



End-Link Collaboration Control Mechanism in Intelligent Networks Based on Traffic Classification

Songer Sun^(✉), Li Zhang, Xiaoping Han, and Chengjie Gu

Security Product Line, H3C Technologies Co., Ltd.,
Innovation Industrial Park Phase II,
Gaoxin District, Hefei, Anhui, People's Republic of China
{sunsonger, billchang}@h3c.com

Abstract. In order to meet special requirements of service transportation and improve integral network performance, we propose self-adaptive control mechanism in intelligent networks based on traffic classification. The self-adaptive control mechanism which considers end-link cooperation in intelligent network implements service transportation with control method of the distributed-centralized combination. The mechanism optimize relevant network elements devices and reasonably allocates limited resources to improve end-to-end target depending on given strategies.

Keywords: End-Link · Collaboration control mechanism ·
Intelligent networks · Traffic classification

1 Introduction

Intelligent network is considered a new way of improving entire network performance and end-to-end system performance as well as simplifying network management [1]. It is the trend of next-generation communication. Intelligent networks are important for ensuring performance in complex and isomerism networks. Self-adaptive control technology can plan and allocate limited network bandwidth effectively so that network performance is improved [2]. The technology also manages and controls network traffic according to service features in order to improve the revenue of unit bandwidth. Therefore, intensive self-adaptive control is essential to solving network QoS problems in next-generation communication.

The foundation of traffic classification is the rapid perception of network environment. Situation awareness needs to observe current network environment information in appropriate time [3]. The information is used in later planning and decision making to determine whether the current network meets user requirements [4]. If not, a suitable reconfiguration method is used to meet user requirements.

The key concept is the network's ability to perceive changes in the intelligent network environment and adjust itself in real time. We propose the self-adaptive control mechanism in intelligent networks based on traffic classification. The remainder of this paper is organized as follows. In Sect. 2, we discuss the traffic classification

using traffic classification based machine learning method. Section 3 gives End-Link collaboration control mechanism in intelligent networks. Finally, we conclude our paper and discuss future work in Sect. 4.

2 Traffic Classification Using Traffic Classification Based on Sparse Proximal SVM

2.1 Intelligent Networks Need Traffic Classification

Intelligent networks need to be self-aware in order to provide resilient applications and services. Such networks should exhibit cognitive properties where actions are based on reasoning, autonomic operations, adaptive functionality and self-manageability [5]. Therefore, the best way to perceive, analyze, determine, and control transmission service initiatives is by using traffic classification (SA) technology. The network system initiatively perceives services on the network, including end user service status and NE (Network element) service status.

Initiative perception and classification based on the service stream is the foundation of service-centered resource configuration, route adjustment, and dynamic self-adaptive traffic control [6]. In service-aware technology, before the SA is introduced, traditional static port method, payload feature method, and stream statistical feature method are used. The methods are effective for perceiving regular services, but they cannot perceive many new services respectively [7].

Environment information perceived by the traffic classification includes network type, network topology, available resources, interface protocols, and network traffic, all of which affect end-to-end transmission performance [8]. Context perception is an important way of improving network intelligence. It determines changes in context information and adjusts itself accordingly. When the network environment changes dynamically, the network makes relevant self-adjustments. This self-adjustment uses a reflection mechanism and a policy mechanism. From the policy definition, the network can pre-define an adjustment method when the context changes.

2.2 Proximal Support Vector Machine

Support vector machine (SVM) based on the statistical learning theory by Vapnik is a new and powerful classification technique and has drawn much attention in recent years [9]. The optimal classifier can be obtained by solving a quadratic programming (QP) problem.

Support Vector Machine (SVM) is known as one of the best machine learning algorithms for classification purpose and has been successfully applied to many classification problems such as image recognition, text categorization, medical diagnosis, remote sensing, and motion classification [10]. SVM method is selected as our classification algorithm due to its ability for simultaneously minimizing the empirical classification error and maximizing the geometric margin classification space [11]. These properties reduce the structural risk of over-learning with limited samples. But standard SVM are still some limitations [12]. Proximal support vector machine

(PSVM) is proposed instead of SVM, which leads to an extremely fast and simple algorithm for generating a system of linear equations. The formulation of PSVM greatly simplifies the problem with considerably faster computational time than SVM.

