

다중 자원의 성능 분석을 통한 클라우드 가상머신 할당 기법¹⁾

채송화*, 이화민*

*순천향대학교 컴퓨터학과

e-mail:shchae@sch.ac.kr, leehm@sch.ac.kr

Virtual Machine Allocation Scheme of Cloud through Performance Analysis of Multi Resource

Song-Hwa Chae*, HwaMin Lee*

*Dept of Computer Science&Engineering, Soonchunhyang University

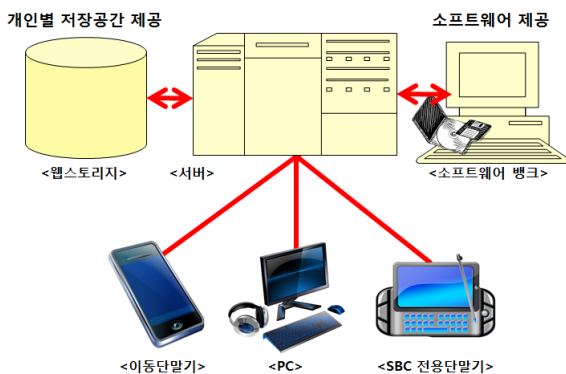
요 약

클라우드 컴퓨팅은 클라우드 서버에서 제공되는 자원을 이용하여 사용자에게 고사양의 컴퓨팅 환경을 제공한다. 클라우드 서비스 환경 구축을 제공하는 유칼립투스에서 사용되는 라운드로빈, 그리디, 파워세이브 등의 가상머신 할당 기법은 자원들을 같은 성능으로 판단하고 사용여부만으로 가상머신을 할당할 노드를 선택하기 때문에 노드의 공정한 사용이 가능하다. 그러나 다른 성능의 자원을 고려하지 않아 자원 사용의 효율성이 결여되었다.

본 논문에서는 가상머신이 최적의 노드에 할당하기 위해 다양한 성능의 노드들을 미리 분석하고 조건에 가장 적합한 노드에 가상머신을 할당하여 사용자에게 제공하는 가상머신 할당 기법을 제안한다. 시스템이 시작되면 노드들의 성능이 분석되어 데이터베이스에 저장되고 이를 바탕으로 가상머신은 노드에 할당된다.

1. 서론

클라우드 컴퓨팅은 사용자의 컴퓨팅 환경에 상관없이 가상화된 클라우드를 이용하여 고사양의 컴퓨팅 환경을 제공하는 서비스로서, CPU, 스토리지 등 직접적인 자원을 제공해주는 IaaS(Infrastructure as a Service), 고사양을 요구하는 애플리케이션이 실행되도록 컴퓨팅 환경을 제공하는 SaaS(Software as a Service), 애플리케이션의 개발을 위한 플랫폼을 제공하는 PaaS(Platform as a Service)의 서비스를 제공한다[1]. (그림1)은 클라우드 컴퓨팅의 개념을 설명한다.



(그림 1) 클라우드 컴퓨팅 개념[1]

1) 본 과제(결과물)는 교육과학기술부의 재원으로 지원을 받아 수행된 산학협력 선도대학(LINC) 육성사업의 연구결과입니다.

념을 설명한다.

클라우드 서비스를 제공하기 위해서 환경 구축을 위한 대표적인 플랫폼 서비스에는 유칼립투스(Eucalyptus), 넘버스(Nimbus), 오픈니블라(OpenNebula) 등이 있으며[2], 각 서비스들은 등록된 클러스터와 노드들을 관리하고 Xen이나 VM ware 등 가상화(Virtualization) 기술을 활용하여 가상머신을 생성, 노드에 할당하여 사용자에게 서비스를 제공한다[3].

유칼립투스는 유칼립투스 시스템에서 제공하는 클라우드 환경 구축 플랫폼으로 가상머신을 효과적으로 노드에 할당을 위한 스케줄러로 라운드로빈(Round-Robin)[4], 그리디(Greedy)[5], 파워세이브(Power-Save)[6] 등을 제공한다[7]. 라운드로빈은 가상머신의 할당이 가능한 노드들을 번갈아가며 사용하는 스케줄러로 다른 노드가 발견되기 전까지 계속 해당 노드에서 가상머신을 실행한다. 한 노드에 집중적인 가상머신 할당 없이 공정한 할당이 가능하지만 가상머신 할당에 우선순위가 없어 단계적 서비스 제공이 어렵다. 그리디는 모든 경우의 수를 부분적으로 나눠 비교하여 맞이하는 경우의 수 중 최적의 경우를 선택해가는 과정을 거쳐 하나의 경우를 선택하는 방식으로, 유칼립투스에서는 가상머신을 실행할 수 있는 노드를 발견하면 가장 첫 번째 노드가 선택된다. 파워세이브는 가상머신이 실행되지 않을 경우 노드를 잠들게 하고, 자원이 요구될 때 다시 각성시키고, 가상머신은 자고 있던 기계 중

가장 먼저 깐 컴퓨터에 배치된다.

