Network bandwidth reservation method combining machine learning and linear programming

Kouichi Genda
College of Engineering, Nihon University
1 Nakagawara, Tokusada, Tamura, Koriyama-shi, Fukushima 963-8642, Japan
genda.kouichi@nihon-u.ac.jp

Abstract: Network bandwidth reservation is an expected service under software-defined network environment, where users directly reserve network resources in an on-demand manner. To provide extensive bandwidth reservation services, instantaneous response to user requests, and high user request acceptance ratio are required. In this study, we propose a novel bandwidth reservation method to meet these two requirements by combining machine learning (ML) and linear programming (LP), for unpredictable bandwidth demands in which the use time is indicated strictly. In the proposed method, a user request is instantaneously judged through ML, and network resource allocation, including traffic routing, is optimally determined through LP. We demonstrate that the proposed method provides a suboptimal acceptance ratio with a difference of less than 1% compared to the optimal solution, and an instantaneous response of less than 0.1 ms under a general computation environment.

Keywords: bandwidth reservation, bandwidth calendaring, machine learning, linear programming, software defined network

Classification: Network

References

[1] S. Kandula, I. Menache, R. Schwartz, and S. R. Babbula, “Calendaring for wide area networks,” Proc. ACM SIGCOMM, pp. 515–526, 2014.

[2] L. Gkatzikis, S. Paris, I. Steiakogiannakis, S. Chouvardas, “Bandwidth calendaring: Dynamic services scheduling over Software Defined Networks,” Proc. IEEE International Conference on Communications (ICC), pp. 22–27, May 2016.

[3] Q. Wu and P. Dharam, “Advance bandwidth scheduling with minimal impact on immediate reservations in high-performance networks,” Proc. IEEE Network Operations and Management Symposium (NOMS), pp.679–682, 2012.

[4] S. Kamamura, R. Hayashi, H. Date, H. Yamamoto, T. Miyamura, Y. Uematsu and K. Genda, “Recommendation-based bandwidth calendaring for packet transport network,” IEICE Trans. Commun., vol. E100-B, no.01, pp.122–130,
1 Introduction

The bandwidth reservation service is one of representative services utilizing attractive characteristics of the software-defined networks (SDNs), where users can directly reserve network resources in an on-demand manner. Bandwidth reservation has attracted considerable research interest with respect to inter-datacenter traffic, in particular, for predictable users demands with delay tolerance such as over-the-top operators’ data transfer [1,2]. Here, we focus on bandwidth reservation for unpredictable bandwidth demands in which the use time is indicated strictly. This is because bandwidth reservation services for demands with the use time indicated strictly are indispensable under a mature on-demand bandwidth reservation service environment.

Fig. 1 illustrates our target. A user sends a request to a bandwidth reservation system; in the request, information such as the strict use time (cal), pair of source and destination locations (s, t), and the required bandwidth (bst) are indicated. In a bandwidth reservation system, a request is judged as accepted or rejected, and the result is sent back to the user. Information on the accepted requests is issued from the bandwidth reservation system to the network management system. This network management system establishes traffic routes with the required bandwidth for the accepted requests during their use time.

For providing bandwidth reservation services extensively, instantaneous responses to user requests (e.g., less than 1 s) is mandatory because users make reservations using an intuitive calendar interface similar to general hotel booking portal sites. In addition, the user request acceptance ratio should be maximized (e.g., more than 90%) not only for user demands’ satisfaction, but also for network utilization increase.

However, it is difficult to meet the two service requirements simultaneously. Ref. [3] proposes a mathematical scheduling algorithm to judge a reservation request without delay tolerance, where Dijkstra’s algorithm is applied to find a traffic route for a request. In [4], to achieve instantaneous response, network resources including the traffic route for user requests are determined using a method based on the K-shortest path (KSP) algorithm [5]. With the Dijkstra and KSP algorithms, as network resources are allocated preferentially to requests arriving earlier and are fixed permanently, the acceptance ratio of user requests is less than optimal. In contrast, an effective bandwidth allocation problem to
maximize the acceptance ratio of user demands can be solved using a method based on the linear programming (LP) approach (LP-based method). Whenever a new request arrives, network resources are re-optimized by solving a maximum flow problem with the objective function of maximizing the sum of the bandwidth of user requests that have been accepted in the use time of the new request, under the condition that multiple traffic routes between the source and destination nodes are allowed. However, in general, it may be difficult to achieve fast response. Consequently, with conventional methods, it is difficult to achieve our target. Therefore, a breakthrough is required to allocate network resources instantaneously and optimally for unpredictable bandwidth demands with strict use time.

Here, we focus on the attractive characteristics of machine learning (ML) technology, its prediction speed and the suboptimal accuracy. ML technology has been aggressively tackled in many application domains. However, effective application in the network domain has not yet been reported because it is difficult to characterize the appropriate features or the appropriate input/output patterns to reflect the highly dynamic and uncertain nature of network behavior [6,7]. In this study, we propose a novel bandwidth reservation method combining ML and LP technologies to satisfy the two service requirements simultaneously.

