Autonomic Resource Allocation for Video Streaming Services in Content Delivery Networks

[1]Sungsu Kim, [1]Sin-seok Seo, [2]Joon-Myung Kang, [3, 4]Guy Pujolle, and [1, 3]James Won-Ki Hong
[1]Department of Computer Science and Engineering, POSTECH, {kiss, sesise, jwkhong}@postech.ac.kr
[2]Department of Electrical and Computer Engineering, University of Toronto, joonmyung.kang@utoronto.ca
[3]Division of IT Convergence and Engineering, POSTECH
[4]LIP6 / UPMC - University of Paris 6, guy.pujolle@lip6.fr

Abstract—**Video streaming is currently one of the most popular online services. In order to improve video the Quality of Service (QoS) of video streaming, Content Delivery Networks (CDNs) have been widely adopted. CDNs replicate their content on distributed servers, meaning that a user can access content from the nearest CDN server. In order to provide better QoS for video streaming, it is necessary to understand the current state of network resources and efficiently allocate them to a service user. In this paper, we capture the current state of a service by observing low-level monitoring data using ontological reasoning. By taking into account both latency and bandwidth utilization, we propose bandwidth allocation and request routing algorithms to guarantee Service Level Agreement.**

Keywords—Autonomic Network Management, Resource Allocation

I. Introduction

Due to the growth of the Internet, multimedia networking applications, such as Video on Demand (VoD) and video streaming services, have become very popular. Compared to web and file transfer services, the provision of a guaranteed Quality of Service (QoS) is more important for multimedia services because of their delay sensitive nature [1]. Many service providers distribute servers and content to multiple locations, known as a Content Delivery Network (CDN), in order to deliver this content more efficiently. Clients who subscribe to a particular service connect to the nearest server to obtain their desired content. This guarantees a better QoS than the traditional client–server model whereby all the clients access a single server providing the service.

Fig. 1 illustrates a video streaming service provided via a CDN. Streaming servers are distributed to multiple locations on the IP network. Content is also distributed to the servers, and content providers use registered servers for their services. This is a real service model that is soon to be commercially launched by Korea Telecom (KT). For example, Content Provider 1 (CP1) uses servers 1 and 6 to provide its content. Because the servers are distributed all around the country, local content providers do not need servers far from their target service area.

This paper presents a method for bandwidth allocation and request routing based on service priorities. In [2], the authors proposed a bandwidth allocation algorithm based on game theory. However, their algorithm does not consider priorities between service classes. Guaranteeing Service Level Agreement (SLA) to users is of utmost importance from a service provider's point of view. We define a utility function for allocating resources to provide better service quality and balance the load between servers. We use cognitive approaches to understand the current state of the network and video streaming service [3]. Ontological reasoning enables us to determine the urgency of the current state using defined rules and relationships between managed elements.

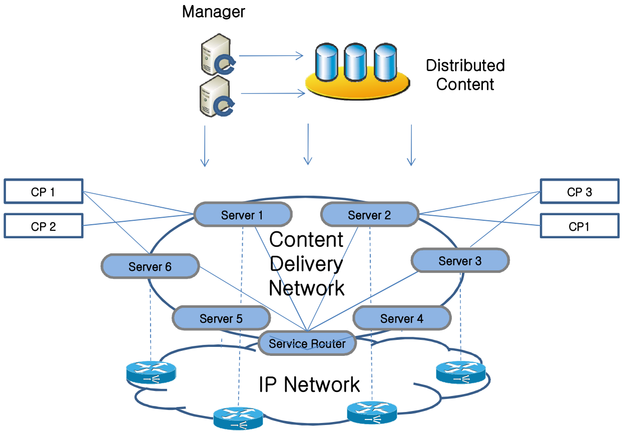

Figure 1. Example of a CDN-based video streaming service architecture

First, we build an ontological model of video streaming service elements in order to understand the current state of service based on observations of low-level monitoring data. Second, we apply the proposed bandwidth allocation algorithm based on the determination of the current state. This guarantees the quality of high priority services under high server loads by allocating bandwidth based on the priority of each service. Finally, the proposed request routing algorithm assigns a request to a server to provide low latency to clients and balance the load between servers.

