A New Optimization Model for Designing Broadband Convergence Network Access Networks

Youngho Lee[1], Jinmo Jung[1], Youngjin Kim[1]
Sunsuk Lee[2], Noik Park[2], Kukchang Kang[2]

[1]Department of Industrial Systems and Information Engineering, Korea University
Address: Sungbuk-gu Anam-dong 5-1, Seoul 136-701, Korea
Email: yhlee@korea.ac.kr, blueflat@korea.ac.kr
TEL: 02-3290-3390, FAX: 02-929-5888

[2]Electronics and Telecommunications Research Intitute
Address: Gajeong-dong, Yuseong-gu, Daejeon, 305-700, Korea
TEL: 042-860-6114

Abstract

In this paper, we deal with a network optimization problem arising from the deployment of Ethernet-based BcN access network. BcN convergence services require that access networks satisfy QoS measures. BcN services have two types of traffics: stream traffic and elastic traffic. Stream traffic uses blocking probability as a QoS measure, while elastic traffic uses delay factor as a QoS measure. Incorporating the QoS requirements, we formulate the problem as a nonlinear mixed-integer programming model. The proposed model seeks to find a minimum cost dimensioning solution, while satisfying the QoS requirement. We propose tabu search heuristic algorithms for solving the problem, and simulate tabu result. We demonstrate the computational efficacy of the proposed algorithm by solving a network design problem.

Key Words: BcN, QoS, Access Network, Mixed Integer Programming, Tabu Search, Heuristic Algorithm

1. Introduction

In this paper, we deal with an access network design problem arising from the deployment of BcN services. BcN, also known as Next Generation Network (NGN), is network which provides multi-service such as telecommunication service, broadcast service and internet service. [11] J. C. Crimi [4] and P. J. Kuhn [12] defined various NGN service. Network dimensioning and QoS are two key components to provide multi-service QoS guarantees without overprovision. Network dimensioning makes the networks not to work with overburden always. Network engineering has to resolve the trade-off between capacity and QoS requirements.

This paper discusses network dimensioning of the access network domain in BcN. BcN consist of service and control domain, transport network domain and access network domain. [11] Service and control domain plays a role of creating service applications through Open Application Programming Interface (Open API) platform and Soft Switch. Transport network domain is in charge of Quality of Service (QoS) which provide voice, data and multimedia services. Access network domain supports accessibility to provide connectedness anywhere, anytime telecommunication accessibility, broadcast, or internet. Critical issue of BcN access network design problem is designing economical access network while guaranteeing QoS of each service.

We discuss BcN access network problem based on Ethernet. Architecture of the Ethernet-based BcN access network is illustrated in Figure 1. [17] Residential Gateway (RG) is a terminal to deliver all BcN service traffics generated by TV, PC, and IP phone in home. Traffics in home are carried to Primary Layer 2 (P-L2) switch and Secondary Layer 2 (S-L2) switch. The S-L2 switch is in charge of user accessing and P-L2 switch is in charge of aggregating traffics in the S-L2 switches. And traffics arrive to Access Edge Service Node (A-ESN) which provides connection between access network and IP transport network such as Synchronous Optical Network (SONET) and Synchronous Digital Hierarchy (SDH). We focus on dimensioning of Ethernet-based access network from the RG to the A-ESN and deal with end-to-end QoS requirement through the access network.

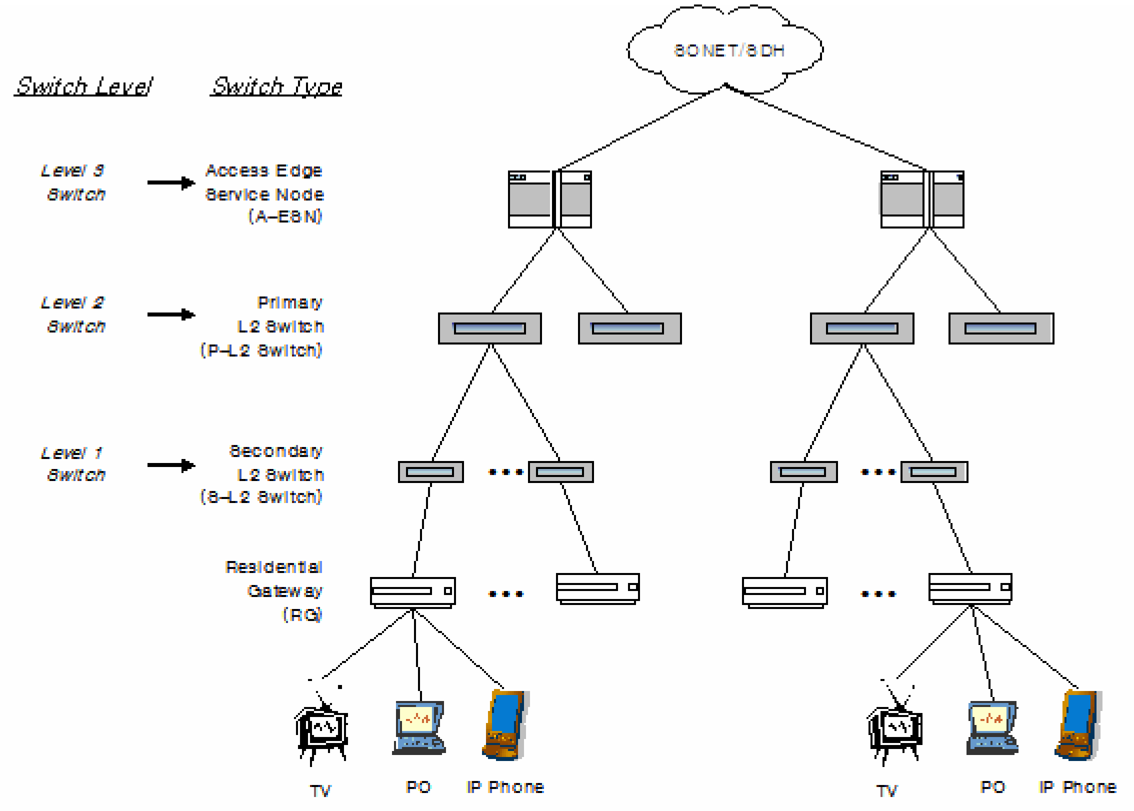


Figure 1. Architecture of Ethernet-based BcN access network [17]

Data packets need to be classified and assigned to a specific flow or class, which can be done either at the network edge (Differentiated Services, DiffServ [1]) or at every individual router along the path (Integrated Services, IntServ [2]). From a network planning perspective it is advisable to initially ignore the very details of the different traffic control schemes and only consider the two main implications: (1) Fixed bandwidth partitioning, i.e., predictable capacity assignment to individual flows or classes without detrimental interference, can be assumed. However, potential multiplexing gains in case of work-conserving systems are neglected. (2) Within an allocated bandwidth fraction, certain traffic streams can be treated preferentially. [14] Based on these two concepts, we apply traffic control method that assigns aggregated traffics to switch capacity and considers QoS limit of each service.