Assume that a training set S is given as $S = \{(x_1, y_1), \dots, (x_l, y_l)\}$, where $x_i \in R^n$, and $y_i \in \{-1, +1\}$. The goal of SVM is to find an optimal separating hyperplane

$$w'x - b = 0 \tag{1}$$

that classifies training samples correctly or basically correctly, where $w \in R^n$, and the scalar $b \in R^n$. Now to find the optimal separating hyperplane is to solve the following constrained optimization problem

$$\min \frac{1}{2} w'w + ce' \zeta \tag{2}$$

$$s.t. \quad D(Aw - eb) + \zeta \geq e \tag{3}$$

$$\zeta \geq 0 \tag{4}$$

Where $\zeta = (\zeta_1, \zeta_2, \dots, \zeta_l)'$, ζ_i is slack variable, $i = 1, 2, \dots, l$, e denotes a column vector of ones of arbitrary dimension, $A = (x_1, x_2, \dots, x_l)'$, D is a diagonal matrix whose entries are given by $D_{ii} = y_i$, and $C > 0$ is a fixed penalty parameter of labeled samples. It controls the tradeoff between complexity of the machine and generalization capacity.

PSVM modifies SVM formulation based on maximizing the separating margin $1/w'w + b^2$ in the space of R^{n+1} and changes the slack variables ζ from the L_1 norm to the L_2 one. Note that the nonnegative constraint on the slack variables ζ in (3) is no longer needed. Furthermore, a fundamental change is replacing the inequality constraint with an equality constraint. This leads to the optimization problem as follows

$$\min \frac{1}{2} (w'w + b^2) + c \frac{1}{2} \|\zeta\|^2 \tag{5}$$

$$s.t. \quad D(Aw - eb) + \zeta = e \tag{6}$$

This modification not only adds advantages such as strong convexity of the objective function, but changes the nature of optimization problem significantly. The planes $w'x - b = \pm 1$ are not bounding planes any more, but can be thought of as “proximal” planes, around which points of the corresponding class are clustered. The formulation of PSVM greatly simplifies the problem and generates a classifier by merely solving a single system of linear equations. However, sometimes the result of PSVM is not accurate when the training set is inadequate or there is a significant deviation between the training and working sets of the total distribution.

In this section, we principally test on different datasets consists of public Moore_Set, Handmade_Set, and Univetsity_Set. For each dataset, we give the 10-fold average testing correctness. In this experiment we compared the performance using SVM methods for classification.

Table 1. Comparing algorithm performance

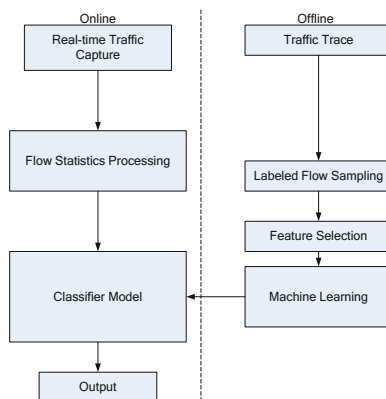
Dataset	PSVM	SVM
	Accuracy(%) Time(Sec)	Accuracy(%) Time(Sec)
Moore_Set	91.66 3.6	91.92 63.7
Handmade_Set	92.73 0.7	92.94 17.3
Univetsity_Set	90.58 75.3	90.69 1668.4

As shown in Table 1, bold type indicates the best result, the accuracy of the four algorithm were very similar but the execution time including ten-fold cross validation for PSVM was smaller by as much as one order of magnitude or more than the other methods tested.

2.3 Real-Time Traffic Classification Framework

Network traffic classification schemes operate on the notion of network flows. A flow is defined to be as a series of packet exchanges between two hosts, identifiable by the 5-tuple (source address, source port, destination address, destination port, transport protocol), with flow termination determined by an assumed timeout or by distinct flow termination semantics. For each flow, network monitors can record statistics such as duration, bytes transferred, mean packet inter arrival time, and mean packet size.

Classifiers based on machine learning use a training dataset that consists of N tuples (x_i, y_i) and learn a mapping $f(x) \rightarrow y$. In the traffic classification context, examples of attributes include flow statistics such as duration and total number of packets. The terms attributes and features are used interchangeably in the machine learning literature. In our supervised Network traffic classification system, Let $X = \{x_1, x_2, \dots, x_n\}$

**Fig. 1.** Architecture of real-time network traffic classification based on proximal SVM

be a set of flows. A flow instance x_i is characterized by a vector of attribute values, $x_i = \{x_{ij} | 1 \leq j \leq m\}$, where m is the number of attributes, and x_{ij} is the value of the j^{th} attribute of the i^{th} flow, and x_i is referred to as a feature vector. Also, let $Y = \{y_1, y_2, \dots, y_q\}$ be the set of traffic classes, where q is the number of classes of interest. The y_i can be classes such as “HTTP”, “FTP”, and “P2P”. Therefore, our goal is to learn a mapping from an m -dimensional variable X to Y .