The rest of this paper is organized as follow. In section II, we present autonomic control loops, bandwidth allocation, and load balancing approaches. Section III describes how we apply cognitive control loops to recognize the current state and automatically commit appropriate actions. Section IV presents the proposed algorithms for bandwidth allocation and server selection considering the service class priorities. In section V, the simulation results of the proposed approaches are analyzed, and, finally, conclusions are drawn in section VI.

II. Related Work

In this paper, we apply autonomic control loops to understand the current state of the service and commit appropriate actions to maintain service in a desired state. Autonomic control loops are the core idea behind autonomic network management. IBM defined the Monitor–Analyze–Plan–Execute (MAPE) control loop in [4]. Sensors and Effectors obtain data from, and provide commands to, both the entity being managed and other Autonomic Managers. Foundation–Observe–Compare–Act–Learn–rEason (FOCALE) control loops have also been applied [5]. Another similar solution has been developed by Ginkgo Networks in [6]. The detailed processes of MAPE, Ginkgo and FOCALE are different, but their fundamental methodology is the same. Collecting data from managed resources and analyzing it to understand the current state of the resources. If the system is not in the desired state, these systems reconfigure the target resources to maintain the system in the desired state.

In order to provide a guaranteed level of service to clients, efficient resource allocation is important, and providing appropriate bandwidth for media streaming services is even more important due to its delay-sensitive nature. Chakareski et al. [7] proposed a bandwidth allocation algorithm to optimize the rate-distortion performance for media streaming applications. Their algorithm assumed that clients change encoding rates based on processing capacity and network bandwidth. For multi-user multimedia rate allocation, a game theory-based algorithm was proposed in [2]. The algorithm focuses on bandwidth allocation with fairness between clients. However, these bandwidth allocation methods do not take into account the priorities among different services [8-10].

III. A Cognitive Approach for Video Streaming Services

In order to provide an efficient video streaming service, we need to recognize the current state of resources and trigger appropriate actions immediately. We envision a cognitive system that can reason which actions should be taken and learn from experience to improve its performance [3].

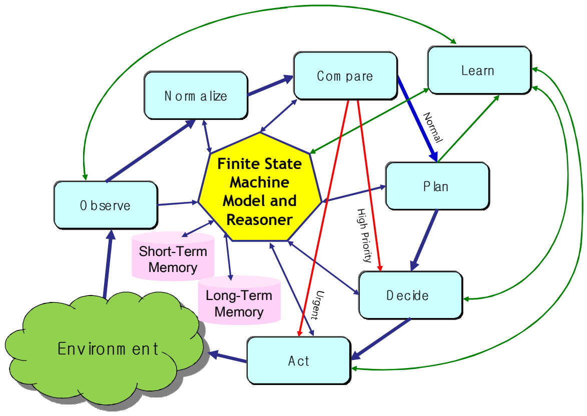

Figure 2. Cognitive control loops

Fig. 2 illustrates cognitive control loops. As the processes are using the finite state machine and reasoner, it can be determined whether the current status is normal or abnormal, and which actions should be taken to correct abnormal states. Based on observations of low-level monitoring information, the system reasons which services and users are affected. Normally, the deliberative process is used to infer the current state and implement actions, which takes an Observe–Normalize–Compare–Plan–Decide–Act path. A reactive process is used when the system determines that the current state is urgent and there is a predefined action for the case. This takes an Observe–Normalize–Compare–Act path. The reactive process enables much of the computationally intensive portions of the control loop to be bypassed. Therefore, problems can be solved faster with the reactive process.

In this paper, we consider a video streaming service that has three different service classes: Gold, Silver, and Bronze. First, the highest priority Gold service, providing HD streaming videos and 8 Mbps of bandwidth, must be guaranteed 99.9% of the total service time, as specified in the SLA of Table 1. Silver and Bronze services provide 5 Mbps and 2 Mbps bandwidths, respectively. However, users of the Silver and Bronze services have a lower priority than Gold service users. Under a heavy load, Silver and Bronze services may not provide enough bandwidth for video streaming.

