Front. Electr. Electron. Eng. China 2010, 5(2): 188–196
DOI 10.1007/s11460-010-0003-4

RESEARCH ARTICLE

Fei GAO, Xuesong QIU

A feedback control mechanism for adaptive SLO maintenance in dynamic service level management

Abstract With the increasing scale of information technology (IT) service system, traditional threshold-based static service level management (SLM) solution appears to be inadequate to meet current increasingly management requirement of SLM. Due to the stochastic service request rate, the random inherent failure and load surge of IT devices during service operating stage of large scaled IT system, service level objective (SLO) maintenance issue has become a realistic and important issue in dynamic SLM. This paper proposes a closed-loop feedback control mechanism to adaptively maintain SLO that service provider (SP) guaranteed at service operation stage. The mechanism can automatically tune the capacity of IT infrastructure according to service performance dispersion and reduce SLO violations. Considering that the tuning operations also affect service performance, fuzzy control is applied to alleviate the negative effect caused by tuning operations. In the dynamic SLM system that is applied with this mechanism compared with the traditional threshold-based solution, it is proved that the amount of SLO violations obviously decreases, the reliability of the service system increases relatively, and the resource utilization of IT infrastructure is optimized.

Keywords dynamic service level management (SLM), service level objective (SLO) maintenance, closed-loop feedback control, fuzzy control

1 Introduction

Service level management (SLM) bridges service provider (SP) and customers, and it is regarded as one of the most important parts of SLM, which is the key factor in providing high-quality information technology (IT) service. In SLM functional area, service level agreement (SLA) management is the core issue. As depicted in both SLM processes of Information Technology Infrastructure Library (ITIL) [1] and SLA handbook of next generation operation support system (NGOSS) [2], SLA management processes consist of the designing, measuring, monitoring, and reporting actions about key performance indicators (KPIs) to IT service quality. Among all these actions, the measuring and monitoring actions exist throughout IT service operation stage, and the goal is to monitor and manage the quality of service (QoS) parameters that SP promised in SLA. In general, these QoS metrics, namely, service level objectives (SLOs), are agreed between SP and customers in SLA according to the capacity of IT infrastructure and the service level requirement (SLR) promoted by customers. Then, SP can ensure the QoS they promised in SLA through maintaining SLO without violation. Many works in IT service management (ITSM) research areas focus on the monitoring of SLO (e.g., Refs. [3,4]), showing that it determines an essential part of SLM.

With the increasing scale of IT service system, however, the interdependent relationship among the inner components and functions becomes much more complex. As a consequence, high-level SLOs relate to more low-level factors so that it is hard to design a series of well-adapted SLOs on SLA design stage. Meanwhile, due to the stochastic service request rate, the random inherent failure and load surge of IT devices during service operating stage, service performance may fluctuate and will lead to SLO violation. Moreover, the negative fluctuation appears more obviously with the increasing complexity of the IT service system.

In traditional threshold-based static SLM solution, the capability of IT infrastructure required by service customers is determined at SLA design stage and cannot be adjusted according to the fluctuation of service performance during operation stage. The fluctuation of service performance will lead to frequent SLO violations, which causes the decline of customer satisfaction. Furthermore, it

Received January 25, 2010; accepted February 11, 2010

Fei GAO (✉), Xuesong QIU
State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China
E-mail: feigao@bupt.cn

needs to redesign the SLO in SLA when the customers' requirement changes; therefore, the static SLM solution appears to be inadequate to meet the current increasingly management requirement of SLM. Nowadays, dynamic SLM especially addressing dynamicity and flexibility is regarded as a more reasonable solution.

For dynamic SLM approach, a realistic problem is how to automatically tune the capacity of large distributed IT infrastructure in the light of real-time service performance in order to continuously provide high-quality IT services. That is to maintain the SLO in a reasonable range adaptively determined by the capacity of its IT infrastructure. Many researchers have studied the regulation of system resources to achieve a high-level objective. Reference [5] uses feedback control theory to dynamically plan the capacity of a heterogeneous server cluster to achieve appropriate high-level performance determined by SLO, which tries to solve the SLO maintenance problem just as ours but from a different standpoint. The work of Ref. [6] is closer to ours, which proposes a generic approach to automatically optimize configuration parameters for achieving SLO. However, Ref. [6] does not consider economizing resource when SLO is over satisfied during idle hours.

In this paper, a closed-loop feedback control mechanism is proposed to adaptively maintain the SLO, which has been agreed in SLA between SP and customers. Meanwhile, since the tuning operation itself also affects service performance, fuzzy control module is added to the feedback loop to avoid frequently tuning the capacity of IT infrastructure and alleviate the effect caused by tuning operations. Moreover, considering the utilization ratio of the entire IT infrastructure, an idle resources releasing scheme is set up in order to provide the just-in-time quality with relative low cost.