For analyzing traffics in home, we classify BcN services to stream and elastic by traffic type. [13] Stream traffic is associated with real-time services, which have strict QoS requirements concerning the transport of packets over IP networks. For these applications the packets have to be delivered in a timely manner. Elastic traffic is generated by non-real time applications and is carried by the Transmission Control Protocol (TCP), the main transport protocol for data in IP networks. Since stream and elastic traffic are differed from traffic distribution, service, and QoS mechanism, we consider network dimensioning with guaranteeing QoS by traffic type.

For the last decade, several studies have been conducted for network designing problem with QoS requirement. International Telecommunication Union – Radio (ITU-R) presented network dimensioning method for Third Generation Wireless Network (3G). [7, 8] ITU-R divided 3G services into simple service and multimedia service, and classified 3G users according to region, environment, and service. When calculating total traffic, ITU-R estimated traffics per cell. However they deal with wireless network only through cell, then it cannot be applied to Ethernet-based BcN access network. Riedl et al. provided network designing framework for BcN service. [14] To provide a foundation for any network dimensioning procedure, they first introduce and categorize current IP Quality of Service mechanisms and identify their implications for network planning. However they cannot expand optimization model that deals with stream traffic and elastic traffic at the same time. Zhao et al. studied dimensioning method for Broadband Residential Ethernet-based Access Network (BREAN) using Differential Service (DiffServ), and expends or amends some study results with DiffServ, such as the universal link model, the dimensioning for the elastic traffic and the stream traffic, traffic matrix and user model. [17] DiffServ is QoS mechanism to deal with traffic to the priority order decided service class. They validated model of network dimensioning by the simulations with pragmatic engineering method. However they did not consider guaranteeing QoS requirement for end-to-end blocking probability of traffic. A separate way to research network designing algorithm, Tsang et al. provided algorithm calculating blocking probability each service on tree network and estimated provided algorithm. [16] They used convolution function, which was calculated blocking probability of tree network by convolution and deconvolution. However they only assumed topology of binary tree, then it is difficult to apply BcN access network.

In this paper, we deal with the Ethernet-based BcN access network design problem considering trade-off between QoS requirement and cost of switches. We formulate optimization model that objective function is total cost of switches and constraints are QoS requirements. Then we develop heuristic algorithm to solve the problem. The remainder of this paper is organized a follows: In Section 2, we present problem. In section 3, we propose tabu search heuristic algorithm with short term and longer term.

In section 4, some preliminary computational results are presented. In section 5, we conclude this paper and propose future research.

2. Problem Formulation

In this section, we develop the optimization model for designing access network to guarantee QoS. First we classify BcN services into two service class type by characteristic of traffic. Next we decide QoS measure and method calculating QoS level. Finally we design optimization model for access network that objective function is minimizing total cost and constraint is guaranteeing QoS requirement of each service. Let us classify BcN services into 5 service classes by traffic type, access bit rate, traffic asymmetry. Traffic type means stream traffic and elastic traffic characteristic, access bit rate means number of bit connecting link per unit time on end of joining service. And traffic asymmetry means discrepancy between up-link traffic and down-link traffic. Traffic asymmetry often is occurred broadcast service. [17] Figure 2 shows BcN service class to classify 5 classes by traffic type, access bit rate, traffic asymmetry.

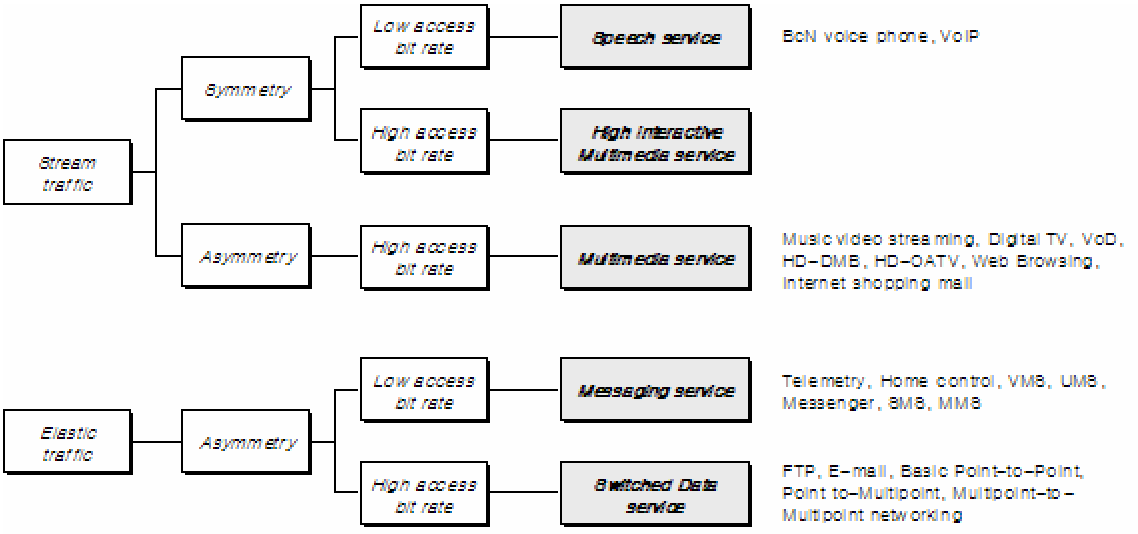


Figure 2. BcN service class

After classifying BcN service class, we need to press out parameters for calculating traffic of each service class. Parameters for calculating traffic of each service class are different by the service class, especially stream traffic and elastic traffic. Stream traffic deals with access bit rate, Busy Hour Call Attempt (BHCA) and call duration. Elastic traffic deals with file size instead of call duration and concurrent connecting rate that is probability connecting session at the same time.

To guarantee QoS of each service, we define method of calculating QoS. Method of calculating QoS is differed with stream traffic and elastic traffic. Stream traffic provide service real time and have characteristic that one call possess one channel until service end. Therefore call of stream traffic cannot connect same service and must connect end-to-end since stream traffic transport traffic real time. If arrival time probability is Poisson density function, stream traffic can be modeling by M/G/C/C queuing model. [13] Blocking probability that is QoS measure of stream traffic can be calculating Erlang-B formula on M/G/C/C queuing model. When access network is tree structure of 3-level as depicted in Figure 3, we calculate end-to-end blocking probability for stream traffic. End-to-end blocking probability means rate of disconnecting calls between RG and A-ESN. To calculate end-to-end blocking probability, we define state that number of links connecting S-L2 switch, because service path is defined that any S-L2 switch connect service on tree structure access network.

