A Practical Network Design for VoD Services

Hoon Lee* ***Lifelong Member***

ABSTRACT

Recently IPTV service is penetrating to the ordinary home users very swiftly. One of the first phase of IPTV service is considered to be VoD, and a nationwide availability of the VoD service imposes tremendous pressure to the network resource due to its requirements for the broad bandwidth, the inherent nature of unicast technology, and the large scalability, etc. This work suggests a novel and practical method to the design of network resource for the VoD service. Especially, we explore the distributed content storage problem that takes into account the popularity of the video contents and its corresponding link dimensioning problem that takes into account the grade of service for the flow level quality of service about the VoD service. By assuming a realistic topology for the nationwide IP backbone network of Korea, which is a typical tree topology, we suggest an analytic method for the design of VoD service.

Keywords : VoD Service, Distributed Servers, Grade of Service, Network Design, Link Dimensioning

Ⅰ. Introduction

With the evolution of IP network toward a BcN (Broadband convergence Network) with enhanced QoS (Quality of Service) functions, multimedia services are at our hands, and voice and video as well as the conventional high-speed Internet services are provided by ISPs (Internet Service Providers) as a package. This packaging is called TPS (Triple Play Service). TPS provides users with rich video contents that include IPTV (Internet Protocol Television) as well as VoD (Video On Demand).

At present it is not clear that a generic IPTV service which includes real-time transfer of live TV programs can be penetrated fully into individual users because of lots of hurdles that present in front of the ISPs and contents providers as well as the incumbent cable TV providers[1, 2].

In such an environment it seems rather easy to spread the VoD services as a first phase of IPTV service. This is because VoD contents are exclusive from the real-time TV service as seen from the business model, and they can be distributed off-line by ISPs even though they are produced and distributed by contents generators.

VoD contents are usually gathered at the video server of a SHO (Super Hub Office), and they are transferred to individual customers as real-time streams when requests on video contents are made by users. It is usual that VoD is unicast traffic because users want different video contents based on their interests.

Note that it is uneconomic to store all the VoD contents at a single location and deliver them to users who are dispersed in a nation-wide geography, because a large portion of users are located very far from the SHO. This incurs long delay to the users and it also causes large amount of bandwidth in the backbone network for the video transfer.

To resolve this problem, it is suggested that a distributed storage scheme is proposed such that popular VoD contents are stored at the VHOs (Video Hub Offices), whereas unpopular contents are stored only in SHO[3]. When new video contents are arrived to SHO and it becomes popular, one can deliver those contents to VHOs at off-peak

* Dept. of Information & Communication Engineering, Changwon National University (hoony@cwnu.ac.kr)
Paper number : KICS2008-10-430, Manuscript Received : 2008년 10월 2일, Final manuscript received : 2009년 2월 26일

time, so that this does not incur traffic congestion in the backbone network.

In designing a distributed network architecture for the VoD service lots of problems have to be resolved such as determination of the number of servers, server location, routing between the server and clients, link dimensioning, guarantee of QoS, etc. There exists lots of work that deals with these problems in the web service. For example, Mao et al. presented a method to determine the location and number of cache servers for the streaming services over the virtual private network[4].

We can find lots of work that promotes the adoption of distributed server placement for the VoD service[3]. If such a service environment is successfully implemented in the real-field, lots of problems have to be resolved such as the modeling of the user behavior (popularity of the video contents), determination of the number of video contents to each server, optimal capacity of the link that takes into account the GoS (Grade of Service), etc. Molina et al. proposed a general model for the CDN (Content Delivery Network) which takes into account the distributed server location. They assumed two-level server model with one origin server and multiple surrogate servers, and they investigated the delay performance of the model[5].

Even though we found lots of work that promote the distributed architecture for the video servers, we could not find any result that models the realistic network and investigates the implication of the model to the design of network resources.

In addition, to the best of author's knowledge, we could find no result that deal with a practical approach about the link dimensioning problem in the distribution network for the VoD service which takes into account the user's behavior about the popularity of the video contents as well as the introduction of GoS to the video flow. This motivated our work.

