KIPS Tr. Comp. and Comm. Sys.
Vol.4, No.3 pp.81~84 pISSN: 2287-5891

http://dx.doi.org/10.3745/KTCCS.2015.4.3.81

Resource Prediction Technique based on Expected Value in Cloud Computing

Yeongho Choi† · Yujin Lim††

ABSTRACT

Cloud service is one of major technologies in modern IT business. Due to the dynamics of user demands, service providers need VM(Virtual Machine) provisioning mechanism to predict the amount of resources demanded by cloud users for the next service and to prepare the resources. VM provisioning provides the QoS to cloud user and maximize the revenue of a service provider by minimizing the expense. In this paper, we propose a new VM provisioning technique to minimize the total expense of a service provider by minimizing the expected value of the expense based on the predicted demands of users. To evaluate the effectiveness of our prediction technique, we compare the total expense of our technique with these of the other prediction techniques with a series of real trace data.

Keywords : Resource Prediction, VM Provisioning, Over-Provisioning, Under-Provisioning, Expected Value

클라우드 환경에서 기대 값 기반의 동적 자원 예측 기법

최 영 호† · 임 유 진††

요 약

클라우드 서비스는 다양한 장점들 덕분에 현대 IT 사업에서 주목을 받고 있다. 클라우드 환경에서 사용자의 요구는 동적이기 때문에 서비스 제공자는 사용자 요구량을 예측하고 이를 기반으로 자원을 제공하는 VM(Virtual Machine) 프로비저닝 기법이 필요하다. VM 프로비저닝은 사용자의 QoS를 만족시키고 자원 관리 비용을 최소화하여 서비스 제공자의 이득을 최대화하는 것을 목적으로 한다. 본 논문에서는 효율적인 VM 프로비저닝을 위해 사용자의 자원 요구량을 예측하고, 이를 기반으로 서비스 제공자의 총 경비에 대한 기대 값을 최소화시키기 위한 새로운 VM 프로비저닝 기법을 제안한다. 또한 제안 기법의 성능 분석을 위하여 실제 데이터를 이용하여 자원 요구 예측량과 자원 제공량을 계산하고, 이를 다른 기법들과 비교함으로써 제안 기법이 서비스 제공자의 총 경비를 최소화함을 보여준다.

키워드 : 자원 예측, VM 프로비저닝, 오버 프로비저닝, 언더 프로비저닝, 기대 값

1. 서 론

클라우드 서비스는 유동성, 확장성, 편의성, 낮은 비용과 같은 다양한 장점들 덕분에 현대 IT 사업에서 많은 주목을 받고 있다. 클라우드 서비스는 서비스 대상에 따라서 기업을 위한 프라이빗 클라우드, 일반 대중을 위한 퍼블릭 클라우드, 보안 기능과 비용의 효율성을 강화한 하이브리드 클라우드로 구분한다[1]. 또한 IaaS(Infrastructure as a Service), PaaS(Platform as a Service), SaaS(Software as a Service) 같이 사용자의 요구에 따라 다양한 유형의 서비스를 제공한다. 특히, IaaS에서 서비스 제공자는 물리적인 자원을 가상화 기술을 이용하여 VM(Virtual Machine) 단위로 사용자들에게 제공한다.

클라우드 환경에서 사용자가 요구하는 자원의 양과 유형은 동적이기 때문에 사용자의 QoS(Quality of Service)를 만족시키는 것은 어렵다. 기존의 전통적인 자원 할당 기법은 사용자 요구가 증가하면 데이터 센터를 확장시키는 방법으로 문제를 해결하였다. 그러나 보다 효율적인 자원 관리를 위해서는 사용자들의 요구를 정확히 예측하고 이를 기반으로 자원을 준비하는 것이 필요하다. 따라서 사용자의 동적인 자원 요구에 따라 효율적으로 자원을 할당하기 위한 VM 프로비저닝(VM provisioning)이 요구된다.