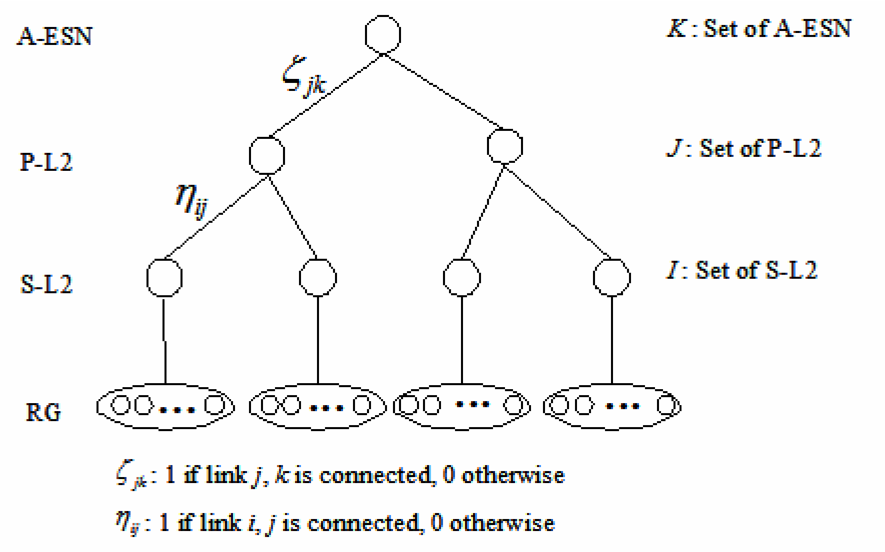


Figure 3. Structure of tree access network

Switch channel capacity is defined that switch capacity divided by access bit rate, Q^S is S-L2 switch channel capacity, Q^P is P-L2 switch channel capacity, Q^A is A-ESN switch channel capacity. Then state set T is defined by (1). [9]

$$\begin{aligned} T = \{n \in Z^{|I|} : n_i \le Q^S, &\quad \forall i \in I, \\ \sum_{i \in I} \eta_{ij} n_i \le Q^P, &\quad \forall j \in J, \\ \sum_{i \in I} \sum_{j \in J} \eta_{ij} \zeta_{jk} n_i \le Q^A, &\quad \forall k \in K\}. \end{aligned} \tag{1}$$

Service possible state of user connecting S-L2 switch i^* is equal to state that channel of S-L2 switch i^*, P-L2 switch j^* of η_{ij} and A-ESN switch k^* of ζ_{jk} is empty more than one. Service possible state is defined by (2).

$$\begin{aligned} T_{i^*} = \{n \in T : n_{i^*} \le Q^S - 1, & \\ \sum_{i \in I} \eta_{ij} n_i \le Q^P - 1, &\quad \text{for } j = j^*, \\ \sum_{i \in I} \sum_{j \in J} \eta_{ij} \zeta_{jk} n_i \le Q^A - 1, &\quad \text{for } k = k^*\}. \end{aligned} \tag{2}$$

Blocking is occurred on service impossible state. Therefore end-to-end blocking probability B_{i^*} of user connecting S-L2 switch i^* is defined by (3). [9, 10]

$$B_{i^*} = 1 - \frac{\sum_{n \in T_{i^*}} \prod_{i \in I} \frac{A_i^{n_i}}{n_i!}}{\sum_{n \in T} \prod_{i \in I} \frac{A_i^{n_i}}{n_i!}}. \tag{3}$$

To embed end-to-end blocking probability (3) into optimization model, we use Reduced Load Fixed Point Approximation. [15] This is approximation that calculates end-to-end blocking probability through blocking probability of each switch level. Traffic transporting to low level switch is decrease by blocking probability of low level switch, accordingly blocking probability of each switch is defined by (4), (5), (6).

$$S-L2:\quad B^{S_i}=E_B(A_i,Q^S),\tag{4}$$

$$P-L2:\quad B^{P_j}=E_B\left(\sum_{i\in I}\eta_{ij}A_i(1-B^{S_i}),\,Q^P\right),\tag{5}$$

$$A-ESN:\quad B^{A_k}=E_B\left(\sum_{i\in I}\sum_{j\in J}\eta_{ij}\zeta_{jk}A_i(1-B^{S_i})(1-B^{P_j}),\,Q^A\right).\tag{6}$$

Now, let us define end-to-end blocking probability through blocking probability of each switch level (5), (6), (7), because traffic intensity is decreased by blocking probability occurred low level switch. End-to-end blocking probability shows (8).

$$B_{i*}=1-(1-B^{S_i})(1-B^{P_j})(1-B^{A_k}).\tag{8}$$

Elastic traffic provide service non real time and divide data to packet in contrast with stream traffic. [13, 14] Also elastic traffic transport TCP level and deal with concurrent connecting rate due to divide data. If arrival time probability is Poisson density function and network is tree structure with 1-level, elastic traffic can be modeling by M/G/C Process Sharing (PS) model. [13] Delay factor that is QoS measure of elastic traffic can be satisfy QoS requirement, and delay factor D is calculated by (9).

$$D=1+\frac{E_C(C\rho,C)}{C(1-\rho)}.\tag{9}$$

Let C be the number of server and be calculating that server capacity R is divided by access bit rate. λ be the BHCA, l be the file size. Furthermore, let ρ be the utilization calculating by $\lambda l/R$, and $E_C(\cdot)$ is Erlang-C formula.

To embed end-to-end delay factor (3) into optimization model, we calculate delay factor of each switch level. Since elastic traffic allows delay of channel and it influences end-to-end delay factor, we first determine delay factor of each switch, defined by (10), (11), (12).

$$S-L2:\quad D_s^S = 1 + \frac{E_C(A_i, C^S)}{C^S - A_i}, \tag{10}$$

$$P-L2:\quad D_s^P = 1 + \frac{E_C(\sum_{i\in I}\eta_{ij} A_i, C^P)}{C^P - A_i}, \tag{11}$$

$$A-ESN:\quad D_s^A = 1 + \frac{E_C(\sum_{i\in I}\sum_{j\in J}\eta_{ij}\zeta_{jk} A_i, C^A)}{C^A - A_i}, \tag{12}$$

Now, let us define end-to-end delay factor through delay factor of each switch level (10), (11), (12), because total delay is increased by delay factor occurred low level switch. End-to-end delay factor shows (13).

$$D = D_s^S D_s^P D_s^A \tag{13}$$

To model the access network design problem, we assume that all users are distributed geographical uniformly. Therefore user rate connecting S-L2 switch are equal to any S-L2 switch. We deal with same type of switch on same level and except link cost. To avoid that traffic occur special section concentrative, we assume that number of low level switches connecting high level switch is same to any high level switch. And we decide switch capacity compared up-link capacity with down-link capacity, since BcN access network provide up-link capacity and down-link capacity equally. Then the sets and parameters for optimization problem designing access network are defined as follows:

S: set of service class
S_{stream}: set of service of stream traffic
$S_{elastic}$: set of service of elastic traffic
N: total number of user
A_s: traffic intensity of service s
r_s: access bit rate of service s
Q_S: capacity of S-L2 switch
Q_P: capacity of P-L2 switch
Q_A: capacity of A-ESN switch
a: cost of S-L2 switch
β: cost of P-L2 switch