Table 1. Example of SLA Specification

Service Class	Resolution	Credit
Gold	1920 × 1080 (8 Mbps)	Guarantee bitrate 99.9% of service time
Silver	1280 × 720 (5 Mbps)	N/A
Bronze	720 × 480 (2 Mbps)	N/A

In the Observe phase of the control loop (Fig. 2), monitoring data including the volume of traffic sent, bandwidth utilization, and the number of users for each service are retrieved continuously from video streaming servers. The observed data are then normalized to designated units in the Normalize phase. In the Compare phase, the current state is analyzed and compared to the desired state. If the current state is determined to be an urgent case, predefined actions are committed reactively without entering the Plan and Decide phases.

To determine the current state during the Compare phase, we use ontology due to its significant role in the harmonizing of information models and semantic representation. As shown in Fig. 3, we build an ontology model to represent the relationships between network elements and services. Node, Service, and Equipment are subclasses of Network Element, and Service uses a Node to provide the service.

Based on the ontology model, we write rules to capture the state of the service. If the bandwidth utilization of a managed server is less than 60%, we define that it is in a normal state. If the bandwidth utilization of a managed server is greater than or equal to 60% and less than 80%, it is in a high priority state, which requires administrators to take action manually. If the bandwidth utilization of a managed server is greater than 80%, it is in an urgent state. Allowed bandwidths for users should be configured based on the bandwidth allocation strategy.

The rule ‘SLAGeHealthy’ determines whether the network is healthy when the greater than or equal to CompareFunction is defined in the Service Level Specification (SLS) and the value of the monitored performance information is greater than or equal to the threshold of the SLS.

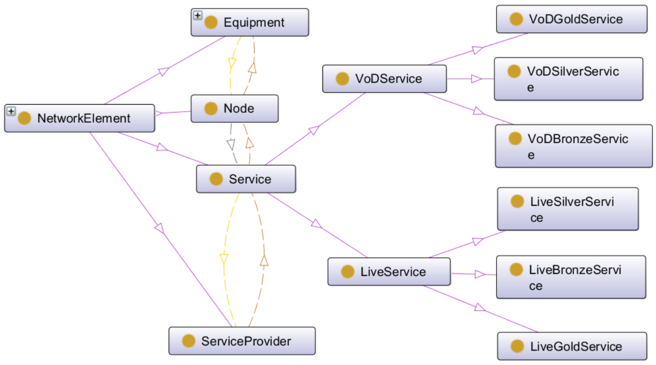

Figure 3. Ontology for network elements and services

IV. Resources Allocation for Video Streaming Service

In this section, we present a bandwidth allocation and server load balancing algorithm for video streaming service. The main objective of the proposed algorithms is allocating network resources to provide a service based on the priority of services.

A. *Bandwidth Allocation*

We allocate an explicit bandwidth to a single user based on the class of service he or she uses. The objective is providing the guaranteed bandwidth to a Gold service user in a high load. It means that the bandwidth specified in the SLA should be provided to Gold service users even if Silver or Bronze service users get service with a lower bandwidth. The available bandwidth of each server is bw_{server}, so a total amount of allocated bandwidth should not exceed bw_{server}. As specified in the SLA, fixed bandwidth bw_G^{fixed} should be allocated to a Gold service user *i*. bw_S^j is allocated to a Silver service user *j* and bw_B^k is allocated to Bronze service user *k*. We use bandwidth allocation policy that allocates the same amount of bandwidth to every user subscribing a same class of service.

If the bandwidth utilization of a video streaming server is not high, Silver and Bronze service users get bandwidth rate bw_S and bw_B, respectively. However, if the total requested bandwidth of Gold, Silver, and Bronze services exceeds bw_{server}, the bandwidth for Silver and Bronze services are degraded.

B. *Server Selection*

Let us suppose *n* servers provide video streaming services and requests from the clients are distributed to the servers. Each contents provider assigns servers to provide their contents. For example, a provider A assigns server 1, 2, and 3 for the service a1. Clients who request service a1 can watch video content of service a1 via servers 1, 2, or 3. The proposed server selection algorithm allocates requests based on the observation of monitoring information. The goal of the algorithm is allocating the request to a server that has the lower bandwidth utilization and latency. We used utility function to select a server which provides fast response time and has low bandwidth utilization. When a new request arrives, latency between the client who sent request and the bandwidth utilization of servers are examined. The request is allocated to a server with the highest utility value. The request is allocated to a server with the highest utility value. Therefore, the utility function for server selection can be defined as:

$$U(u_i, r) = \alpha \cdot f(u_i) + g(l)$$

A high value of $U(u_i, r)$ means that the server *i* is highly appropriate for an incoming request. In terms of bandwidth utilization u_i, $f(u_i)$ indicates a decreasing utility as the bandwidth utilization increases. Utility decreases sharply as u_i exceeds 0.5. $g(l_i)$ is utility determined by latency of a server , we use Round Trip Time (RTT) from a server *i* to the client who sent a request. However, various matrices can be used to represent latency. The decrease of utility is sharp as the RTT approaches 2 second because we assume that RTT less than 2 second is acceptable.

