs. the subscribing success ratio

CONCLUSION

In this paper, we adapt gamma distribution to model the segment latency in mesh-based P2P streaming. The impressive representation of gamma distribution helps us to propose a new content subscribing mechanism to address latency issue. Simulation and Internet experiments show that our method can achieve both high scalability and low latency.

In next step, we plan to improve our implementation and conduct more experiments to verify and expand our analytical theory.

References

Banerjee, S., Bhattacharjee, B., Kommareddy, C., 2002. Scalable Application Layer Multicast. Proc. Conf. on Applications, Technologies, Architectures, and Protocols for Computer Communications. ACM Press, New York, p.205-217.

Banerjee, S., Lee, S., Bhattacharjee, B., Srinivasan, A., 2003. Resilient Multicast Using Overlays. Proc. ACM SIGMETRICS Int. Conf. on Measurement and Modeling of Computer Systems. ACM Press, New York, p.102-113. [doi:10.1145/781027.781041]

Bawa, M., Deshpande, H., Garcia-Molina, H., 2003. Transience of peers & streaming media. *SIGCOMM Comput. Commun. Rev.*, **33**(1):107-112. [doi:10.1145/774763.774780]

Castro, M., Druschel, P., Kermarrec, A., Rowstron, A., 2002. SCRIBE: a large-scale and decentralized application-level multicast infrastructure. *IEEE J. Selected Areas Commun.*, **20**(8):1489-1499. [doi:10.1109/JSAC.2002.803069]

Chu, Y.H., Rao, S.G., Zhang, H., 2000. A case for end system multicast. *Meas. Model. Computer Syst.*, **6**:1-12.

Liu, L.S., Zimmermann, R., 2006. Adaptive low-latency peer-to-peer streaming and its application. *Multimedia Systems Journal*, **11**(6):497-512. [doi:10.1007/s00530-006-0030-4]

Pai, V.S., Kumar, K., Tamilmani, K., Sambamurthy, V., Mohr, A.E., 2003. Chainsaw: Eliminating Trees from Overlay Multicast. IPTPS, p.127-140.

Ratnasamy, S., Handley, M., Karp, R., Shenker, S., 2001. Application Level Multicast Using Content-addressable Networks. Proc. 3rd Int. Workshop on Networked Group Communication. London, UK, p.14-29.

Tran, D., Hua, K., Do, T., 2003. Zigzag: An Efficient Peer to Peer Scheme for Media Streaming. IEEE INFOCOM, p.1283-1293.

Verma, S., Ooi, W.T., 2005. Controlling Gossip Protocol Infection Pattern Using Adaptive Fanout. Proc. IEEE ICDCS, p.665-674.

Zhang, X.Y., Liu, J.C., Li, B., Yum, T.P., 2005. Coolstreaming/Donet: A Data-driven Overlay Network for Peer-to-Peer Live Media Streaming. Proc. IEEE INFOCOM, p.2102-2111.

Zhang, M., Zhao, L., Tang, Y., Luo, J.G., Yang, S.Q., 2005. Large-scale Live Media Streaming Over Peer-to-Peer Networks through Global Internet. Proc. ACM Workshop on Advances in Peer-to-Peer Multimedia Streaming. ACM Press, New York, p.21-28. [doi:10.1145/1099384.1099388]