As illustrated in Fig. 1, the architecture of online network traffic classification based on proximal SVM includes two portions: off-line ML modeling and on-line ML classification. The system carries out feature selection to eliminate the redundant and irrelevant features to get optimal feature subset. In the stage of online ML classification, the flows which can be real-time collected are trained using the selected ML classifier. Flow statistics is computed in terms of each selected feature and stored them to the corresponding database in the flow preprocessing module. The output classifier model produced by off-line ML modeling is applied to classify the captured traffic. Eventually, the classification output would be applied to network activities such as network surveillance, QoS.

3 End-Link Collaboration Control Mechanism in Intelligent Networks

3.1 Intelligent Network Components

The Internet has become a large non-linear complex giant system in the past years. Early network management systems were limited to simple command line interfaces (CLI) provided to network administrators who were responsible for querying and manually configure every network component. The result is that the management of current and emerging network technologies is becoming the main bottleneck to any further advance [13].

If the network is not meeting the end-to-end requirements of users, the intelligent network adjusts the protocol stack parameters of the relevant NE to meet these requirements. The adjustment process is the reconfiguration of the network [14]. Intelligent network emphasizes the end-to-end target, and it should provide end-to-end re-configurability. Software radio technology is limited to reconfiguring the terminal, but intelligent network involves all layers of the NEs and protocol standards that a stream passes through. It is a scheme with foresight that ensures QoS targets are met. More factors are considered in end-to-end reconfiguration.

Realization of intelligent network is based on reconfiguration of the NE. The reconfiguration process can also be implemented through software, but the technical level of this reconfiguration is higher. Terminal reconfiguration, network reconfiguration, and service reconfiguration are contained, and this configuration is not limited to a single node. Multiple NEs on the end-to-end path are covered. This is called end-to-end reconfiguration (E2R). The complexity and importance of E2R is greater than terminal reconfiguration.

According to the existing literature, intelligent network should have at least the following five main functions. Self-aware should be able to know what is happening inside; then according to what it knows to determine appropriate actions to achieve goals and to learn. Self-configuration can adapt immediately—and with minimal human intervention to configure itself in the dynamically network environment [15].

3.2 Control Architecture in Intelligent Networks

As shown in Fig. 2, the network decision-making and control architecture has a three-level structure, composed of NE cognitive module, autonomous domain cognitive server, and central cognitive server. Each part provides cognitive capability (self-awareness, self-learning, and self-decision making). The NE cognitive module is the basic unit of the intelligent network awareness, analysis, and control system. It provides awareness and decision-making capability and dynamically adjusts NE parameters or configuration. The autonomous domain cognitive server deployed with cognitive module form a cognitive autonomous domain that is responsible for managing and controlling the NE device, service traffic, and network resources.

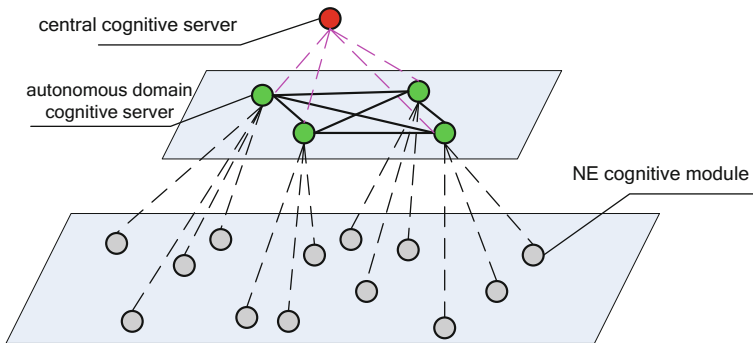


Fig. 2. The network decision-making and control architecture

At the same time, a central cognitive sever configured in the architecture is responsible for monitoring, awareness, and management of the running status of the entire network. The layered structure reduces the load on the central cognitive server. Even if the server fails temporarily, service QoS guarantee and management throughout the entire network is not affected.

Distributed networking and communication is enabled between autonomous cognitive servers so that information is exchanged in real time. The reason for using distributed management in the domain cognitive servers is to increase system reliability, flexibility, and expansibility. In the autonomous domain, adjacent nodes communicate so that distributed cooperative monitoring and self-adaptive processing is possible [16]. The architecture integrates the features of centralized architecture and distributed processing technology.