This work proposes an analytic method to design a VoD service network by taking into account those problems in a systematic approach. Especially, we propose a method to the provisioning of the bandwidth for the nationwide VoD service with the distributed server placement.

This work is composed as follows: In Section II we present an architecture for the VoD service network and discuss requirements for the economic provision of the VoD service. In Section III characteristics of the VoD service are given. In Section IV network topology and the problem of server location is defined. In Section V a mathematical model for the link dimensioning is suggested. In Section VI results for the numerical experiment is given. Finally, in Section VII, we summarize this work.

Ⅱ. Network Architecture for VoD Service

Network architecture for the VoD service varies depending on countries or ISPs. In general one can imagine that there are three distinct subjects for the VoD service: Contents providers (servers), network providers (usually IP network), and users. If we depict those elements in an abstract manner we obtain a model that is given in Fig.1.

Users are connected to a broadband IP network via an access network such as a DSL (Digital subscriber Line), and video servers are located on the other side of the network via VDS (Video distribution system), where VDS corresponds to SHO and it acts as a gateway between video server and network.

The simplest way upon which VoD servers are located in the network is a single-node storage system that is located in a centralized manner, which means that there is only one SHO in the network.

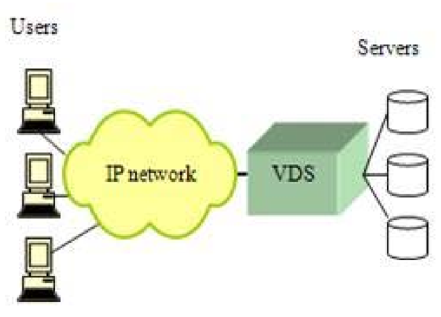


Fig.1. Network architecture for VoD services

However, there is a drawback for this architecture when a nationwide service is assumed for the VoD service, because even the largest centralized server cannot scale to large number (e.g., millions) of subscribers.

Recently, total separation of storage from streaming is a trend for large-scale VoD services, where the most popular content is cached at the edge of the network, whereas less popular content is transferred from storage, via which scaling problem can be resolved. This results in a distributed server location. In case video servers are located in distributed manner, there exist problems that have to be resolved: physical server location problem and link dimensioning problem. Our main focus lies in the link dimensioning problem. To do that let us first investigate the characteristics of the VoD service.

Ⅲ. Service Model

Note that bandwidth guarantee for the VoD service is easily implemented at the edge router by allocating appropriate bandwidth for each connection. The streaming nature of VoD requires sustained guarantee of a fixed amount of bandwidth throughout the connection holding time of a video program. Therefore, we envision that a flow level GoS (Grade of service) such as a CBP (Connection Blocking Probability) has to be defined for a VoD flow. To model such behavior of a VoD service, let us assume a few conditions. First, let us assume that connections for the request of video contents are generated by a Poisson process with mean rate λ. This is rational because the scale of nationwide VoD service is sufficiently large. Second, let us assume that duration of video programs varies depending on the type of video, and we assume that it is generally distributed with mean duration τ, and durations for the different flows are independently distributed. Then, the system load is given by $\rho = \lambda\tau$.

Finally, let us assume that link capacity of a VDS that accommodates multiple VoD programs is finite, and it can accommodate at most c connections simultaneously.

Taking into account theses facts, we argue that we can use an *M/GI/c/c* queue model for calculating the CBP. The probability $p_c(n)$, where c is the number of channels and n is the number of connections in the system at steady-state, is given by[6]

$$p_c(n) = p_0 \frac{\rho^n}{n!}, 0 \le n \le c, \tag{1}$$

where the initial condition is given by

$$p_0 = \frac{1}{\sum_{n=0}^{c} \frac{\rho^n}{n!}}.$$

Note that this is a truncated Poisson distribution. The probability $p_c(c)$ that the system is fully occupied and a new connection is blocked, that is the connection blocking probability, is given as follows:

$$p_c(c) = \frac{\frac{\rho^c}{c!}}{\sum_{n=0}^{c} \frac{\rho^n}{n!}}. \tag{2}$$

Note that the above function is also called as Erlang-B formula.