γ: cost of A-ESN switch
ε_s: blocking probability limit of service s
δ_s: delay factor limit of service s
M: constant

We define decision variable x as number of user to be connected to single S-L2, y^S as number of S-L2 switch to be connected to single P-L2, y^P as number of P-L2 switch to be connected to single A-ESN, and y^A as number of A-ESN switch. And we define u_s as the rate of S-L2 switch capacity of service s, v_s as the rate of P-L2 switch capacity of service s, w_s as the rate of A-ESN switch capacity of service s. Then the optimization problem designing access network can be formulated as follows:

Minimize $\alpha y^S y^P y^A + \beta y^P y^A + \gamma y^A$ (14)

Subject to

$$x = \left\lceil \frac{N}{y^S y^P y^A} \right\rceil \tag{15}$$

$$B_s^S = E_B\left(xA_s, \left\lfloor \frac{C^S u_s}{r_s} \right\rfloor \right), \quad s \in S_{stream}, \tag{16}$$

$$B_s^P = E_B\left(xy^S A_s(1-B_s^S), \left\lfloor \frac{C^P v_s}{r_s} \right\rfloor \right), \quad s \in S_{stream}, \tag{17}$$

$$B_s^A = E_B\left(xy^S y^P A_s(1-B_s^S)(1-B_s^P), \left\lfloor \frac{C^A w_s}{r_s} \right\rfloor \right), \quad s \in S_{stream}, \tag{18}$$

$$1-(1-B_s^S)(1-B_s^P)(1-B_s^A) \le \varepsilon_s, \quad s \in S_{stream}, \tag{19}$$

$$D_s^S = 1 + \frac{E_C(xA_s, \lfloor C^S u_s / r_s \rfloor)}{C^S u_s - xA_s}, \quad s \in S_{elastic}, \tag{20}$$

$$D_s^P = 1 + \frac{E_C(xy^S A_s, \lfloor C^P v_s / r_s \rfloor)}{C^P v_s - xy^S A_s}, \quad s \in S_{elastic}, \tag{21}$$

$$D_s^A = 1 + \frac{E_C(xy^S y^P A_s, \lfloor C^A w_s / r_s \rfloor)}{C^A w_s - xy^S y^P A_s}, \quad s \in S_{elastic}, \tag{22}$$

$$D_s^S D_s^P D_s^A \le \delta_s, \quad s \in S_{elastic}, \tag{23}$$

$$\sum_{s\in S} u_s = 1, \sum_{s\in S} v_s = 1, \sum_{s\in S} w_s = 1, \tag{24}$$

$$u_s \ge 0, \quad s \in S, \tag{25}$$

$$v_s \ge 0, \quad s \in S, \tag{26}$$

$$w_s \ge 0, \quad s \in S, \tag{27}$$

$$x, y^S, y^P, y^A \ge 0, \ \textit{and integer.} \tag{28}$$

The first term of the objective function is the sum of cost of S-L2 switches. The second term is the sum of cost of P-L2 switches and Third term is the sum of cost of A-ESN switches. These cost include only switch cost. Constraint (15) restricts the number of user connecting S-L2 switch. Constraints (16), (17), (18) calculate blocking probability of each switch level. Constraint (19) restricts end-to-end blocking probability to be less than or equal to the requirement QoS limit. Constraints (20), (21), (22) calculate delay factor of each switch level. Constraint (23) restricts delay factor to be less than or equal to the requirement QoS limit. Constraint (24) ensures that the sum of rate of each switch capacity is equal to 1 and constraints (25), (26), (27) ensures that rate of each switch capacity are always nonnegative. Constraints (28) also ensure that decision variables are nonnegative and integer.

3. Tabu Search Heuristic Algorithm

Optimization model designing access network is composed integer decision variable and nonlinear constraints. Therefore this model is nonlinear mixed integer programming problem. It thus provides opportunities for the application of heuristic methods. In recent years, tabu search (TS) has been applied with a high degree of success to a variety of nonlinear mixed integer programming problems. It is basically an iterative neighborhood search strategy that allows moves that degrade the objective function. Through such moves, the method can escape from bad local. To avoid cycling, a short term memory, known as the tabu list, stores previously visited solutions or components of previously visited solutions. It is then forbidden or tabu to come back to these solutions for a certain number of iterations. Our TS heuristic follows the general guidelines provided in F. Glover. [5, 6]

We first briefly sketch the general algorithmic structure of our TS heuristic. Figure 4 shows our TS algorithm structure. Each component is then presented in greater detail in the following subsections.

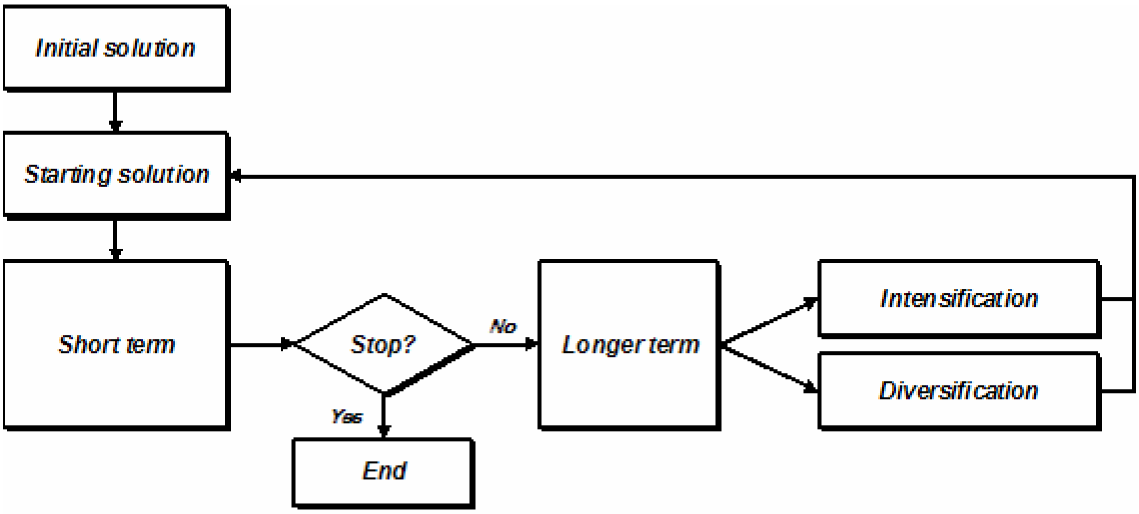


Figure 4. Tabu search algorithm structure

In this algorithm, the variables s^* and s^{best} are used to store the best solution of the current TS restart and best overall solution, respectively.

3.1. Initial solution

At the outset, an initial solution is produced by local search heuristic algorithm. Local search heuristic algorithm decides number of A-ESN switch. After that, we decide number of P-L2 and S-L2. Figure 5 shows local search heuristic algorithm for initial solution. This local search heuristic algorithm gives network structure of sufficient QoS.