The rest of this paper is organized as follows: Section 2 details the feedback control mechanism for adaptive SLO maintenance in dynamic SLM system and describes how to tune the capacity of related IT infrastructure with fuzzy inference. Section 3 presents an experimental IT service scenario and simulation result. Finally, Section 4 discusses related work, and Section 5 offers the conclusion.

2 Feedback control mechanism for adaptive SLO maintenance

The goal of adaptive SLO maintenance is to adaptively tune the capacity of the related IT infrastructure according to the variation of service performance metrics in a certain range so that it can keep the service performance metrics in the SLO restricted range automatically.

The tuning quantity of the capacity is based on the difference between actual service performance at the specific IT service level and the referenced SLO guaranteed in SLA. The so-called capacity tuning, however, is actually realized by regulating the resources of the related low-level IT infrastructure. Hence, we establish the mapping from service performance domain to service capacity of IT infrastructure domain and the mapping from capacity domain of IT infrastructure to resource domain of IT infrastructure. Then, we can take difference of the resource metrics as the direct regulation quantity.

The practical tuning strategies are as follows. When the real-time service performance metric cannot meet SLO, the system should regulate the key resource metric of IT infrastructure associated with the service performance adaptively, so as to compensate the service performance variation. The key resource metric here is the one that has the greatest impact on service performance. During idle hours, however, the concurrent service requests retain a low state, and the performance metric usually over satisfies SLO. It could release some vacant resources of related IT infrastructure, so as to increase the utilization ratio of the entire IT infrastructure.

In addition, since the frequent tuning operations cost considerable system resource and lead to the service performance degradation, it also needs to take into account the variation rate of service performance. When the service performance metric exceeds the normal SLO range stipulated by SLA and the metric unfolds a rapid recovery trend back to the normal range, it could not tune the capacity of related IT infrastructure. Moreover, even when the service performance metric stays within the normal SLO range while the metric unfolds a rapid deviation trend away from the normal range, it could tune the capacity in advance in order to avoid SLO violation. To cope with this situation, fuzzy control module is adopted to process both the service performance metric and its variation rate and to generate a less intrusion regulation quantity of the capacity.

2.1 Architecture

The architecture of adaptive SLO maintenance system is proposed based on the architecture of SLM described in Refs. [7,8]. As shown in Fig. 1, the part surrounded by dashed lines is the management functional area of SLM. Moreover, it also needs to interact with other management functional areas in ITSM, such as capacity management, configuration management, etc., to fulfill the management objective of SLM.

The entire process of adaptive SLO maintenance proposed in this paper is a closed-loop with feedback control mechanism. It includes two parts: 1) forward monitoring process, as shown with solid arrows in Fig. 1, which consists of task controller (TC), performance monitor (PM), and performance measuring and calculating (PMC) module, and 2) feedback control process, as shown with hollow arrows in Fig. 1, which involves comparator, metric converter (MC), fuzzy controller (FC), and capacity

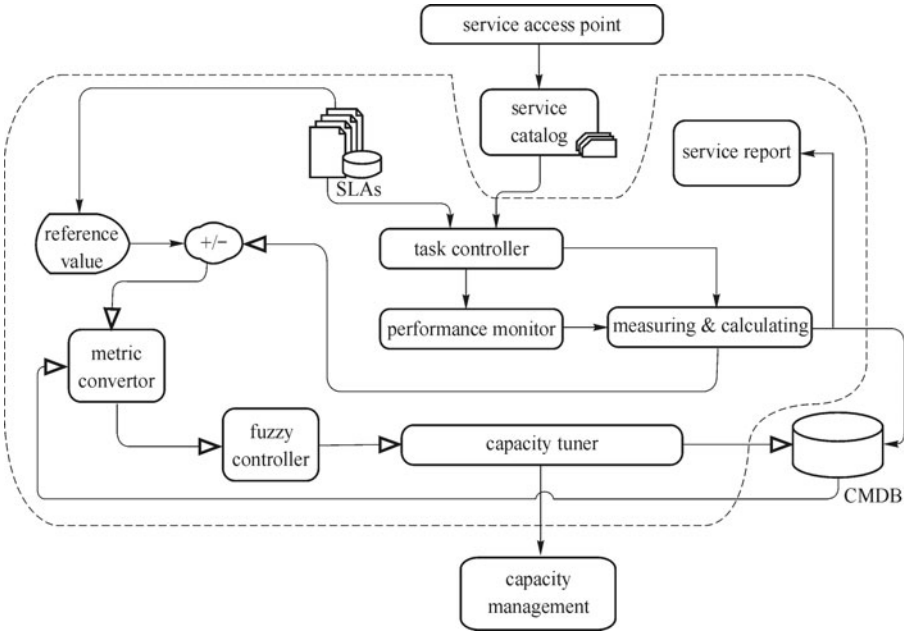

Fig. 1 Architecture of adaptive SLO maintenance system

tuner (CT). The outputs of adaptive SLO maintenance system are the readable service operating reports and real-time update information to configuration management database (CMDB).