Ⅳ. Server Location

It is usual that network topology for the backbone of Korean nationwide IP network is a hybrid of mesh and star topology. To be specific, the territory of South Korea is divided into four areas such as Capital area, Chungcheong area, Youngnam area, and Honam area.

In each area, there exists a core node, which is looked upon as a core of the core, and those four nodes compose a mesh network, and the other nodes in each area are edge of the core, and they compose a star network. Those four areas are physically interconnected by tens of giga bps optical links, and logical paths exist between them in a mesh topology.

Reflecting the realistic topology of Korean IP backbone network, let us assume that the backbone

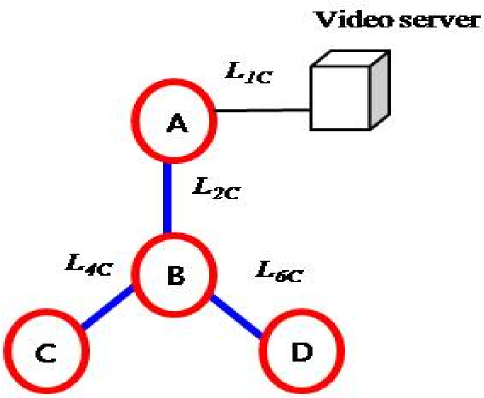


Fig. 2. Centralized server location

network for the distribution of VoD service is assumed to be a four-node topology, which is given in Fig.2.

Let us assume that the number of customers that are subscribed to the VoD service at node *X*, *X=A*, *B*, *C*, and *D*, is N_X. As to the placement of video server we have two alternatives: CSM (centralized Server Model) and DSM (Distributed Server Model)[7].

When it comes to the centralized server placement for the tree topology, the VoD server is usually located at the root, which is node A that is located at the capital area of South Korea.

By the way, there are three inherent problems in distributing a large number of video streams from a single server via unicast technology. First a link between the centralized server and network might become overloaded. For example, when a video server has 1,000 HDTV-level movies and each video stream requires 8Mbps of bandwidth, and if there are 1,000 customers who want to watch those movies (note that there is no synchronization between different customers) at the same time with random time difference, then the link capacity has to be at least 8 tera bps (1 tera=10^{12}).

To overcome the scalability problem for a single server one can construct a server cluster by using a load-balancing technique between multiple servers[8]. However, this does not solve the problem of link congestion in a node that accommodates a server cluster. Second, congestion will spread to the other part of the network such as backbone network as well as access network. Thus the long-haul network is congested. Third, users may perceive large delay and jitter for the download of a video content when it is delivered from a server which is located far away.

To resolve this problem, a distributed placement of video servers is used, via which the amount of bandwidth of ISP's distribution network is decreased at the price of increased cost for additional storage. One major trend toward this approach is CDN (Content Delivery Network). CDN improves client access to contents by providing low response time through the delivery of contents close to end users. By distributing the servers throughout the network scalability both in bandwidth and server capacity can be guaranteed. Therefore, main features of CDN are two-fold: First, contents are stored in decentralized manner by moving content closer to clients. Second, by delivering content as locally as possible, bandwidth of the backbone network can be preserved[5].

When it comes to a distributed placement of video servers two different schemes can be adopted: one is a complete disjoint storage of different video contents to each server and the other is a duplicated storage of video contents by adopting duplicated location of video contents throughout the multiple servers in hierarchical manner.

When it comes to complete disjoint storage of different video contents to each server, there may exist practical problems. First, there arises a complexity problem in searching the contents and routing the traffic in the network. For example, consider that there exists 100 video contents, and they are divided into four disjoint groups and each group is stored at each node in the network. When a customer is located at node A and a video content he wants to view is located only at node C. Then what happens? Does he have to seek every server until he finds the content he wants? The simplest way to this situation is to locate a broker in the network, and the broker mediates this transaction. However, this does not always guarantee optimality to the customers, especially if the number of the customers is large and they are dispersed throughout the nation. Second, at present, it is usual that almost the entire video contents for the commercial VoD services are generated by contents providers that are located in the vicinity of the capital area in Korea. Therefore, it is not economical to store those contents in a disjoint manner. From these two reasons, we do not assume this approach.