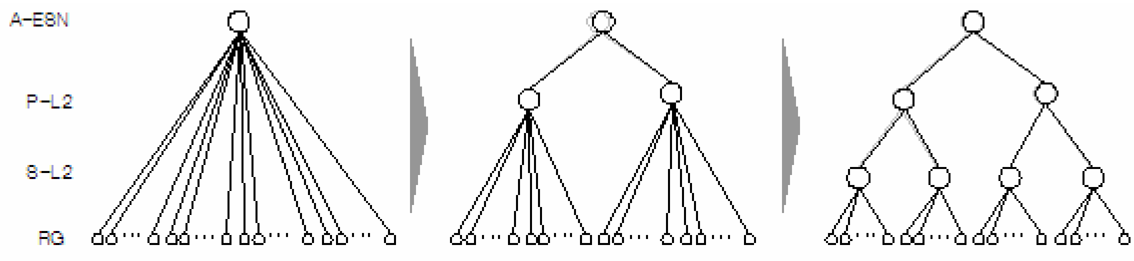


Figure 5. Local search heuristic algorithm for initial solution

3.2. Starting solution

TS is search procedure to explore the solution space beyond local optimality. TS do not guarantee that local optimality is same global optimality. If we have good starting solution for TS, we have more chance to search global optimum. We use starting solution as section 3.1 first, and other solutions as solution of longer term procedure.

3.3. Short term tabu search

An important distinction in TS arises by differentiating between short term memory and longer term memory. Each type of memory is accompanied by its own special strategies. However, the effect of both types of memory may be viewed as modifying the neighborhood $N(x)$ of the current solution x. We deal with neighborhood in section 3.3.1. The modified neighborhood, which we denote by $N^*(x)$, is the result of maintaining a selective history of the states encountered during the search. In the TS strategies based on short term considerations, $N^*(x)$ characteristically is a subset of $N^*(x)$, and the tabu classification serves to identify elements of $N(x)$ excluded from $N^*(x)$. In TS strategies that include longer term considerations, $N^*(x)$ may also be expanded to include solutions not ordinarily found in $N(x)$.

We compose short term tabu search as neighborhood, move, evaluation function, recency-based memory, tabu list, aspiration criteria, and stopping rule.

3.3.1. Neighborhood

The neighborhood structure is the most important issue in the development of a tabu search heuristic algorithm. We define a solution (*S-L2 switches connecting P-L2 switch, P-L2 switches connecting A-ESN switch, A-ESN switches*) as tree structure. At any given iteration, the current solution is a tree. And we assume that all S-L2 switches have same number of RG. We define a neighbor of that solution as all the trees that can be obtained from it by a add and drop operation that keeps the tree topology, namely
(1) add a switch to the current tree and
(2) drop a switch to the current tree

3.3.2. Move

A move is defined by a triplet of node numbers (i, j, k). Node i means change of S-L2 switches connecting P-L2 switch, and node j means change of P-L2 switches connecting A-ESN switch, and node k means change of A-ESN. We define the move based on add and drop operation of each switch. We classify moves as

$m^*(i, j, k)$ the move that would yield a feasible tree with a cost z lower than that of the best feasible solution so far,

$m_b(i, j, k)$ the move not on the tabu list that produces the infeasible solution with the least penalized cost ˜z and

$m_g(i, j, k)$ the tabu move that yields the lowest penalized cost.

3.3.3. Evaluation function

An important element of the procedure is the rule for evaluating the quality of a solution. For the feasible solution, this is simply the value of the objective function z to describe cost of switches. For infeasible solution, we define a penalized objective $\check{z}$ as follows. Let B_s be the blocking probability of service s of current solution and ε_s be the given blocking probability of service s. The amount of constraint violation for blocking probability is defined as $B_s^+ = \max \{B_s - \varepsilon_s\}$. Let D_s be the delay factor of service s of current solution and δ_s be the given delay factor of service s. The amount of constraint violation for delay factor is defined as $D_s^+ = \max\{D_s - \delta_s\}$. Given a penalty coefficient of blocking probability α and delay factor β, the penalized cost of infeasible solution, denoted $\breve{z}$ is given by

$$\breve{z} = z + \alpha\sum_{s=1}^{S} B_s^+ + \beta\sum_{s=1}^{S} D_s^+ \tag{24}$$

3.3.4. Recency-based memory

The most commonly used short term memory keeps tack of solutions that have changed during the recent past, and is called recency-based memory. We use recency-based memory to determine that move is tabu and to update tabu list.

3.3.5. Tabu lists

A move is declared tabu simply by declaring that move is itself tabu and cannot be used for a certain number of iterations. As such, there is no distinct tabu list and the tabu character of a move is stored in the data structure. A move keeps its tabu status for k iterations where k is randomly chosen in a given interval.

3.3.6. Aspiration criteria

Tabu restrictions may be overridden under certain conditions, called aspiration criteria. Aspiration criteria are possible and we have chosen the simplest one mostly for ease of implementation. Aspiration criteria define rules that govern whether next configuration is considered as a possible move even it is tabu. One widely used aspiration criterion consists of removing a tabu classification from a move when the move leads to a solution better than that obtained. However aspiration criteria can occur cycle, we use penalized value of tabu in evaluation function. This penalized value makes aspiration criteria more tightly.

3.3.7. Stopping criteria

We have used a simple stopping rule: the algorithm stops if there has been K=100× $Num(U)$, number of user, iterations without improving the feasible solution or if there are no possible moves from the current point. This rule has the advantage that it depends on the size of the problem so that we run less chance of overlooking a good solution in a large problem than if we used a fixed iteration count as the stopping rule.

The short term tabu search algorithm can be described as follows, with s the iteration index.

1. Initialization.
2. Compute $m^*(i, j, k)$, $m_b(i, j, k)$ and $m_g(i, j, k)$ in the current neighborhood.
3. If $m^*(i, j, k)$ exists, update the best feasible solution. This is equivalent to making a move that improves the cost while leading to a feasible tree. This move is made even if it is tabu.
4. Otherwise if the real cost produced by $m_b(i, j, k)$ is lower than that of the current solution, then apply the move.
5. Otherwise, apply the move $m_g(i, j, k)$. In the end, if we cannot improve the solution by a feasible or an infeasible move, we forget about the tabu list and make the best infeasible move.
6. Update solution.
7. Stop when $s = N$.
8. End.

3.4. Longer term tabu search algorithm

The main handicaps of the short term tabu search are the search over the full neighborhood, the random nature of the tabu list and the choice of the move at iteration and these elements, with some others, have been expanded. Longer term tabu search are composed by longer term memory and its associated strategies. In the longer term tabu search strategies, the modified neighborhood produced by tabu search may contain solutions not in the original one, generally consisting of selected elite solutions encountered at various points in the solution process.

Two highly important components of tabu search are intensification and diversification strategies. Intensification strategies are based on modifying choice rules to encourage move combinations and solution features historically found good. Diversification strategies encourage the search process to examine unvisited regions and to generate solutions that differ in various significant ways from those seen before.