2.2 Forward monitoring process

In forward monitoring process, the service access point receives user requests to IT service. The service catalogue sends the requested service ID and user ID to TC after receiving the request. TC searches SLA for the specific service level and related SLOs according to the service ID and user ID. Then, the service performance monitoring task and strategy can be built by TC.

Service performance monitoring task includes the following information: the list of performance metrics, the period of monitoring, the monitor mode, etc. Simple service performance metric can be collected by directly measuring the specific interface, such as response time. However, some compound service performance metric needs to calculate multiple simple metrics, such as throughput of service system. In our architecture, simple metrics can be monitored by PM module, and compound metrics should be calculated by PMC module based on related simple metrics.

At the end of forward monitoring process, the real-time performance value collected by PM and PMC updates CMDB, as well as generates the service report according to SLA between SP and customer. The service report can be either independence spreadsheet restricted in SLM functional area or the dashboard integrated with other functional areas, such as incident management, change management, etc.

2.3 Feedback control mechanism

2.3.1 Performance comparator

In feedback control loop [9], performance comparator gets the real-time service performance value from the output of forward monitoring process, compares it with the reference SLO value, and calculates the performance difference.

Let r denote the real-time service performance value, and r_{ref} denote the reference SLO value. Then, the service performance difference can be represented as

$$\Delta r = r - r_{\text{ref}}. \tag{1}$$

In principle, if Δr is positive, r is greater than r_{ref}, which means that CT can increase current capacity of IT infrastructure; if Δr is negative, r is less than r_{ref}, which means that CT can release part of the current capacity. However, the actual regulation quantity will be figured out by the following MC and FC modules.

2.3.2 Metric conversion

In order to convert service performance metric into capacity metric of IT infrastructure (e.g., the conversion from service response time to service request rate), we establish the mapping from performance domain to capacity domain through MC [8].

The role of MC can be concluded as detecting the reason in capacity domain of IT infrastructure, which causes the variation in service performance domain. MC receives the performance difference from performance comparator and builds the utility function of performance and capacity

according to the service structure and configuration information described in CMDB. Then, we can obtain the capacity difference through MC module and use the capacity difference as the regulation quantity to maintain the specific SLO.

In general, the service is a complex set of many types of IT devices. The capacity of IT infrastructure is determined by the associated resource quantity. It can be depicted as the following model. Assuming that the service covers n physical devices, the resource set of device j can be represent as

$$C_j = (C_1, C_2, ..., C_k, ..., C_n), \tag{2}$$

where C_k denotes the kth resource metric of device j, e.g., C_1 may denote CPU utilization, and C_2 may denote the amount of memory allocation. Let u_j denote the capacity of device j, i.e., service rate of device j. The service rate u_j correlates to the resource set C_j, and u_j can be represented as the linear combination of C_j:

$$u_j = \beta_j C_j^{\mathrm{T}}, \tag{3}$$

where the proportional coefficient $\beta_j = (\beta_1, \beta_2, ..., \beta_n)$ can be obtained by a multiple linear regression analysis.

In addition, different customers have different service levels; the mapping from service capacity domain to resource domain should take into account the disparity in the capacity of related IT infrastructure required by the different service levels. Assuming that the requested service has i levels, and each service level has different requirement to the service rate. Let u_i denote the service rate at service level i, and then, let u_{ij} denote the service rate of device j at level i. Due to the cask effect, the overall service rate is determined by the lowest service rate among all the devices involved in the current service. Hence, the u_i of related IT infrastructure at level i can be represented as

$$u_i = \min_j (\beta_j C_j^{\mathrm{T}}). \tag{4}$$

Since the real-time service performance has tight association with service rate and user request intensity. The user request intensity cannot be controlled by the SLM system. Therefore, MC only needs to focus on the correlation between service performance and service rate. Given the service rate u_i at the service level i, the kth service performance metric can be represented as

$$r_{ik} = f_k(u_i),\ k \in [1, m], \tag{5}$$

where f_k denote the mapping from the service rate to the kth performance metric. Since usually there is not a common model in IT service area, f_k changes with the specific service. We can build the specific f_k according to the service model information stored in CMDB. In this way, we get the entire relationship among resource domain, capacity domain, and performance domain, as shown in Fig. 2.