VM 프로비저닝은 가까운 미래에 발생할 서비스의 자원 요구량을 계산하고, 해당 요구량을 만족시키기 위한 자원을 미리 준비하는 기술이다. VM 프로비저닝은 사용자의 QoS를 만족시키고 자원 관리 비용을 최소화하여 서비스 제공자의 이득을 최대화한다. 서비스 제공자는 사용자에게 SLA(Service Level Agreement)에 계약된 QoS를 제공해야 한다. 서비스 제공자가 QoS를 만족시키면 사용자로부터 서비스 이용에 대한 대가를

※ 본 연구는 경기도의 경기도지역협력연구센터(GRRC) 사업의 일환으로 수행하였음[(GRRC수원2014-B3), 클라우드 기반 지능형 영상 보안 감시 시스템 개발].
※ 이 논문은 2014년도 정부(교육과학기술부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임(2014043915).
† 비 회 원 : 수원대학교 컴퓨터학과 석사과정
†† 종신회원 : 수원대학교 정보미디어학과 교수
Manuscript Received : September 16, 2014
First Revision : November 5, 2014
Accepted : November 10, 2014
* Corresponding Author : yujin@suwon.ac.kr

받지만, 만족시키지 못하면 서비스 제공자는 사용자에게 그에 따른 보상을 지급해야 한다. QoS를 떨어뜨리는 주된 원인 중 하나는 VM 프로비저닝 지연시간(delay)이다. VM 프로비저닝에서 VM 생성, VM 할당 및 배분, 소프트웨어 설치, VM 검증, 안전성 확인 등의 요인으로 지연시간이 발생되는 것은 피할 수 없다. 따라서 VM 프로비저닝 지연시간을 줄이기 위하여 사전에 자원 요구량을 예측하고 필요한 서비스 자원을 준비하는 기법이 요구된다.

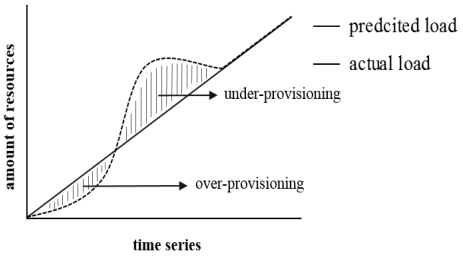


Fig. 1. Two cases of VM provisioning

VM 프로비저닝에서 발생할 수 있는 상황은 Fig. 1과 같다. 첫째로, 오버 프로비저닝(over-provisioning)은 서비스 제공자의 준비된 자원이 실제 요구량에 비해 많은 경우로 서비스 제공자에게는 잉여 자원만큼의 손해가 발생한다. 둘째로, 언더 프로비저닝(under-provisioning)은 서비스 제공자의 준비된 자원이 실제 요구량에 비해 부족한 경우로, 부족한 VM 준비를 위한 VM 프로비저닝 지연시간이 발생할 수 있으며, 이로 인하여 서비스의 품질이 하락할 수 있다. 서비스 제공자는 서비스 품질 하락으로 인한 SLA 위반에 대한 보상을 지급해야 할 수 있다. 따라서 2절에서는 서비스 품질을 위한 VM 프로비저닝 방법을 설명하고, 3절과 4절에서는 사용자의 QoS를 만족시키고 서비스 제공자의 이득을 최대화하기 위한 새로운 VM 프로비저닝 기법을 제안하고 성능을 평가한다.

2. 관련 연구

클라우드 환경에서 사용자에게 QoS를 제공하기 위해서는 VM 프로비저닝 지연시간을 최소화해야 한다. VM 프로비저닝 지연시간을 최소화하기 위한 대표적인 기술은 서비스의 자원 요구량에 대한 예측이다. VM 프로비저닝을 위한 추가적인 지연시간의 발생을 최소화하기 위해서 서비스 제공자는 사전에 다음 서비스의 자원 요구량을 예측하고 이를 기반으로 자원을 미리 준비한다.