Note, however, that this approach is known to be useful when we assume a P2P-based solution for the distribution of large-scale VoD[9]. We admit that this is also an interesting issue.

Next, the duplicated location scheme is also widely adopted in the real-field as well as in the research work[3, 5, 10, 11] . In this work we assume this architecture, because a commercial VoD service in the incumbent business model for the IPTV services adopts duplicated storage of video contents.

When it comes to duplicated storage of video contents hierarchical location of video contents is needed such that some server has a complete list of video contents, and some server has a moderate number of popular video contents, whereas some server has a few most popular video contents. In this case, customers' requests for video contents are first processed at the local server. If it does not find a match for video contents at the local server the request is transferred to the nearest higher level server at which more video contents are stored. This is repeated until the request reaches to the main server.

Fig.3 illustrates the placement of servers for the distributed model. There are three types of servers in the network: main server, sub server, and cache server. As to the storage of video contents we propose the following policy: Main server has 100% of video contents. Sub server stores upper ψ% popular video program (e.g., ψ=50). Finally, cache server stores upper ζ % popular video program (e.g., ζ=10).

Usually it is known that link dimensioning uses reference traffic at a busy hour[12]. Now that the number of actual viewers for VoD contents is only a portion of entire subscribers, let us assume that only $\gamma \times 100\%$ of subscribers are viewing video contents in a busy hour.

We have the following policy: main server never dies. This can be realized by duplication of main

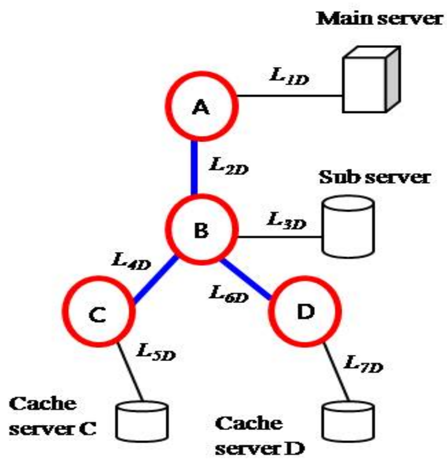


Fig. 3. Distributed server location

server. Second, we have the following counterattack for server failure: if sub server dies, requests are transferred to the main server where the entire video contents are stored. If cache server dies, requests are transferred to sub server.

Finally, as to the characteristics of the users' popularity for video contents, it is usually known that users' popularity for the video contents follows Zipf's law[11]. That is, users' popularity for video contents is focused heavily on the upper few percents. Recently it is known that top 10% of video programs occupy 48% of rentals[13]. From these facts we argue that one can model the probability distribution function *F(n)* for video requests by using a power law function, which is given as follows:

$$F(n) = 1 - \left(\frac{1}{n}\right)^{\alpha} \quad (3)$$

In (3) n is the number of the most populated video requests that is counted from the top rank and α is an impact factor.

Note that the probability density function $p(n)$, which is the probability that a user requests for the *n-th* most popular video, is given as follows:

$$p(n) = \alpha n^{-\alpha - 1}. \quad (4)$$

Note also that the probability of rental for the most popular video is α.

If we assume that the total number of video contents is V, then we obtain a formula *R(n)* for the number of requests for the *n-th* popular video content, which is given by

$$R(n) = p(n) \times \gamma \times N_X, n = 1, 2, ..., V.$$

Ⅴ. Link Dimensioning

Now let us present a method to determine the link capacity. Our concern lies in the determination of the link capacity between two neighboring nodes as well as the link capacity between a node and a server in the network. The most important parameter as an input to the link dimensioning is the offered load for the target link, which is denoted by ρ_X, where *X* is the index for the link or node. When it comes to the link index *X=1C*, *2C*, *4C*, and *6C* for the CSM and *X=1D* to *7D* for the DSM. When it comes to the node index *X=A* to *D*, which is common to the CSM and DSM.