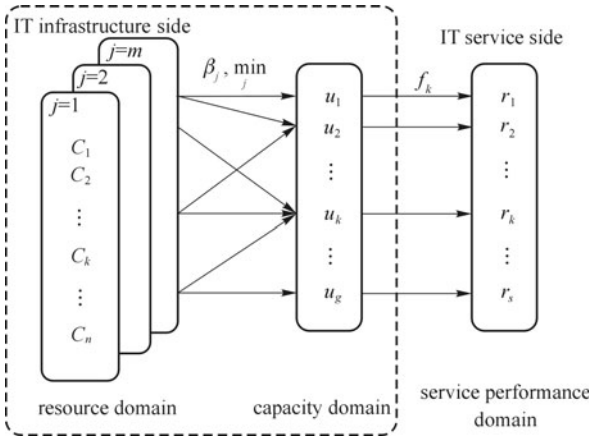

Fig. 2 Mappings of resource domain to capacity domain and capacity domain to service performance domain

We set MC to convert the performance metrics to capacity metrics by referring the clear mapping equations before. As a consequence, u_i can be represented as

$$u_i = f_k^{-1}(r_{ik}),\ k \in [1, m], \tag{6}$$

where f_k^{-1} denotes the inverse mapping determined by f_k, and m is numbers of the SLOs at level i specified in SLA of the requested service. Let U denote the maximal capacity that the related IT infrastructure can produce, and let U_{free} denote the free capacity of the current related IT infrastructure. Then, we can obtain Δu ($U_{\mathrm{free}} - U \leqslant \Delta u \leqslant U_{\mathrm{free}}$) caused by Δr, where Δu denotes the capacity difference, and Δr denotes the performance difference.

2.3.3 Capacity tuning based on fuzzy control logic

FC obtains the capacity difference Δu from the output of MC and takes it as primary compensation dosage of the service capacity, denoted by $e(t)$. Let $e'(t)$ denote the variation rate, then

$$e'(t) = \frac{\mathrm{d}e(t)}{\mathrm{d}t}. \tag{7}$$

Hence, the input of FC is $e(t)$ and $e'(t)$.

Let $U(t)$ denote the fuzzy output of FC after fuzzy inference. As shown in Fig. 3, fuzzification module builds the membership functions [10] of the input variables $e(t)$ and $e'(t)$ and the output variable $U(t)$ at the beginning of the fuzzy control process. The fuzzification can be regarded as the mapping form concrete variable to fuzzy variable.

After fuzzification, we obtain the fuzzy set of $e(t)$, $e'(t)$, and $U(t)$, denoted, respectively, by A, B, and Z, as shown in Tables 1 and 2. Then, we set a series of fuzzy rules for fuzzy inference, as shown in Table 3. In general, fuzzy rules are represented as the expression "IF…THEN…". The result of fuzzy inference is an integrated set according

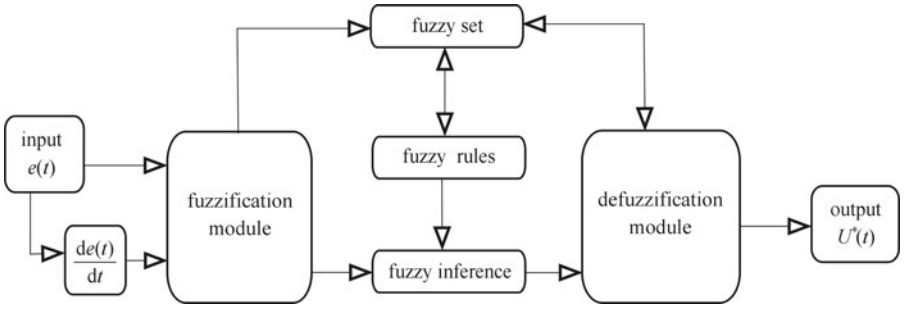

Fig. 3 Architecture of fuzzy controller

Table 1 Fuzzy sets of input variable

input	fuzzy set	description
	A1	greater than reference value
$e(t)$: A	A2	equal to reference value
	A3	lower than reference value
	B1	increasing rapidly
$e'(t)$: B	B2	almost no change
	B3	reducing rapidly

Table 2 Fuzzy set of output variable

output	fuzzy set	description
	Z1	increase the capacity
$U(t)$: Z	Z2	no change
	Z3	reduce the capacity

Table 3 Fuzzy rules

fuzzy set		input B		
		B1	B2	B3
	A1	Z3	Z3	Z2
input A	A2	Z3	Z2	Z1
	A3	Z2	Z1	Z1

to each fuzzy rule. In some complex models, however, the fuzzy rules [10] are not easy to determine and need to associate with knowledge base and expert system.