We compose longer term tabu search as longer term starting point, frequency-based memory, intensification strategy, and diversification strategy.

3.4.1. Longer term starting point

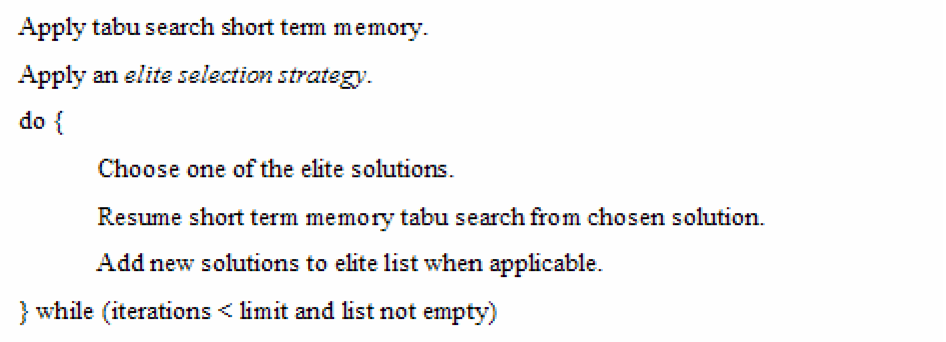
If short term tabu search is sufficient to stopping rule, longer term tabu search start. At that time, tabu search algorithm inserts best solution saving recency-based memory into elite solution. Elite solution is described in section 3.4.3. Recency-based memory is initialized and we go to longer term procedure.

3.4.2. Frequency-based memory

Frequency-based memory provides a type of information that complements the information provided by recency-based memory, broadening the foundation for selecting preferred moves. Frequency-based memory is a tabu search structure that remembers the number of solution. Frequency-based memory is not initialized after short term and longer term, because we determine new starting solution of short term through this memory.

3.4.3. Intensification

Intensification strategies are based on modifying choice rules to encourage move combinations and solution features historically found good. A simple instance of intensification strategy is shown as pseudo code in figure 6. [6]



```
Apply tabu search short term memory.
Apply an elite selection strategy.
do {
        Choose one of the elite solutions.
        Resume short term memory tabu search from chosen solution.
        Add new solutions to elite list when applicable.
} while (iterations < limit and list not empty)
```

Figure 6. Simple tabu search intensification approach [6]

The strategy for selecting elite solutions is the key of intensification strategy. In this paper, we apply backtrack strategy that user can track the search back to the point where it found the best solution. In the backtrack strategy, we determine one elite solution. Next we track the search back fixed number of steps from elite solution by using recency-based memory. Finally back tracking moves assign tabu, because object of longer term tabu search is exploring unvisited search space.

3.4.4. Diversification

We apply diversification strategies to increase effectiveness in exploring the solution space defined by an access network dimensioning problem and to prevent searching process from cycling, i.e., from endlessly executing the same sequence of moves. Frequency information can be used in different ways to design restarting mechanism within tabu search. Basic idea of restarting mechanism is that new starting solution is determined by random selecting among feasible solutions. We consider only feasible solution which of frequency is below two, because of problem scale.

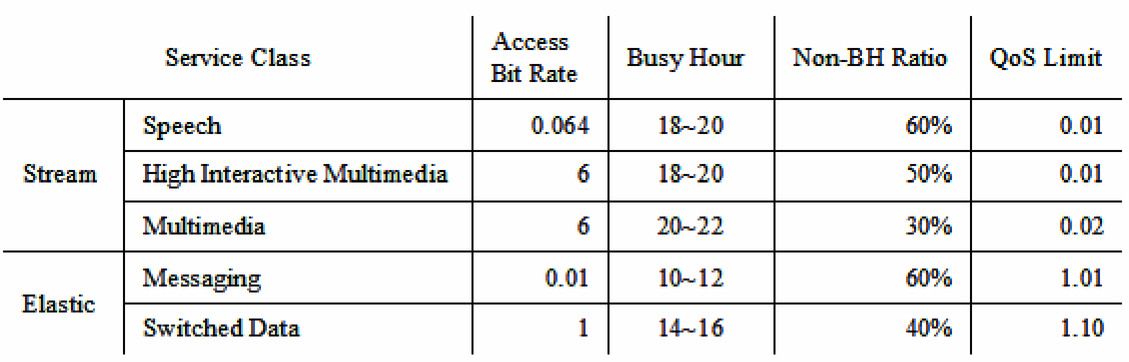
4. Computational Results

The value of a heuristic optimization method can only be evaluated from numerical results. We have chosen access networks with 1,000 users as test cases. The results that are presented here are mostly to illustrate the impact of the parameter changes that we made on the quality of the solution. We will also see that the simulation result of our solutions.

4.1. Artificial test network

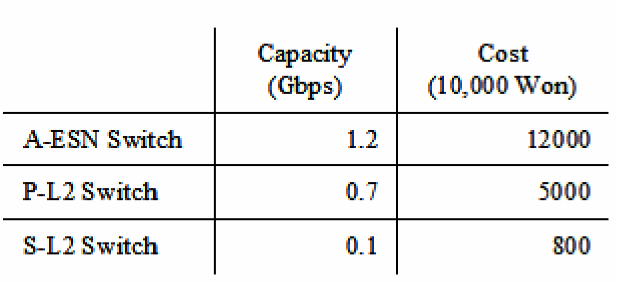
Our test environments are same as those assumptions about topology, services, QoS mechanisms in section 2. In order to accurately validate the effect of our method of network dimensioning, we test six cases which of parameters are different. In case 1, the data for user behavior is given in table 1 which we have the service class, access bit rate in Mbps, busy hour in Sec, and non busy hour ratio in %. Since all users do not have same behavior for BcN services, we divide user into three types of heavy user (30%), middle user (40%), and light user (30%). In our optimization model, objective function is minimizing total switch cost, and their capacity and cost are found in table 2. Table 3 shows user type and parameters which we have the busy hour call attempt in Calls/Sec, file size for elastic traffic in Mbps, call duration for stream traffic in Sec, and concurrent connecting rate for elastic traffic in %. Total user number *N* is 10,000.