VM 프로비저닝은 두 가지 방법(Reactive와 Proactive)으로 자원을 관리한다[1]. 첫째로, Reactive 방법은 서비스의 요청이 발생하면 즉각적으로 자원을 할당한다. 다시 말해서 서버가 모니터링을 통하여 자원 요청량이 증가했음을 확인하면 그때 자원을 할당한다. Reactive 방법은 새로운 VM의 생성과 할당이 즉시 실행된다면 효과적이다. 하지만 실제로 VM의 생성과 할당에는 지연시간이 요구된다. 이것은 일반적인 서비스에는 적합할 수도 있지만 실시간 처리를 요구하는 서비스에는 적합하지 않다. 따라서 Reactive 방법은 VM 프로비저닝 지연시간을 발생시키고 사용자의 QoS를 떨어뜨릴 수 있다. 둘째로, Proactive 방법은 다음 서비스의 자원 요구량을 예측하여 사전에 필요한 자원을 준비하는 방법이다. 서비스 제공자는 서비스의 다양한 자원 요구량을 예측하여 미리 준비함으로써 VM 프로비저닝을 위한 추가적인 지연시간의 발생을 예방한다. 따라서 정확하게 자원 요구량을 예측하고 이를 기반으로 자원 준비량을 결정할 수 있는 기법이 필요하다. [2]는 오버 프로비저닝과 언더 프로비저닝의 경우를 고려하여 서비스 제공자의 손해를 계산하는 함수를 제안하였다. 서비스 제공자가 실제 서비스 요청량에 비하여 VM을 많이 준비하는 경우에는 서비스를 처리하고 남는 잉여 자원에 대하여 손해가 발생한다. 실제 서비스 요청량에 비하여 VM을 적게 준비하는 경우는 부족한 자원 준비를 위한 VM 프로비저닝 지연시간이 발생하게 되며, 이로 인하여 SLA에 대한 위반이 발생할 수 있으며 이때 서비스 제공자는 사용자에게 벌금을 지급해야 한다. 따라서 서비스 제공자의 손해는 잉여 자원에 대한 손해와 SLA 위반에 대한 손해를 합한 것이 된다. [1]은 소비자의 QoS를 만족시키면서 지원하는 서비스 개수를 최대화하는 기법이다. 이는 서비스의 자원 요청량에 제한을 두어 서버의 처리 능력을 일정 수준 이상으로 유지함으로써 지원하는 서비스 개수를 최대화시킨다. 또한 SLA 위반에 대한 벌금을 최소화하여 사용자에게 QoS를 보장하고 서비스 제공자의 이득을 최대화한다.

기존의 많은 VM 프로비저닝 기법들은 서비스의 자원 요구량을 예측하고, 해당 예측량을 다음 서비스에 제공할 자원량으로서 자원을 준비한다. 그러나 이들 기법은 사용자에게 QoS는 제공하지만 서비스 제공자의 이득을 최대화하지는 않는다. 본 논문에서는 기존의 자원 예측 기법을 기반으로 서비스 제공자의 총 경비에 대한 기대 값을 최소화하여 서비스 제공자의 이득을 최대화하는 자원 제공량 결정 기법을 제안한다.

3. 제안 기법

본 논문에서는 자원 제공량 결정을 위하여 서비스 제공자의 총 경비에 대한 기대 값을 최소화하기 위한 함수를 다음과 같이 제안한다. 총 경비에 대한 기대 값은 오버 프로비저닝과 언더 프로비저닝이 발생할 각 확률에 그에 따른 발생 경비를 곱한 값의 합으로 계산한다. 오버 프로비저닝 상황에서는 서비스 사용자의 손해가 없지만 서비스 제공자는 잉여 자원에 대한 손해가 발생한다. 언더 프로비저닝 상황에서 서비스 제공자는 VM 프로비저닝 지연시간으로 인한 SLA 위반에 대한 벌금을 사용자에게 지급해야 한다. 따라서 서비스 제공자의 총 경비에 대한 기대 값 C_{total}은 다음과 같이 계산할 수 있다.

$$C_{total} = (1-P)C_{over} + P \cdot C_{under} \quad (1)$$

P는 언더 프로비저닝이 발생할 확률이며, C_{over}와 C_{under}는 각기 오버 프로비저닝과 언더 프로비저닝일 때 서비스 제공자의 손해비용을 나타낸다. 제안된 기법에서는 M/G/1

큐잉모델에 따라 VM을 모델링하였으며 트랜잭션 단위로 서비스를 처리한다. Table 1은 제안 기법의 주요 변수들을 설명한 것이다.