Through the fuzzy inference, we can get a series of the fuzzy numbers of output variable according to each fuzzy rule that we defined before. Let θ_k denote the fuzzy number and $\theta_k[U(t)]$ denote the specific fuzzy set, where $\theta_k \in (0,1)$. Then, we can obtain the overall output fuzzy set $Z[U(t)]$ by union of $\theta_k[U(t)]$, i.e.,

$$Z[U(t)] = \bigcup_{k=1}^{n} \theta_k[U(t)]. \tag{8}$$

However, the output fuzzy set only reflects the tendency of capacity tuning and is not sufficient to be the regulation quantity of capacity. Therefore, the output fuzzy sets still need defuzzification. We can get the concrete output $U^*(t)$ through defuzzifying $Z[U(t)]$ and take it as the concrete regulation quantity.

There are various ways of defuzzification, and we adopt the center of gravity defuzzification (CGD) here. CGD is the gravity center of the area covered by the membership function curve of output fuzzy sets. For continuous variable, we can calculate the concrete value as

$$U^*(t) = \frac{\int_U U(t)Z[U(t)]\mathrm{d}U(t)}{\int_U Z[U(t)]\mathrm{d}U(t)}. \tag{9}$$

2.3.4 Resource regulation of IT infrastructure

The tuning capacity of related IT infrastructure is actually the regulation of IT infrastructure resource allocation. CT is used to regulate the resource allocation of related IT infrastructure and interacts with capacity management system.

CT receives $U^*(t)$ from FC as input variable. Then, CT regulates the IT infrastructure resource allocation through invoking the specific management interfaces of service management functional area. The regulation strategy is as follows: we first regulate the kth resource metric C_{jk} of the device j; if C_{jk} makes the greatest impact on the service rate u_i, then we regulate the other metric that makes the second greatest impact on the service if the capacity variation caused by C_{jk} regulation cannot meet $U^*(t)$. Likewise, we continue to regulate the resource metric until the capacity variation meets $U^*(t)$.

3 Simulation and analysis

3.1 IT service simulation scenarios

We consider the online payment service scenarios of a bank as an example. At present, the online payment service system of the bank is still using the traditional threshold-based management strategy for the monitoring and management of SLO. When the performance metric exceeds the threshold stipulated by its SLO, the system generates alarms to other management processes.

We take service response time as the service performance metric in the simulation. For the online payment service, service response time can be defined as the processing time of transaction consumed by the service side.

The online payment service can be simply regarded as $M/M/1$ queuing model. The users request rate of the online payment service follows λ parameterized Poisson distribution and the service time follows u parameterized negative exponential distribution. According to the queuing-theoretic formula of $M/M/1$, the response time can be represented as

$$r_i = \frac{1}{u_i - \lambda_i}, \tag{10}$$

where λ_i is the user request rate of service level i. The variation of service rate $e(t)$ can be represented as

$$e(t) = \frac{r_i - r_{\text{ref}}}{\partial r_i / \partial u_i} = -(r_i - r_{\text{ref}})(u_i - \lambda_i)^2. \tag{11}$$

We use $e(t)$ and $e'(t)$ as the inputs of FC. For continuous input variables, the membership function is generally the Gaussian function, as shown in Table 4. Through the fuzzy inference according to the fuzzy rules depicted in Table 3, we obtain the output fuzzy set $Z[U(t)]$, which denotes current Δu_i roughly, as shown by the surface in Fig. 4. Then, we defuzzify the output fuzzy set $Z[U(t)]$ with the CGD algorithm depicted in Eq. (4) to get the concrete output value $U^*(t)$, i.e., the capacity regulation quantity.

Table 4 Membership functions of input and output variables

input and output	fuzzy set	membership function
$e(t)$: A	A1(x)	gaussmf(x, [0.3 1])
	A2(x)	gaussmf(x, [0.3 1])
	A3(x)	gaussmf(x, [0.3 −1])
$e'(t)$: B	B1(x)	smf(x, [0.0825 0.495])
	B2(x)	gaussmf(x, [0.214 0])
	B3(x)	zmf(x, [−0.5 −0.087])
$U(t)$: Z	Z1(x)	smf(x, [0.0608 0.595])
	Z2(x)	gaussmf(x, [0.1416 0])
	Z3(x)	zmf(x, [−0.5423 −0.0608])

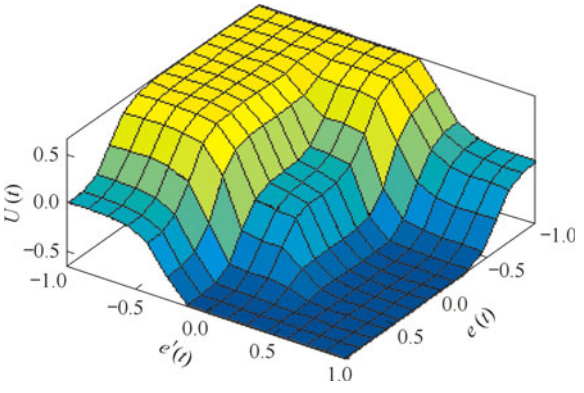

Fig. 4 Output fuzzy set after fuzzy inference

At the end of the feedback loop, CT takes $U^*(t)$ as the capacity regulation quantity of related IT infrastructure at service level i. According to the mapping from resource domain to capacity domain, CT invokes the specific management interfaces of service management functional area to increase or decrease the corresponding infrastructure resource metric C_j. Therefore, u_i changes with the variation of resource allocation. As we depicted before, the regulation of resources is according to the strategy that defines the priority to regulate the kth metric C_{jk} of the device j. The priority is based on the resource metric that makes the greatest impact on u_i.