Following our previous discussion about the queuing system model the offered load (in the unit of Erlang) to the system is given as follows:

$$\rho_X = \frac{\lambda_Y \times \tau}{3600} [Erl] \tag{5}$$

In (5) one may note that the index for ρ and λ are different. That is, the indexes for ρ and λ are denoted as *X* and *Y*, respectively. This is because they can be the same or different, which depends on the situation. This is indicated in detail in the subsection below.

Note that λ_Y is the mean number of requests for video content that are generated by customers that are directly connected to node *Y* during a busy hour, and τ is the mean holding time of the video connection. Note that we can compute the parameters λ_X and τ if we have information about the number of customers and the attributes of the users. λ_Y is computed in the following subsection by assuming specific parameters for each case.

Now let us define the link capacity as follows: Link capacity is the amount of bandwidth that is required to guarantee a predefined GoS, for example ϕ%, for the given offered load in the link.

Because bandwidth is allocated to each video stream, we can abstract the link capacity as a unit of channel with a predefined amount of bandwidth.

Let us assume that each video stream is homogeneous, and the number of channels in the link is c. If we let the required bandwidth of a video channel is β, then we obtain the following formula for the link capacity Γ :

$$\Gamma = c \times \beta \tag{6}$$

The number of channel c is computed from (2), whereas β is assumed to be fixed such that an STV (Standard TV) program requires 3 Mbps and an HDTV (High Definition TV) program requires 8 Mbps.

If c is large, it is not easy to compute $p_c(c)$ in eq.(2) by using a normal PC, instead we have to use the following recursive formula[6] :

$$p_c(c) = \frac{\rho \times p_c(c-1)}{c + \rho \times p_c(c-1)} \tag{7}$$

where $p_c(0) = 1$.

In our discussion about the link dimensioning we consider two scenarios: centralized server and decentralized server.

5.1 Centralized Server Model

Network architecture for CSM is given in Fig.2. For CSM, the entire video programs are located at the main server, which implies that we do not have to concern about which video file is rented by customers.

First, let us compute the offered load of link L_{4C}, which is denoted by ρ_{4C}, and it is given by $\rho_{4C} = \frac{\lambda_C \times \tau}{3600}$, where $\lambda_C = N_C \times \gamma$. Note that $\rho_{4C} = \rho_C$.

In the same way, we can compute the offered load of link L_{6C}, which is denoted by ρ_{6C}, and it is given by $\rho_{6C} = \frac{\lambda_D \times \tau}{3600}$, where $\lambda_D = N_D \times \gamma$. Note that $\rho_{6C} = \rho_D$.

Now let us compute the offered load to link L_{2C}, which is denoted by ρ_{2C}. Note that the link L_{2C} is used for the customers in node *B*, *C* and *D* that bound for main server, and the offered load to link L_{2C} from node *B*, *C* and *D* is denoted as $\rho_{B,}\rho_{4C}$ and ρ_{6C}, respectively, where $\rho_B=\frac{\lambda_B\times\tau}{3600}$ and $\lambda_B=N_B\times\gamma$.

Summarizing these three kinds of customers, we obtain $\rho_{2C}=\rho_B+\rho_{4C}+\rho_{6C}$.

Finally, let us compute the offered load to link L_{1C}, which is denoted by ρ_{1C}. Note that the link L_{1C} is used for the customers in node *B*, *C* and *D* that bound for main server as well as the customers in node *A*.

Note that the offered load to link L_{1C} from the customers in node *B*, *C* and *D* is the same as that of link L_{2C}. On the other hand, the offered load to link L_{1C} from node *A* is ρ_A, which is given by $\rho_A=\frac{\lambda_A\times\tau}{3600}$, where $\lambda_A=N_A\times\gamma$. Finally, we obtain $\rho_{1C}=\rho_A+\rho_{2C}$.