오버 프로비저닝에서 서비스 제공자에게는 서비스 제공 경비와 잉여 자원에 대한 손해가 발생한다. 오버 프로비저닝 상황($x \geq \lambda$)에서 총 경비는 다음과 같이 계산한다.

$$C_{over} = \lambda \cdot C + (x-\lambda) C_{wasted} \tag{2}$$

언더 프로비저닝에서 총 경비는 서비스 제공 경비와 SLA 위반에 대한 벌금을 포함한다. 부족한 자원 전체에 대하여 SLA 위반이 발생하는 것은 아니므로 SLA에 대한 위반 확률은 요청된 서비스의 응답시간을 SLA의 응답시간과 비교하여 계산한다. 만약 서비스 제공자의 준비된 자원이 사용자의 요구량보다 적다면 VM 프로비저닝을 위한 추가적인 지연시간이 발생할 수 있다. 이 지연시간은 SLA의 응답시간을 초과할 수 있다. 제안 함수에서는 SLA이 위반될 확률을 $Pr(T_{res} \geq T_{SLA})$로 나타내며, 마르코브 부등식을 이용하여 다음과 같이 계산할 수 있다[3].

$$Pr(T_{res} \geq T_{SLA}) \leq \frac{E[T_{res}]}{T_{SLA}} \tag{3}$$

$$Pr(T_{res} \geq T_{SLA}) \fallingdotseq \min\left(\frac{E[T_{res}]}{T_{SLA}}, 1\right) \tag{4}$$

평균 서비스 응답시간 $E[T_{res}]$는 평균 서비스 처리시간을 이용하여 $\frac{\lambda}{|Avg-\lambda|x}$로 계산한다[4]. 결과적으로 언더 프로비저닝 상황($x < \lambda$)에서 총 경비는 다음과 같이 계산한다.

$$C_{under} = x \cdot C + (\lambda - x)\min\left(\frac{\lambda}{|Avg-\lambda|x\, T_{SLA}}, 1\right) C_{penalty} \tag{5}$$

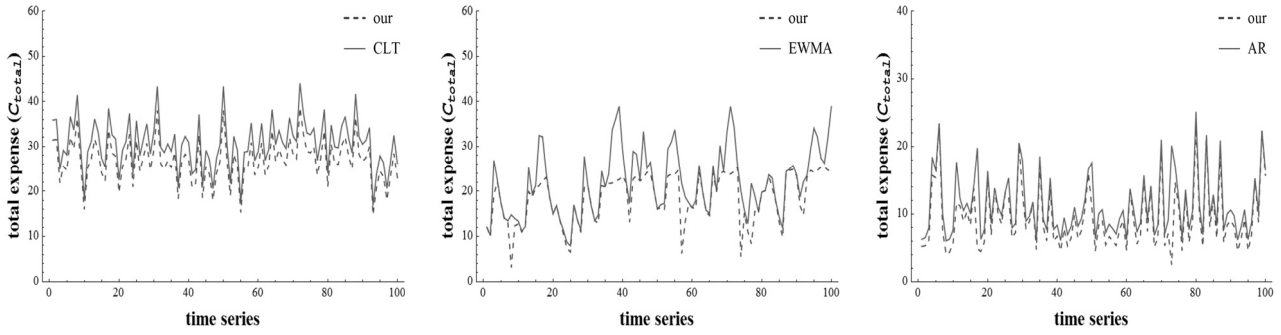
Table 1. The main parameters

C	서비스 제공 비용
C_{wasted}	잉여 자원당 손해액
$C_{penalty}$	SLA 위반에 따른 서비스당 벌금
x	다음 서비스 제공을 위해 준비할 VM 개수
λ	다음 서비스에서 요청할 것으로 예측되는 VM 개수
Avg	서비스당 처리한 자원량(VM 개수)의 평균
T_{res}	서비스 응답시간
T_{SLA}	SLA에서 약속된 서비스 응답시간

Equation (1)은 Equation (2)와 (5)에 의하여 다음과 같이 계산된다.