3.2 Simulation setup

First, we build a simulation adaptive SLO maintenance system on Sun V440 (Solaris 10) based on the feedback mechanism proposed in this paper. CMDB derived from the practical online payment service system of the bank runs on an HP DL580, and this machine uses the Windows Server 2003 operating system (OS) and runs on Oracle 9i database. The service access portal runs on a Lenovo ThinkPad T400 with Windows XP. Then, we use the real-time data of service request rate (shown in Fig. 5) within one week as the input data of our simulation system, and the data derived from the threshold-based IT service management system of the bank. By simulation, we can obtain the service performance metric based on our simulation system and adopt the feedback control mechanism.

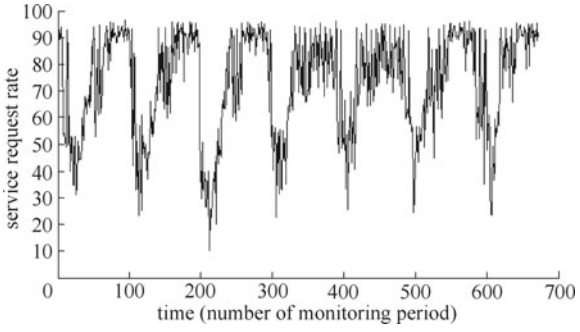

Fig. 5 Service request rate of online payment service

We set the period of performance metric monitoring to 15 min, just the same as the threshold-based management system, and set service rate of the static threshold-based management system to 100. In this simulation system, we focus on the SLO of response time at the individual service level of the online payment service system. Assuming that the SLO of response time is less than 120 ms, and we set its breach budget [11] to 25%. The breach budget here is the

allowed percentage of violation of the target average response times during the evaluation period. As a consequence, the alarm events will be triggered once the service response time exceeds 150 ms.

3.3 Experimental result

After the simulation with the feedback control mechanism we proposed in this paper, we obtained the response time of the original threshold-based IT service management system, as shown in Fig. 6(a), and the optimized data of response time of the online payment service at individual service level, as shown in Fig. 6(b), where the straight line (120 ms) denotes the threshold defined in SLO, and the dotted line (150 ms) denotes the upper limit of the breach budget.

In the original system, the worst service response time is greater than 280 ms during busy hours and frequently breaches the alarm threshold. During the idle hours, the service response time even is less than 20 min. We can see that the overall capacity of the IT infrastructure is adequate to meet the service requirement. However, it lacks dynamic runtime controlling and regulation, which leads to the sharp fluctuation of the response time. Moreover, the utilization of the IT resource is unreasonable.

Compared with the original management system, the service response time of our management system is obviously optimized. Simulation results show that after adaptive tuning with the feedback control mechanism we proposed, the worst service response time is controlled below 230 ms during the rush hours, and the violations of service response time has been significantly reduced. In our simulation system, the service response time is almost stabilized in the dynamic range restricted by SLO.

Meanwhile, the feedback control mechanism we proposed can intelligently release some resources of the IT infrastructure, which will increase the service response time a little during the idle hours. As a consequence, the resource utilization of the IT infrastructure becomes relatively more reasonable. We observed the central processing unit (CPU) utilization of web application server of both original threshold-based system and our proposal experimental system. As shown in Fig. 7, the CPU utilization of web application server appears flatter after applying the feedback mechanism in this paper.

4 Related work

There have been several proposals of the automatic regulation of system resources to achieve high-level SLO in dynamic SLM research area, especially using control theory. Reference [12] describes a system that performs online optimization of a web server by using hill climbing techniques. However, the approach taken requires a detailed knowledge of the system being optimized in order to construct the queuing models. A similar concern arises with Ref. [13], which considers how to maximize