3.3 Self-adaptive Control Method Based on End-Link Collaboration

Network service types are varied, and network environments are complex and dynamic. Traditional end-to-end insurance technology lacks intelligent inference and self-learning capabilities. Therefore, it cannot adapt to provide ideal service under dynamically changing network conditions. Intelligent networks was different from the existing network of one of the most important characteristic is achieve end to end goals and its intelligence, autonomic, self-awareness and self-adaptive learning.

Intelligent network end-to-end targets is guaranteed by cognitive NEs. Cognitive NEs are cooperative or independent. The NEs perceive the network condition in real time, bring the trends together, and analyze the network condition. They configure themselves based on existing policies for achieving end-to-end targets.

The following describes the integration of service source-end control and link control in the intelligent network based on traffic classification, resource appointment concept, and control theory. A collaborative port and path policy-based self-adaptive control mechanism is proposed to solve the problem of end-to-end QoS guarantee for service traffic. The mechanism sends real-time network parameters to the autonomous domain server (or central cognitive server) through a feedback control. As a result, the self-adaptive control mode is integrated into the terminal NE and routers. The history of the network condition is compared with the current condition to form a control policy and to update the policy library through self-learning. At this point, the control policy is optimal. The mechanism can ensure the normal operation of a single NE and demonstrate the features of intelligent network. The mechanism uses relevant NE devices and reasonably allocates limited resources to improve end-to-end QoE and QoS. In this way, the performance of the entire network is optimized. Figure 3 shows the awareness-based service source end control layer and distributed awareness-based link control layer.

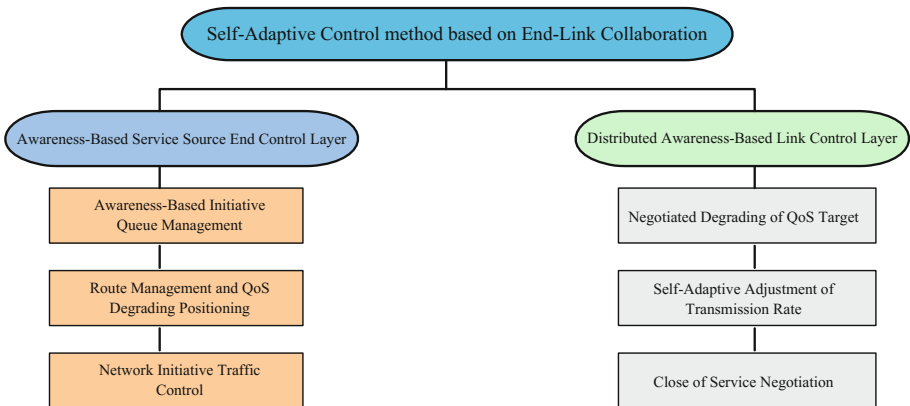


Fig. 3. Self-adaptive Control method based on End-Link collaboration

Awareness-based service control at the source end is implemented through self-adjustment of the source-end transmission rate, initiative closing of service, and initiative decrease of target. When the service source end launches a service in the traditional network, the current network condition is not considered. The intelligent network service end has certain cognitive functions; therefore, the cognitive information comes from the domain server or central control server. When certain conditions are met—for example, when bandwidth is sufficient—resources in the network can accept the access of other service traffic, and the service traffic can be transmitted to the peer end. When high-priority users need to transmit services, but the current network does not provide sufficient resources as required by the SLA, the central cognitive server (or domain cognitive server) negotiates with users at the service source end. If the user accepts a reduction of QoS, the source end transmits the service traffic according to the negotiated results. If QoS requirements cannot be reduced, the cognitive server recycles the network resources being used according to the resource distribution policy or even forcibly closes certain low-priority services.

Link control based on distributed awareness is implemented through initiative control of NE traffic, route management, QoS degradation positioning, and initiative queue management. By perceiving the network and making decisions based on this information, switches or routers with cognitive functions in the network can initiatively control traffic of different services and ensure the volume of trusted service traffic and key service traffic. They can also limit the volume of unsafe traffic or non-key traffic.