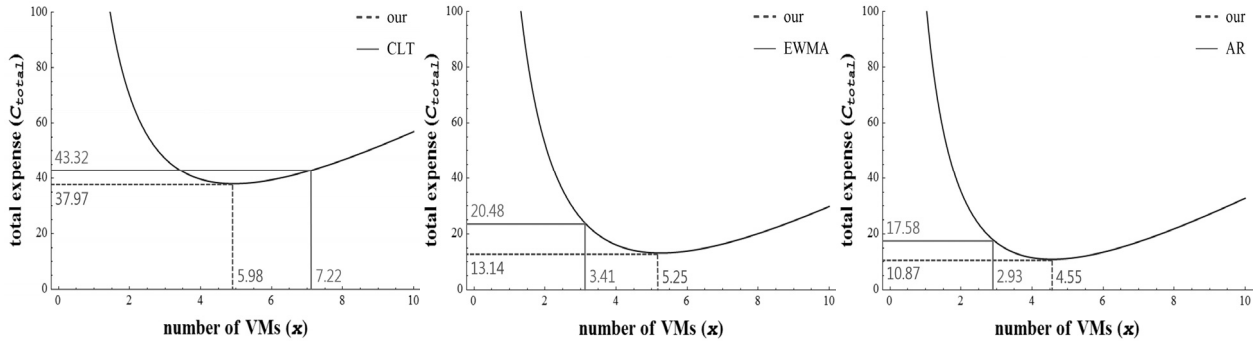
$$\begin{aligned} C_{total} &= (1-P)\{\lambda \cdot C + (x-\lambda) C_{wasted}\} \\ &+ P\left\{x \cdot C + (\lambda - x)\min\left(\frac{\lambda}{|Avg-\lambda|x\, T_{SLA}}, 1\right) C_{penalty}\right\} \end{aligned} \tag{6}$$

서비스 제공자의 총 경비에 대한 기대 값은 다음 서비스에서 요청할 것으로 예측되는 λ의 양에 의해 결정된다. 기존 기법들은 다음 서비스 제공을 위해 준비할 자원의 양을



(A) with CLT (B) with EWMA (C) with AR

Fig. 2. Comparison with our mechanism and other prediction mechanisms



(A) with CLT (B) with EWMA (C) with AR

Fig. 3. Comparison of the total expense at a specific time

λ와 동일하게 결정한다. 이 경우 서비스 제공자의 경비 최소화에 대한 보장이 없다. 따라서 제안 기법은 다른 예측 기법의 결과 값 λ를 기반으로 C_{total}을 최소화하는 자원 제공량 x를 계산한다. Equation (6)을 최소화하기 위하여 $\min(\frac{\lambda}{|Avg-\lambda|xT_{SLA}},1)$을 $\frac{\lambda}{|Avg-\lambda|xT_{SLA}}$로서 계산한다. 결과적으로 Equation (6)는 다음과 같다.

$$C_{total}=(1-P)\{\lambda \cdot C+(x-\lambda)C_{wasted}\} + P\left\{x \cdot C+(\lambda-x)\frac{\lambda}{|Avg-\lambda|xT_{SLA}}C_{penalty}\right\} \quad (7)$$

subject to

$$x,\lambda>0,\ T_{res},T_{SLA}>0$$
$$C,C_{wasted}>0,C_{penalty}>C_{wasted}$$

Equation (7)은 오목 함수(convex function)이고 최솟값을 갖는 부분의 접선의 기울기는 0이다. C_{total}의 최솟값을 계산하기 위하여 Equation (7)을 미분하여 $\frac{dC_{total}}{dx}=0$이 되게 하는 x를 다음과 같이 계산할 수 있다.

$$x=\frac{\sqrt{C_{penalty}P\lambda}}{\sqrt{T_{SLA}|Avg-\lambda|(C_{wasted}+C \cdot P-C_{wasted}P)}} \quad (8)$$

따라서 서비스 제공자는 계산된 x만큼의 자원을 미리 준비하여 총 경비를 최소화할 수 있다.

4. 성능 분석

본 논문에서 제안 기법의 성능을 분석하기 위해 사용자의 요구에 대한 실측 데이터(Intel Netbatch workload archive, 2012년 10~11월까지의 데이터)를 이용하였다[5]. 또한 CLT (Central Limit Theorem), EWMA(Exponentially Weighted Moving Average), AR(Auto Regression)과 같은 예측 기법을 이용하여 λ를 계산하였다. 본 논문에서는 서비스 제공자가 λ를 준비했을 때 서비스 제공자의 총 경비와 제안된 기법의 결과 값 x를 준비했을 때의 총 경비를 비교하여 제안된 기법의 우수성을 증명한다. 실험 환경에서 주요 변수들은 $P=0.3$, $C=6$, $C_{wasted}=8$, $C_{penalty}=10$, $T_{SLA}=0.7$로 설정한다[1].