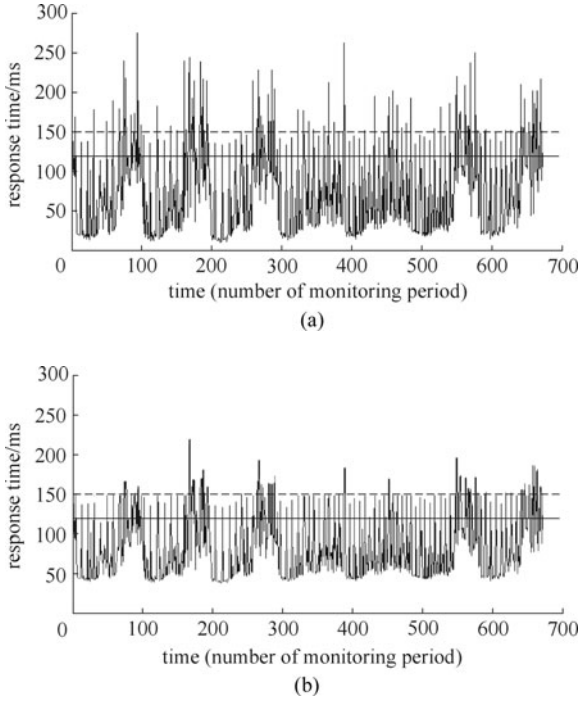

Fig. 6 Comparison of service response time. (a) Response time of original threshold-based system; (b) response time of proposal system

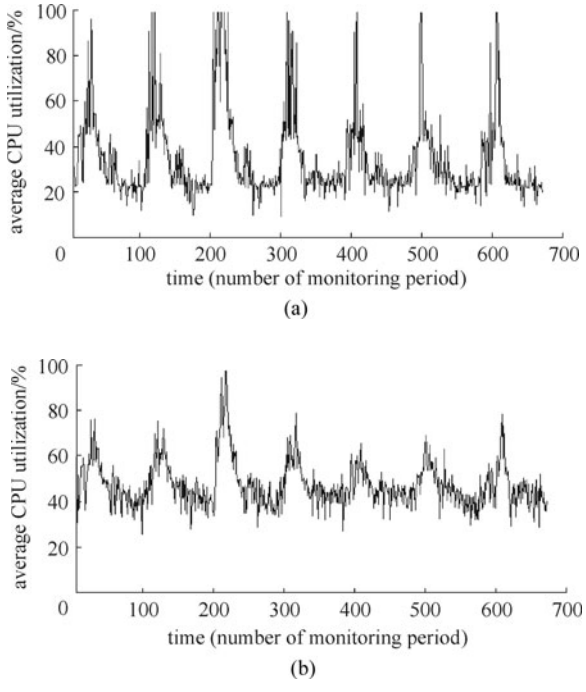

Fig. 7 Comparison of CPU utilization of web application server. (a) Average CPU utilization of web application server of threshold-based system; (b) average CPU utilization of web application server of proposal system

profits based on queuing-theoretic formulas. However, they only consider regulating one resource metric. In this paper, we build the mapping relationship between the service performance metrics and resources metrics. Based on the mapping relationship, we can handle the inter-dependencies between the resource metrics and support online optimization.

For the resource allocation optimization problems in IT service management, there have been several studies. For example, Ref. [14] proposes a strategy to optimize the resource allocation in order to minimize network traffic and resource underutilization. Similar concerns arise with studies in Ref. [15], and they deal with the problem of resource allocation for maximizing SLA profits of an e-commerce provider. Reference [16] uses an adaptive strategy to manage which connections are affected by failures and maximize the compliance with the SLAs. However, all these works assume service system workload is static. In contrast, our work considers the service performance fluctuation caused by dynamic workload and sets up a compensation scheme with feedback control theory.

Many researchers focus about the problem of adaptive resource regulation but from the capacity management perspective, which closely relates with SLM. For instance, Ref. [17] presents a model for self-adaptive capacity management in shared environments, driven by a cost model based on SLA contracts. Reference [18] considers mapping the workload level of a data center to its observed influence on the system. Aiming at maximizing the sum of utility functions of the performance, Ref. [18] relies on a state-space based on the past behavior so as to provide information for future capacity regulation decisions.

5 Conclusion and future work

A closed-loop feedback controlling mechanism is proposed for dynamically maintaining the SLO that SP and customer determined in SLA during service operation stage in this paper. The mechanism can automatically tune the capacity of IT infrastructure according to service performance dispersion and reduce SLO violations. Considering that the tuning operations also affect service performance, FC module is added to alleviate the negative effect caused by tuning operations and to avoid the frequent service capability tuning operations.

Compared with the static threshold-based SLM system, the system adopting the feedback control mechanism proposed in this paper can effectively reduce SLO violations in the service system. The worst service response time is controlled in a reasonable range during the rush hours. Hence, the user satisfaction of the requested service is improved. At the same time, the resource utilization of the related IT infrastructure has also been optimized.