As service requirements and network resources are changing in real time, bottlenecks or QoS-degrading parts of the end-to-end network can be detected by cognitive route management and QoS degradation positioning. In addition, analysis and decision making can be performed, and service traffic can be re-routed. An intelligent and initiative queue management algorithm can also be used to determine congestion in the intelligent network. Awareness-based initiative queue management is oriented to the server's collaborative drive policy. This policy is integrated into the initiative queue management method to improve the resource appointment algorithm and router buffer management mode. Resources of the router or end system can then be reserved.

Network state can be evaluated and corrective actions can be initiated without disrupting system operation by self-healing. The network environment as a whole becomes more resilient as changes are made to reduce or help to eliminate the impact of failing components. Self-optimizing refers to the ability of the network environment to efficiently maximize resource allocation and utilization to achieve end-to-end goals with a minimal human intervention. Self-protecting can detect hostile or intrusive behavior as it occurs and take autonomous actions to make itself less vulnerable to unauthorized access and use, viruses, denial-of-service attacks and general failures.

4 Conclusion

Intelligent networks need to be self-aware in order to provide resilient applications and services. This paper proposes self-adaptive control mechanism in intelligent networks based on traffic classification. Drawing on control theory, network traffic can be controlled with a self-adaptive control mechanism that has end-link collaboration in

intelligent networks. Such networks should exhibit cognitive properties where actions are based on reasoning, autonomic operations, adaptive functionality and self-manageability. Through self-learning mechanism and adaptive mechanisms of the network, it achieves the end-end goals. In this architecture, network flow can be identified and classified by intelligent situation-aware model. This mechanism can adjust resource allocation, adapt to a changeable network environment.

References

1. Kanaumi, Y., Saito, S., Kawai, E.: RISE: a wide-area hybrid OpenFlow network testbed. *IEICE Trans. Commun.* **96**(1), 108–118 (2013)
2. Bari, M., Chowdhury, S., Ahmed, R.: On orchestrating virtual network functions. In: *International Conference on Network and Service Management*, pp. 50–56 (2015)
3. Marias, J., Garay, J., Toledo, N.: Toward an SDN-enabled NFV architecture. *IEEE Commun. Magazine* **53**(4), 187–193 (2015)
4. Han, B., Gopalakrishnan, V.: Network function virtualization: challenges and opportunities for innovations. *IEEE Commun. Mag.* **53**(2), 90–97 (2015)
5. Moore, A.W., Papagiannaki, K.: Toward the accurate identification of network applications. In: *Dovrolis, C. (ed.) PAM 2005. LNCS, vol. 3431, pp. 41–54. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-540-31966-5_4*
6. Xia, M., Shirazipour, M., Zhang, Y.: Network function placement for NFV chaining in packet/optical data centers. *J. Light. Technol.* **33**(8), 1565–1570 (2015)
7. Sen, S., Spatscheck, O., Wang, D.: Accurate, scalable in-network identification of p2p traffic using application signatures. In: *Proceedings of the 13th International Conference on World Wide Web*, pp. 512–521 (2004)
8. Nguyen, T., Armitage, G.: A survey of techniques for internet traffic classification using machine learning. *IEEE Commun. Surv. Tutor.* **11**(3), 37–52 (2008)
9. Vapnik, V.: SVM method of estimating density, conditional probability, and conditional density. In: *IEEE International Symposium on Circuits and Systems*, pp. 749–752 (2002)
10. Yan, H., Xu, D.: An approach to estimating product design time based on fuzzy v-support vector machine. *IEEE Trans. Neural Netw.* **18**(3), 721–732 (2007)
11. Hichem, S., Donald, G.: A hierarchy of support vector machines for pattern detection. *J. Mach. Learn. Res.* **7**(10), 2087–2123 (2006)
12. Tran, D.A., Nguyen, T.: Localization in wireless sensor networks based on support vector machines. *IEEE Trans. Parallel Distrib. Syst.* **19**(7), 981–994 (2008)
13. Benfano, S., Lucas, V., Ning, W.: Hybrid pattern matching for trusted intrusion detection. *Secur. Commun. Netw.* **4**(1), 33–43 (2011)
14. Chenfeng, Z., Christopher, L., Shanika, K.: A survey of coordinated attacks and collaborative intrusion detection. *Comput. Secur.* **29**(1), 124–140 (2010)
15. Sang, H., Won, S.: An anomaly intrusion detection method by clustering normal user behavior. *Comput. Secur.* **22**(7), 596–612 (2003)
16. Jing, J., Papavassiliou, S.: Enhancing network traffic prediction and anomaly detection via statistical network traffic separation and combination strategies. *Comput. Commun.* **29**(10), 1627–1638 (2006)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