5.2 Decentralized Server Model

Network architecture for DSM is given in Fig.3. Here we have to be careful about the flow of traffic in this case. For DSM, the video programs that are located at sub server and cache servers have different penetration rates depending on the type of server, which implies that we have to discriminate which video file is rented from which server. What we need to know is the source of video as well as the offered load to the system.

Let δ_1 and δ_2 be the portion of video contents that is requested to a cache server and a sub server, respectively, by customers in a node. Let ρ_X denotes the offered load from node *X*. Now let us compute the offered load of each link, from which the capacity of each link in the network can be calculated.

First, let us compute the offered load of link L_{7D}, which is denoted by $\rho_{7D}=\rho_D\delta_1$, where ρ_D is the same as ρ_{6C}.

In the same way, we can compute the offered load of link L_{5D}, which is denoted by $\rho_{5D}=\rho_C\delta_1$, where ρ_C is the same as ρ_{4C}.

Now let us compute the offered load ρ_{6D} of link λ_{6D}. Link λ_{6D} is used for the customers in node *D* that bound for sub server and main server, offered load of which is $\rho_D(\delta_2-\delta_1)$ and $\rho_D(1-\delta_2)$, respectively. Therefore, we obtain $\rho_{6D}=\rho_D(1-\delta_1)$.

In the same way, we can compute the offered load of link L_{4D}, which is denoted by ρ_{4D}. Note that Link L_{4D} is used for the customers in node *C* that bound for the sub server and main server, offered load of which is $\rho_C(\delta_2-\delta_1)$ and $\rho_C(1-\delta_2)$, respectively. Therefore, we obtain $\rho_{4D}=\rho_C(1-\delta_1)$.

Now let us compute the offered load to link L_{3D} which is denoted by ρ_{3D}. Note that the link L_{3D} is used for the customers in node *C* and *D* that bound for the sub server as well as the customers located in node *B*. The customers in node *C* and *D* is accessing this server for requesting videos that are not stored in the cache server *C* and *D*, respectively, and the offered load to link L_{3D} from node *C* and *D* is $\rho_C(\delta_2-\delta_1)$ and $\rho_D(\delta_2-\delta_1)$, respectively. The customers in node *B* are accessing this server for renting the upper ψ % popular video program. The offered load to link L_{3D} from node *B* is $\rho_B\delta_2$. Summarizing these two types of customers, we obtain $\rho_{3D}=(\rho_C+\rho_D)(\delta_2-\delta_1)+\rho_B\delta_2$.

Now let us compute the offered load to link L_{2D}, which is denoted by ρ_{2D}. Note that the link L_{2D} is used for the customers in node *B*, *C* and *D* that bound for main server. The customers in node *B*, *C* and *D* is accessing this server for requesting videos that are not stored in the sub server *B*, cache servers *C* and *D*, respectively, and the offered load to link L_{2D} from node *B*, *C* and *D* is $\rho_B(1-\delta_2)$, $\rho_C(1-\delta_2)$ and $\rho_D(1-\delta_2)$, respectively. Summarizing these three kinds of customers, we obtain $\rho_{2D}=(\rho_B+\rho_C+\rho_D)(1-\delta_2)$.

Finally, let us compute the offered load to the link L_{1D}, which is denoted by ρ_{1D}. Note that the link L_{1D} is used for the customers in node *B*, *C* and *D* that bound for main server as well as the customers in node A that are accessing this server for renting the entire video program.

Note that the offered load to link L_{1D} from the customers in node B, C and D is the same as that of link L_{2D}. On the other hand, the offered load to link L_{1D} from node A is ρ_A. Summarizing these facts, we obtain $\rho_{1D} = \rho_A + \rho_{2D}$.

Now that the offered load to each link is obtained we can compute the required number of channels by using Erlang-B formula.