$$f(u_i) = 1 - \frac{e^{\beta_1 u_i}}{k}$$

$$g(l_i) = \frac{e^{-l_i + \beta_2}}{1 + e^{-l_i + \beta_2}}$$

α is a parameter that controls the balance of the utility function between the bandwidth and latency. The proposed utility function is used to assign a server to a request. This parameter can be tuned adaptively using a learning component of the cognitive control loops. Different requirements may need a different α value. For example, α is set higher if latency is more important than load balancing.

V. Evaluation

In order to show the effectiveness of the proposed approaches, a simulation of bandwidth allocation and request routing was performed. We synthetically generated a number of requests for Gold, Silver, and Bronze services and show allocated bandwidths to users and bandwidth utilization of each server. The proposed algorithms were implemented in Java and the simulation was run on a Windows machine with Intel Pentium Dual CPU E2140 running at 1.6 GHz and 2 GB RAM. First, we demonstrate bandwidth allocation for the video streaming service. Each server provides video streaming services, which are classified as Gold, Silver, and Bronze service. As the number of users is increasing, a current state is determined as urgent and the proposed bandwidth allocation algorithm is applied. Fig. 4 shows the number of users for each service class. In our simulation scenario, the number of users for Gold service class is increasing gradually and the number of users for Silver and Bronze service class is stable.

Fig. 5 shows the bandwidth utilization of each class. The bandwidth utilization of each service class is proportional to the number of users. From 10 seconds, the total bandwidth utilization exceeds 80% and the state is determined as an urgent state and there is service degradation from 20 seconds due to the exceeding of available bandwidth.

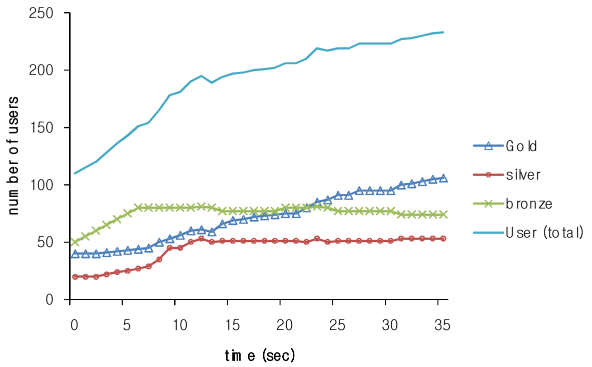

Figure 4. Number of users of Gold, Silver, and Bronze Services

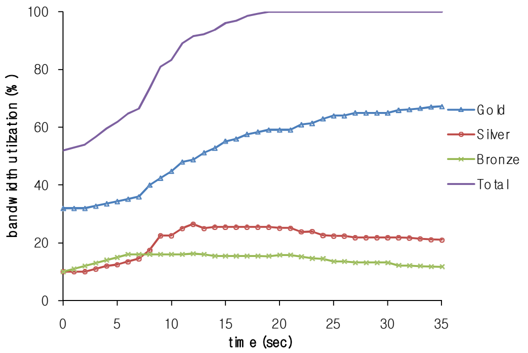

Figure 5. Bandwidth utilization of Gold, Silver, and Bronze Services

Fig. 6 shows the bandwidth allocated to each user. Bandwidths which are enough to provide a service specified in the SLA are allocated to users till 20 second. As the number of users is increasing, the bandwidth allocated for each user is decreasing. Especially, Gold service users cannot use video streaming service with the bandwidth as specified in the SLA. The bandwidth allocated to gold class service user degraded to 6.3Mbps. Fig. 6 also shows the bandwidth allocated to users with the proposed allocation algorithm. 8Mbps of bandwidth is allocated to each Gold service user consistently. However, the bandwidths for Silver and Bronze class services are getting less comparing the case that no action for bandwidth management is applied.