Table 1. Parameters for user behavior of service class – Case 1



Service Class		Access Bit Rate	Busy Hour	Non-BH Ratio	QoS Limit
Stream	Speech	0.064	18~20	60%	0.01
	High Interactive Multimedia	6	18~20	50%	0.01
	Multimedia	6	20~22	30%	0.02
Elastic	Messaging	0.01	10~12	60%	1.01
	Switched Data	1	14~16	40%	1.10

Table 2. Switch capacity and cost – Case 1

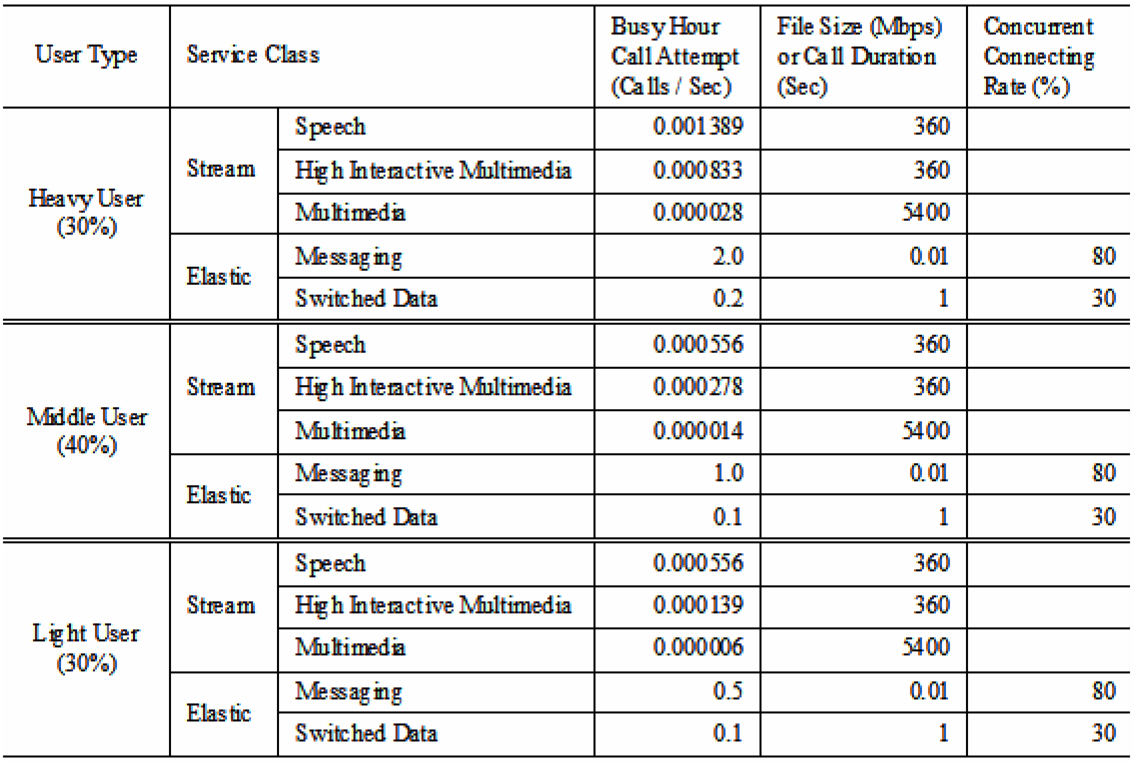


	Capacity (Gbps)	Cost (10,000 Won)
A-ESN Switch	1.2	12000
P-L2 Switch	0.7	5000
S-L2 Switch	0.1	800

Since users are classified by three groups, each parameter has different value and makes different traffics. Therefore we defined aggregated traffic as sum of traffics from three groups.

In case 2, we investigate the QoS limit with light burden, and the result indicates the network can design more chipper than case 1. In case 3, we increase access bit rate of each service, and the result indicates we need more switches for guaranteeing QoS limit. In case 4, we increase switch capability fixing switch cost, and the result indicates total cost decrease remarkably. In case 5, we increase busy hour call attempt of each service, and result shows network can design very high total cost. Finally, in case 6, we increase file size and call duration, and result indicates it is increasing total cost.

Table 3. Parameters for user type – Case 1



User Type	Service Class		Busy Hour Call Attempt (Calls / Sec)	File Size (Mbps) or Call Duration (Sec)	Concurrent Connecting Rate (%)
Heavy User (30%)	Stream	Speech	0.001389	360	
		High Interactive Multimedia	0.000833	360	
		Multimedia	0.000028	5400	
	Elastic	Messaging	2.0	0.01	80
		Switched Data	0.2	1	30
Middle User (40%)	Stream	Speech	0.000556	360	
		High Interactive Multimedia	0.000278	360	
		Multimedia	0.000014	5400	
	Elastic	Messaging	1.0	0.01	80
		Switched Data	0.1	1	30
Light User (30%)	Stream	Speech	0.000556	360	
		High Interactive Multimedia	0.000139	360	
		Multimedia	0.000006	5400	
	Elastic	Messaging	0.5	0.01	80
		Switched Data	0.1	1	30

4.2. Tabu Search

For each heuristic, we then calculated the gap of objective value defined by

$$Gap = \frac{Z^{short} - Z^{longer}}{Z^{short}}$$

In this expression, Z^{short} is the objective value of tabu search applied short term process only. Z^{longer} is the objective value of tabu search applied longer term. Table 4 presents a summary of the results.

Table 4. Computational results for tabu search

N= 10,000	Apply short term only				Apply longer term				
	A-ESN	P-L2	S-L2	Z^{short}	A-ESN	P-L2	S-L2	Z^{longer}	*Gap*
Case 1	18	3	5	702,000	18	2	9	655,200	0.0714
Case 2	18	3	4	658,800	18	2	7	597,600	0.1024
Case 3	26	3	3	889,200	24	2	8	835,200	0.0647
Case 4	18	2	3	482,400	18	1	10	450,000	0.0720
Case 5	28	2	7	929,600	28	3	2	890,400	0.0440
Case 6	22	4	5	1,056,000	21	3	9	1,020,600	0.0347

4.4. Simulation study

Our simulation environments are same as tabu search. Noted, we generated the stream traffic by explicitly access bit rate, busy hour, non-busy hour ratio, busy hour call attempt, and call duration. And we generated the elastic traffic by access bit rate, busy hour, non-busy hour ratio, busy hour call attempt, file size, and concurrent connecting rate. Simulation time is 3,000,000 and we calculate blocking probability and delay factor. Before we simulate each case, we determine warm-up period. In our simulation, warm-up period is 200,000. Figure 6 presents warm-up period of speech service in case 1.