Ⅵ. Numerical Results and Discussion

In order to investigate the implication of the proposed model for the network design, let us carry out a numerical experiment. In our experiment, let us assume that the basic parameters for the video service are as follows: υ=100, ζ=10, ψ=50, γ=0.2, and τ=1hour.

Note that the parameter α is computed from (3) by taking into the fact that "*top 10% of video programs occupy 48% of rentals*"[14], from which we have α=0.248. From this result we obtain δ_1 =0.48 and δ_2=0.67.

Let us assume that all the video contents are in the form of HDTV program, which require bandwidth of 8Mbps per channel. The required GoS for the video flow, which is defined to be CBP, is assumed to be 1%, which is the same as that of the current phone service.

As to the number of customers, let us assume two different cases: homogeneous distribution (HD) and inhomogeneous distribution (IHD), the number of customers of which is given by:

HD: N_X=200, where X=A, B, C, and D
IHD: N_A=400, N_B=200, N_C=N_D=100

Note that the total number of customers is the same with 800 people. Note also that we assumed a small number of customers for the purpose of computational simplicity, and an extension to a large-scale network is trivial. Table1 illustrates the required link capacity for the network when we adopt CSM [unit: Mbps], whereas Table2 illustrates the result for DSM.

Table 1. Link capacity for CSM [Mbps]

Link ID	HD	IHD
L_{1C}	1,440	1,440
L_{2C}	1,104	768
L_{3C}	424	240
L_{4C}	424	240
Total	3,392	2,688

Table 2. Link capacity for DSM [Mbps]

Link ID	HD	IHD
L_{1D}	768	992
L_{2D}	416	296
L_{3D}	440	368
L_{4D}	248	144
L_{5D}	232	136
L_{6D}	248	144
L_{7D}	232	136
Total	2,584	2,216

From Table1 we found that the required capacity for the link depends on the distribution of the customers in the network. This implies that information about the distribution of customers in the network has to be carefully managed in the network if one wants to guarantee satisfactory GoS to the customers.

We can also find that the ratio between the required capacity of the link for DSM over CSM is 0.76 and 0.82 for the homogeneous and inhomogeneous customer distribution, respectively. Therefore, we can argue that DSM is more effective than CSM in the dimensioning of the link capacity because the required link capacity is smaller.

Note, however, that the high efficiency in the link capacity of DSM over CSM comes at the cost of increased price of storage cost. This is because the DSM requires 4 places for the storage of video contents, whereas the CSM requires only 1 place (at node A). If we assume that each video program requires S unit of storage space, then CSM requires only 100S unit of storage space, whereas DSM requires 170S unit of storage space (100S for main server, 50 for sub server, and $20S$ for two cache servers). Therefore, the cost for DSM is 70% higher than that for CSM.

Note also that we do not mind about the cost of update for the video servers when new videos are released, because the update is carried out in off-line at midnight when there is little traffic in the network.

From the above result we have the following finding: First, placement of the video server (main server, sub server, and cache server) imposes great impact on the required link capacity of the network. That is, DSM requires less bandwidth as compared to CSM. Second, the link capacity for VoD network is affected by the distribution of customers, too. As we have presented in this work, this is more evident for the tree-like topology where node A is heavily populated, whereas node *C* and *D* is less populated. Note that this is very similar to the current situation of Korea, where a large portion of people lives in the area near the capital and almost all the video distributors as well as the video contents are located in that area.

Therefore, this result can be applied to the design of the backbone network for the VoD service of Korea if the scale of the users and video contents is extended to the realistic data.

Ⅶ. Conclusions

In this work we proposed a practical method for the design of a distribution network of VoD service in South Korea. Specifically, we introduced three factors in the design of the network, which are the main contribution of this work. First, we introduced the realistic behavior of users about the popularity of users' request for the video contents, which is incorporated into the model by a power law distribution for the probability of video requirements. Second, we proposed a novel method to distribute the video contents based on three-level distributed server (main, sub, and cache) throughout the nodes based on the rank of the user-popularity, via which customers can download video contents from the nearest server. This results in saving of bandwidth in the backbone network for the video service provider. Finally, we introduced the concept of GoS to the design of link capacity for the VoD service. This is realistic because some users can be blocked from the service if there is no available bandwidth in the network. Therefore, the design of the link with the consideration of connection blocking probability provides the users with satisfactory GoS.