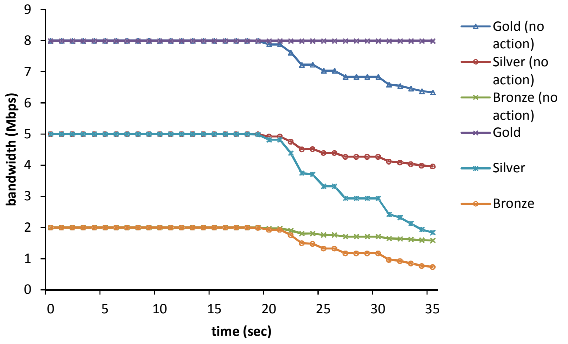

Figure 6. Bandwidth allocated to a single user of Gold, Silver and Bronze Services

VI. CONCLUSIONS

In this paper, we have proposed bandwidth allocation and request routing algorithms for load balancing to guarantee quality of video streaming service on CDN. We have designed an ontological model to capture a current state. Based on the determined state, we have applied our algorithms for bandwidth allocation and request routing. We have defined a utility function for balancing latency and load when a server is assigned to a request. In the experiment, we have showed that our algorithm has enabled network resources to be used more efficiently with a reasonable latency overhead.

For future work, we consider advanced matrices to allocate network resources more efficiently. In addition, we will validate the proposed method with real enterprise network traces.

ACKNOWLEDGEMENTS

This research was supported by World Class University program funded by the Ministry of Education, Science and Technology through the National Research Foundation of Korea (R31-10100)

REFERENCES

[1] A. Katsaggelos, Y. Eisenberg, F. Zhai, R. Berry, and T. Pappas, “Advances in Efficient Resource Allocation for Packet-Based Real-Time Video Transmission,” in *Proceedings of the IEEE*, vol. 93, no. 1, Jan. 2005, pp. 135–147.

[2] Y. Chen, B. Wang, and K. Liu, “A Game-theoretic Framework for Multi-User Multimedia Rate Allocation,” in *Proceedings of the 34th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2009)*, pp. 1997-2000, Apr. 2009.

[3] J. Strassner, J. W. K. Hong, and S. van der Meer, “The Design of an Autonomic Element for Managing Emerging Networks and Services,” in *Proceedings of International Congress on Ultra Modern Telecommunications (ICUMT 2009)*, Oct.12-14, 2009, pp. 1-8.

[4] IBM, “An Architectural Blueprint for Autonomic Computing, v7”, http://www-03.ibm.com/autonomic/pdfs/AC%20Blueprint%20White%20Paper%20V7.pdf. [Online Available: April 20, 2012]

[5] J. Strassner, N. Agoulmine, and E. Lehtihet, “FOCALE – A Novel Autonomic Networking Architecture”, *ITSSA Journal*, vol. 3, no. 1, May 2007, pp 64-79.

[6] [128]I. Fajjari, O. Braham, M. Ayari, G. Pujolle, H. Zimmermann - AAVP: An Innovative Autonomic Architecture for Virtual network Piloting, IJNGC 2(3), March 2011.

[7] J. Chakareski and B. Girod, “Computing Rate-Distortion Optimized Policies for Streaming Media with Rich Acknowledgements,” in *Proceedings of Data Compression Conference (DCC 2004)*, Mar. 2004, pp. 202-211.

[8] Y. Yan, A. El-Atawy, and E. Al-Shaer, “A Game-Theoritic Model for Capacity-Constrained Fair Bandwidth Allocation,” *International Journal of Network Management*, vol. 18, no. 6, Nov. 2008, pp. 485-504.

[9] Y. Yan, A. El-Atawy, and E. Al-Shaer, “Fair Bandwidth Allocation under User Capacity Constraints,” in *Proceedings of the 10th IEEE/IFIP Network Operations and Management Symposium (NOMS 2006),* Apr. 2006, pp. 138-149.

[10] K. Ivesic, M. Matijasevic, and L. Skorin-Kapov, “Simulation Based Evaluation of Dynamic Resource Allocation for Adaptive Multimedia Services,” in *Proceedings of the 7th International Conference on Network and Service Management (CNSM 2011)*, Oct. 2011, pp. 1-4.