2 Proposed bandwidth reservation method

Fig. 2 depicts the time sequence of the proposed method. We define the unit of the use time as a timeslot (TS) (e.g., 1 h). Three phases are executed in a pipeline manner. In Request judgment, a new request is instantaneously judged as accepted or rejected through ML. The requests in each TS are received within the designated time window. For instance, requests that indicate the bandwidth use at TS #f are received within the time window TS #b to #d. The network resource allocation for each request is not determined at the request judgment time. The network resources for the accepted requests are collectively determined at the designated TS in Route allocation. Accepted requests with the same source and destination are bundled into a single request per logical path. The network resources for the logical paths are optimally determined. As shown in Route allocation, the accepted requests for TS #f are bundled into a request per logical path, and the network resource for the logical path is determined at TS #e. At TS #f,
logical paths with reserved bandwidths are established, and bandwidth reservation services are provided.

An ML classifier is used for request judgment. The classifier is designed using the supervised learning approach, where labeled datasets generated using the LP-based method are trained. The datasets include new request information \((cal, s, t, b^e)\), the network state at \(cal\), and the label of the request (accepted or rejected). We focus on the information of the logical network for judging the request. This is because we avoid using the concrete information on the physical network including network resource allocation at the judging time. Thus, in the network state, we apply the reserved bandwidth for all the logical paths.

The network resources for the accepted requests are optimally determined by solving an LP problem with the objective function of minimizing the maximum link utilization over the network, where the link capacity is designed for the bandwidth reservation service in advance. Note that the LP formulations are different from those of the LP-based method. In request judgment through ML, the bandwidth required for certain links may exceed the designed capacity due to misclassification. Thus, we adopt an objective function to minimize maximum excess of the designed capacity.

3 Performance evaluation

Based on the two service requirements, we evaluate the performance of the proposed method in terms of the request blocking ratio and request judgment time. The request blocking ratio is \((1 – \text{the request acceptance ratio})\). The proposed method is compared with two methods. One is the KSP-based method for the benchmark of the minimum request judgment time, in which a new request is simply judged based on the residual link capacity using the KSP algorithm [5]. The other is the LP-based method for the benchmark of the minimum request blocking ratio.

30 requests in a TS are supposed to be constantly received to simplify the discussion. Note that the number of requests in a TS is not restricted with the proposed method. The number of evaluated TSs and user requests are 1,000 and 30,000, respectively. The source and destination node pair of a request is randomly distributed. The request bandwidth follows an exponential distribution with an average bandwidth of 8 [arb. unit]. The holding time of a request is constantly one TS. The problems formulated through LP are solved using the GNU Linear
The ML classifier is developed using the scikit-learn library.

For a first step toward the challenge of judging requests through ML, as the ML classifier model, we adopt the linear support vector machine, which is one of representative supervised learning models. By training 500,000 labeled datasets, the receiver operating characteristic’s area under the curve score, called ROC AUC, of the ML classifier adopted is approximately 0.9. The score is enough large to evaluate the effectiveness of the proposed method.

Fig. 3(a) shows network models evaluated. The link capacity designed for the bandwidth reservation service is 20 [arb. unit].

Fig. 3(b) depicts the request blocking ratio defined in Eq. (1), which is the average cumulative blocking ratio during a TS in 1,000 TSs. The value of $n_j(i)$ is 1, when the $i$-th request is accepted in TS $j$. $N \ (N=1,000)$ is the number of evaluated TSs. $k \ (1 \leq k \leq 30)$ is the request order in a TS. The results of the proposed method denote the request blocking ratio for the requests accepted except for fault positive requests.

\[
P_b(k) = 1 - \frac{\sum_{j=1}^{N} \sum_{i=1}^{k} n_j(i)}{N \cdot k},
\]  

Fig. 3 Performance evaluation results.
In both network models, the difference between the LP-based method and the proposed method is less than 1%. The difference is attributed to the misclassification of the ML classifier. In contrast, the difference between the LP-based method and the KSP-based method is considerable because a single route is allocated for each logical path in the KSP-based method. The results indicate that the ML classifier provides suboptimal solutions for judging user requests.

Although we omitted the detailed discussion of network resource allocation for accepted requests, the bandwidth required of some links exceed the designed capacity due to ML misclassification, even with the proposed method discussed in Section 2. When the network resource margin is not sufficiently large, misclassification should be reduced to avoid degrading the quality of the other network services. We intend to study this in future.

Fig. 3(c) shows the request judging time under a general Core i9 with 32-GB memory environment. The judging time of the $k$-th request is the average of the time for judging the $k$-th request in 1,000 TSs. The results indicate that, in both network models, the time of the proposed method is less than 0.1 ms, and is sufficiently less to meet the service requirements.

4 Conclusion

A novel ML and LP combined bandwidth reservation method was proposed to evolve the network bandwidth reservation service. Simulation results indicate that the proposed method provides the service requirements; suboptimal acceptance ratio and instantaneous request judgment. In future, to enhance our method for practical application, we intend to minimize misclassification.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number JP20K11798.