Fig. 2는 제안 기법을 사용했을 때의 총 경비가 다른 예측 기법을 사용했을 때의 총 경비들보다 작음을 보여준다. Fig. 2A에서 제안 기법의 평균값 26.81은 CLT 기법의 평균값 30.59보다 14% 낮다. Fig. 2B에서 제안 기법의 평균값 16.32는 EWMA 기법의 평균값 19.07보다 16% 낮다. Fig. 2C에서 제안 기법의 평균값 9.46은 AR 기법의 평균값 11.30보다 19% 낮다. Fig. 3은 Fig. 2의 한 구간에서 다른 예측 기법과 제안 기법의 총 경비를 비교하여 보여준다. Fig. 3에서 제안 기법의 결과 값은 다른 기법의 결과 값보다 작고 오목 그래프에서 가장 작은 값을 가진다. 이러한 실험 결과를 바탕으로 본 논문의 제안 기법이 총 경비에 대하여 가장 낮은 기대 값을 갖는 자원의 양을 결정할 수 있음을 알 수 있다.

5. 결 론

본 논문에서는 서비스 제공자의 총 경비에 대한 기댓값을 최소화하여 이득을 최대화하는 VM 프로비저닝 기법을 제안했다. 클라우드 환경에서 사용자에게 QoS를 제공하고 서비스 제공자의 이득을 최대화하기 위해서는 정확하게 자원 요구량을 예측하고 이를 기반으로 자원 준비량을 결정할 수 있는 기법이 필요하다. 하지만 기존의 많은 기법들은 서비스의 자원 요구량을 예측하고, 해당 예측량을 다음 서비스에 제공할 자원량으로 결정한다. 그러나 이들 기법은 사용자에게 QoS는 제공하지만 서비스 제공자의 이득을 최대화하지는 않는다. 따라서 본 논문에서는 오버 프로비저닝과 언더 프로비저닝 상황에서의 경비를 고려하고 다른 예측 기법의 결과 값을 기반으로 서비스 제공자의 총 경비에 대한 기대 값을 최소화 시키는 자원 제공량 결정 기법을 제안하였다. 또한 성능 분석을 통하여 제안된 기법이 다른 예측 기법만을 적용한 결과 보다 총 경비가 더 낮다 것을 증명하였다.

References

[1] J. Almeida, V. Almedia, D. Ardagna, C. Francalanci, and M. Trubian, "Resource management in the autonomic service-oriented architecture," in *Proceedings of the IEEE International Conference on Autonomic Computing*, Dublin, pp.557-568, 2006.

[2] Y. Jiang, C. Perng, T. Li, and R. N. Chang, "Cloud analytics for capacity planning and instant VM provisioning," *The IEEE Transactions on Network and Service Management:* Vol.10, No.3, pp.312-325, 2013.

[3] K. Leonard, "Queueing System Volume 1:Theory," Wiley Interscience, 1975.

[4] P. Athanasions, P. S. Unnikrishna, "Probability Random Variables, and Stochastic processes," Prentice Hall, 2002.

[5] D. G. Feitelson, Pallel workloads archive: Logs [Internet], http://www.cs.huji.ac.il/labs/parallel/workload/

최 영 호

e-mail : ceewoo@suwon.ac.kr

2012년 수원대학교 정보미디어학과(학사)

2014년 수원대학교 컴퓨터학과 석사과정

현 재 경기도 지역협력연구센터 U-City 보안감시 기술협력센터 연구원

관심분야 : Cloud Computing & Resource Allocation

임 유 진

e-mail : yujin@suwon.ac.kr

2000년 숙명여자대학교 전산학과(박사)

2013년 Tohoku University, Dept. of Information Sciences(박사)

현 재 수원대학교 정보미디어학과 교수

관심분야 : Wireless Communication, Cloud Computing