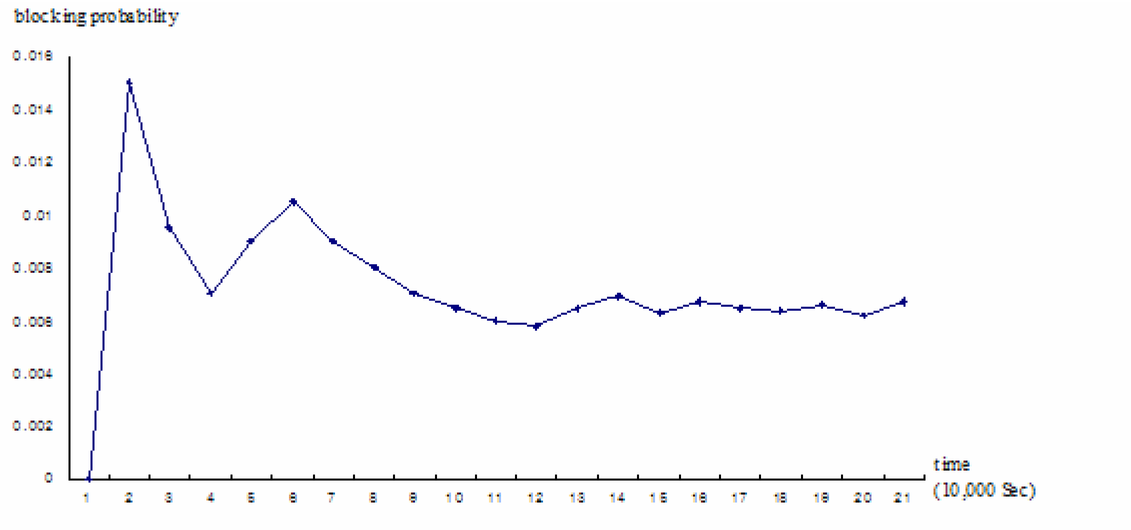


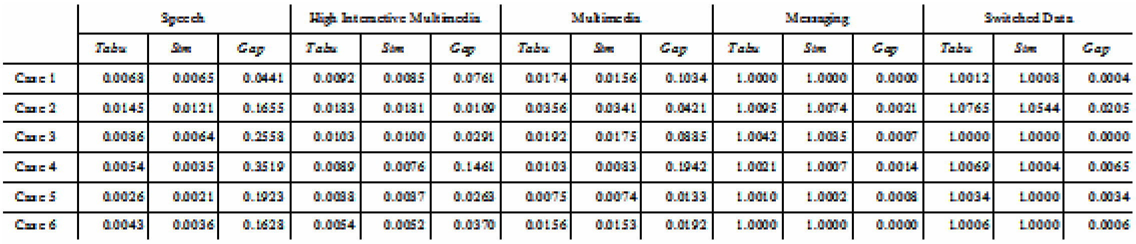
Figure 6. Warm-up period of speech service in case 1

Table 5 presents a summary of simulation results. For each simulation, we then calculated the gap of QoS defined by

$$Gap = \frac{Tabu - Sim}{Tabu}$$

In this expression, $Tabu$ is the QoS from tabu search applied longer term process. Sim is the QoS from simulation. The simulation results illustrate that QoS from simulation are less than QoS from tabu search in all cases. These results are caused by approximation for calculating end-to-end QoS.

Table 5. Simulation results for QoS



	Speech			High Interactive Multimedia			Multimedia			Messaging			Switched Data		
	Tabu	*Sim*	*Gap*	*Tabu*	*Sim*	*Gap*	*Tabu*	*Sim*	*Gap*	*Tabu*	*Sim*	*Gap*	*Tabu*	*Sim*	*Gap*
Case 1	0.0068	0.0065	0.0441	0.0092	0.0085	0.0761	0.0174	0.0156	0.1034	1.0000	1.0000	0.0000	1.0012	1.0008	0.0004
Case 2	0.0145	0.0121	0.1655	0.0183	0.0181	0.0109	0.0356	0.0341	0.0421	1.0095	1.0074	0.0021	1.0765	1.0544	0.0205
Case 3	0.0086	0.0064	0.2558	0.0103	0.0100	0.0291	0.0192	0.0175	0.0885	1.0042	1.0035	0.0007	1.0000	1.0000	0.0000
Case 4	0.0054	0.0035	0.3519	0.0089	0.0076	0.1461	0.0103	0.0083	0.1942	1.0021	1.0007	0.0014	1.0069	1.0004	0.0065
Case 5	0.0026	0.0021	0.1923	0.0038	0.0037	0.0263	0.0075	0.0074	0.0133	1.0010	1.0002	0.0008	1.0034	1.0000	0.0034
Case 6	0.0043	0.0036	0.1628	0.0054	0.0052	0.0370	0.0156	0.0153	0.0192	1.0000	1.0000	0.0000	1.0006	1.0000	0.0006

5. Conclusion

Combining network dimensioning and QoS mechanisms is and appropriate way to implement QoS provisioning in the Ethernet-based BcN access network. We discuss the main components of network dimensioning and formulating problem. . Incorporating the QoS requirements, we formulate the problem as a nonlinear mixed-integer programming model. For solving optimization model, we develop tabu search with short term and longer term. The simulation studies validate that our tabu search of network dimensioning is sufficient QoS requirement restrict.

The implementation of Ethernet-based BcN access network is still progress. First we assume that number of user of single S-L2 switch is same wherever. However we decrease total cost by moving user from switch to another switches and removing empty switches. And we apply diversification strategy using random select during feasible solutions, but considering infeasible solutions can explore unvisited search space more than current method.

Reference

[1] S. Blake et al., “An Architecture for Differentiated Services,” IETF RFC 2475, 1998

[2] R. Braden et al., “Integrated Services in the Internet Architecture: an Overview,” IETF RFC 1633, 1994

[3] G. Carello, F. Della Croce, M. Ghirardi, and R. Tadei, “Solving the hub location problem in telecommunication network design: A local search approach,” Networks, 2004

[4] J. C. Crimi, “Next generation network services,” A Telcordia Technologies White Paper, 2003

[5] F. Glover, “Tabu search and adaptive memory programming—advances, applications and challenges,” Optimization and stochastic modeling technologies, 1997

[6] F. Glover, M. Laguna, “Tabu search,” Kluwer Academic, 1998

[7] ITU-R working group, “Methodology for the calculation of IMT-2000 terrestrial spectrum requirements,” Recommendation ITU-R M.1390, 2001

[8] ITU-R working group, “Spectrum requirements for international mobile telecommunications-2000 (IMT-2000),” Report ITU-R M.2023, 2001

[9] F. P. Kelly, “Blocking probabilities in large circuit-switching networks,” Advances in applied probability, 18:473-505, 1986

[10] F. P. Kelly, “Reversibility and stochastic networks,” John wiley and sons, New York, 1979

[11] H. K. Kim, “ICT Standardization Strategy in Korea,” Telecommunication Technology Association, 2005

[12] P. J. Kuhn, “The development of the next generation network,” Institute of Communication Networks and Computer Engineering, 2005

[13] A. Riedl, “Routing optimization and capacity assignment in multi-service IP networks,” Dissertation at technische universität münchen, 2004

[14] A. Riedl, Thomas Bauschert, Jochen Frings, “A framework for multi-service IP network planning,” 10th international telecommunication network strategy and planning symposium (Networks 2002), Munich, 2002

[15] Z. Rosberg, H. Le Vu, M. Zukerman, J. White, “Blocking probabilities of optical burst switching networks based on reduced load fixed point approximations,” Department of communication systems engineering ben gurion university, IEEE, 2003

[16] K. Tsang, K. W. Ross, “Algorithms to determine exact blocking probabilities for multirate tree networks,” IEEE transactions on communications, 1990

[17] Z. Zhao, Q. Liu, H. Guan, “A method and simulation study: Network dimensioning of

the broadband residential ethernet-based access network," Multimedia communication & network engineering research center, 2003