At present, we regard the mapping from the resources domain of IT infrastructure to the capacity domain of IT infrastructure as linear mapping. However, the mapping is nonlinear in actual service systems. In the future work, we will build nonlinear mapping model on this basis. Nowadays, ITSM concerns not only IT infrastructure and service processes but also the business performance factors. References [19,20] first introduce the idea of business-driven IT management (BDIM) and propose that the service management activities in ITSM must be aligned with business, in order to gain the maximum business revenue. We will add the business performance factors into our feedback control mechanism for SLO maintenance in SLM system and extend SLM to the business layer in our future work.

Acknowledgements This work was partly supported by the State Key Development Program for Basic Research of China (No. 2007CB310703), the Foundation for Innovative Research Groups of the National Natural Science Foundation of China (Grant No. 60821001), and the National High Technology Research and Development Program of China (No. 2008AA01Z201).

References

1. IT Infrastructure Library. ITIL Service Delivery and ITIL Service Support (Version 2). OGC, UK
2. SLA Management Handbook: Volume 1—Executive Overview. Morristown: Tele Management Forum (TMF), 2005
3. Rosario S, Benveniste A, Jard C. Monitoring probabilistic SLAs in Web service orchestrations. In: Proceedings of the 11th IFIP/IEEE International Symposium on Integrated Network Management. 2009, 474–481
4. Racz H P, Stiller B. Monitoring of SLA compliances for hosted streaming services. In: Proceedings of the 11th IFIP/IEEE International Symposium on Integrated Network Management. 2009, 251–258
5. Nakadai S, Taniguchi K. Server capacity planning with priority allocation for service level management in heterogeneous server clusters. In: Proceedings of the 10th IFIP/IEEE International Symposium on Integrated Network Management. 2007, 753–756
6. Diao Y, Eskesen F, Froehlich S, Hellerstein J L, Keller A, Spainhower L F, Surendra M. Service level management: a dynamic discovery and optimization approach. IEEE Transactions on Network and Service Management, 2004, 1(2): 83–91
7. Diao Y, Hellerstein J L, Parekh S. Using fuzzy control to maximize profits in service level management. IBM Systems Journal, 2002, 41 (3): 403–420
8. Chen Y, Iyer S, Liu X, Milojicic D, Sahai A. SLA decomposition: translating service level objectives to system level thresholds. In: Proceedings of the Fourth International Conference on Autonomic Computing. 2007, 3–3
9. Hellerstein J L, Diao Y, Parekh S, Tilbury D M. Feedback Control of Computing Systems. MA: Wiley InterScience, 2004, 23–45
10. Zhang H, Liu D. Fuzzy Modeling and Fuzzy Control. MA: Birkhäuser, 2006, 89–101

11. Breitgand D, Henis E A, Shehory O, Lake J M. Derivation of response time service level objectives for business services. In: Proceedings of the 2nd IEEE/IFIP International Workshop on Business-Driven IT Management. 2007, 29–38
12. Menascé D A, Barbará D, Dodge R. Preserving QoS of e-commerce sites through self-tuning: a performance model approach. In: Proceedings of the 3rd ACM Conference on Electronic Commerce. 2001, 224–234
13. Liu Z, Squillante M S, Wolf J L. On maximizing service-level-agreement profits. In: Proceedings of the 3rd ACM Conference on Electronic Commerce. 2001, 213–223
14. Zhu X, Santos C, Ward J, Beyer D, Singhal S. Resource assignment for large-scale computing utilities using mathematical programming. Hewllet Packard Laboratories. Technical Report. HPL-2003-243R1, 2003
15. Villela D, Pradhan P, Rubenstein D. Provisioning servers in the application tier for e-commerce systems. In: Proceedings of the Twelfth IEEE International Workshop on Quality of Service. 2004, 57–66
16. Mykkeltveit A, Helvik B E. Adaptive management of connections to meet availability guarantees in SLAs. In: Proceedings of the 11th IFIP/IEEE International Symposium on Integrated Network Management. 2009, 545–552
17. Abrahao B, Almeida V, Almeida J, Zhang A, Beyer D, Safai F. Self-adaptive SLA-driven capacity management for Internet services. In: Proceedings of the 10th IEEE/IFIP Network Operations and Management Symposium. 2006, 557–568
18. Walsh W E, Tesauro G, Kephart J O, Das R. Utility functions in autonomic systems. In: Proceedings of the First International Conference on Autonomic Computing. 2004, 70–77
19. Moura A, Sauve J, Bartolini C. Business-driven IT management—upping the ante of IT: exploring the linkage between it and business to improve both IT and business results. IEEE Communications Magazine, 2008, 46(10): 148–153
20. Marques F, Sauve J, Moura A. Business-oriented capacity planning of IT infrastructure to handle load surges. In: Proceedings of the 10th IEEE/IFIP Network Operations and Management Symposium. 2006, 1–4