By carrying out numerical experiments we could find the effectiveness of the DSM as compared with CSM, which illustrated the validity and practical implication of the proposition.

There are lots of future research areas that can be evolved from this work. First, we will extend this model to a general one by extending to more general topology for the nodes and various architectures for the location of video servers. Second, we will investigate the user behavior in more detail by accumulating the real-field data about the popularity of video contents for the ongoing IPTV service in Korea. We will also investigate the user behavior about the network congestion. Especially, we will investigate whether users balk or renege in accessing to the VoD service when the network is congested. This can be incorporated into the estimation of the GoS.

Acknowledgments

The authors are grateful to the anonymous reviewers for their fruitful comments, which improved the presentation as well as the contents of the paper. This work is supported by NURI Project (Project #: 06가A02).

References

[1] K. Takahashi, “The nature of IPTV Technologies”, *NIKKEI COMMUNICATIONS*, May 15, 2008.

[2] S.V. Vasudevan, X. Liu, and K. Kollmansberger, “IPTV architecture for cable systems: An evolutionary approach”, *IEEE Communications Magazine*, May 2008.

[3] M. Cha et al., “Case study: resilient backbone design for IPTV services”, *IPTV Workshop,*

International WWW conference, May 23, 2006, Edinburgh, Scotland, UK.

[4] Z.M. Mao, et al., "Efficient and robust streaming provisioning in VPNs", *Proc. of the 12th International WWW Conference*, 20-24 May 2003, Budapest, Hungary

[5] B. Molina, C. E. Palau, and M. Esteve, "Modeling content delivery networks and their performance", *Computer Communications* 27 (2004).

[6] Hoon Lee, *Queuing systems for Engineers*, Hongneung Publishing co., 2008, Korea. [In Korean]

[7] G. O'Driscoll, Next generation IPTV services and technologies, *Wiley-Interscience*, 2007.

[8] Y. Takeda, Y. Kurihara, "Construction of fault-tolerant networks using server-load balancing", *NIKKEI COMMUNICATIONS*, May 15, 2008. [In Japanese]

[9] A. Nafaa, S. Murphy, and L. Murphy, "Analysis of a large-scale VoD architecture for broadband operators: A P2P-based solution", *IEEE Communication Magazine*, December 2008.

[10] J. Akiba and H. Abe, "Scalable content delivery technology", *NTT Technical Review*, Vol.1, No.7, October 2003.

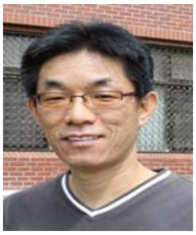
[11] D. Agrawal, M. S. Beigi, C. Bisdikian, K.-W. Lee, "Planning and managing the IPTV service deployment", *10th IFIP/IEEE Int'l symposium on integrated network management*, Munich, Germany, May 2007.

[12] T. Satake, Y. Chiba, and T. Okamoto, "Reference traffic for network resource dimensioning under unexpected and unevenly distributed traffic load", *IEICE Tr. on Communications* Vol.J91-B, No.7, 2008. [In Japanese]

[13] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-like distributions: evidence and implications", *Proc. of the INFOCOM*, March 1999.

[14] H.S. Kim, "If your tail is long, it will be bitten: Long tail theory-a reverse prospect", *Chosunilbo*, July 5th 2008.

Hoon Lee　　　Lifelong Member



B.E. and M.E. from Kyungpook National University

Ph.D. from Tohoku University, Japan.

Feb. 1986~Feb.2001 KT Network Laboratories

Feb. 2001~ Changwon National University

Mar.2005~Aug.2006 Visiting Scholar, University of Missouri-Kansas city, USA.

<Research fields> Network design, Performance evaluation, Internet QoS and charging