유칼립투스에서 제공하는 스케줄러는 노드의 상태를 기반으로 가상머신을 할당할 노드를 선택함으로써 가상머신이 모든 노드를 공정하게 사용할 수 있도록 하며 파워셰이브의 경우 전력을 최소화함으로써 관리비용을 줄일 수 있다. 그러나 노드의 성능 차이가 클 경우 고성능의 노드와 저성능의 노드가 가상머신 할당에 같은 기회를 가져 단편화 등의 문제가 발생되고 고성능의 노드를 효율적으로 사용하는데 제약이 생긴다.

본 논문에서는 노드들의 성능 차이와 사용자 레벨을 스케줄러에 반영하여 가상머신을 할당할 최적의 노드를 선택하는 자원의 성능 분석을 통한 가상머신 할당 기법을 제안한다. 2장에서는 효율적인 가상머신 할당을 위한 기존의 연구를 살펴보고, 3장에서는 노드들의 성능을 분석하기 위한 방법을, 4장에서는 본 논문에서 제안하는 가상머신 할당 기법을 소개하고 다른 스케줄러와의 비교 분석 결과를, 5장에서는 결론을 맺는다.

2. 관련 연구

연구 [8, 9]은 동적 가상머신 할당을 위해 자원 관리 컨트롤러를 가상머신에서 사용하는 자원량과 응용 프로그램에서 요구하는 자원량을 파악하는 지역 컨트롤러와 자원 상황이 파악되면 가상머신 할당의 요청에 응하는 전역 컨트롤러로 나누었다.

전자 공학을 이용한 [8]과 요구하는 자원량 파악을 위해 담금질 기법을 사용하는 [9]이 제시되었다. [8]과 [9]에서는 지역 컨트롤러를 구현하는데 환경에 영향을 받지 않지만, 가상머신을 할당하는데 생기는 지연 등을 방지 및 해결하는 방법이 제안되지 않았다.

수정된 SEDF CPU 스케줄러를 제안한 연구 [10]은 Xen의 SEDF(Simple Early Deadline First) 스케줄러를 기반으로 가상머신은 반환한 CPU가 다른 CPU가 가진 대기중인 가상머신을 대신 처리하여 지연을 줄이고자 하였다.

CPU 사용율과 사용 여부를 데이터베이스에 기록하여 최적의 노드를 추출하는 연구 [11]는 사용율이 낮은 CPU의 노드에 가상머신을 할당하여 부하를 분산시키고자 하였다. 두 개의 레이어를 사용하여 과부하를 줄이려고 하였지만 가상머신이 사용하는 자원량과 노드의 성능의 우선순위 파악이 불가능하다.

3. 노드의 성능 분석

본 논문에서 제안하는 자원의 성능 분석을 통한 가상머신 할당 기법은 시스템 시작될 때 노드들의 연산 능력 평가를 위한 기준을 기반으로 성능을 평가한다. 연산 능력은 LU Decomposition[12]을 이용한 행렬의 역행렬을 구하는데 소요되는 시간으로 CPU의 처리속도와 메모리 적재 능력을 평가하여 노드의 성능을 기준 짓는데 사용한다.

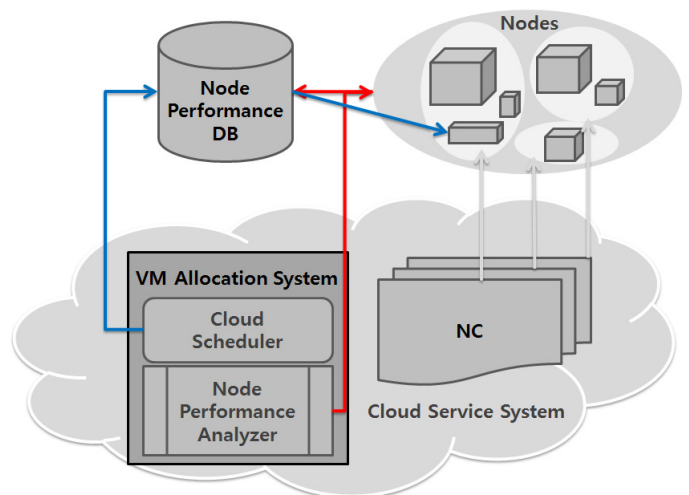
LU Decomposition은 행렬을 하 삼각행렬 L과 상 삼각

행렬 U로 분해하여 $LU = A(n \text{ by } b)$ 가 되는 때를 만드는 것을 말하며 $Ax = b$ 의 선형계로 표시한다. LU Decomposition을 이용해 역행렬을 구하기 위해서는 하나의 행렬을 두 개의 삼각행렬로 분해하고 Gaussian Elimination을 이용하여 계산한다. LU Decomposition의 비용이 적게 들고 간단하며, 하 삼각행렬 L과 상 삼각행렬 U는 다음과 같은 형태를 가진다.

$$U = \begin{bmatrix} u_{11} & \dots & u_{1n} \\ 0 & \ddots & \vdots \\ \vdots & & \\ 0 & \dots & u_{mn} \end{bmatrix} \quad L = \begin{bmatrix} 1 & 0 & \dots & 0 \\ m_{21} & \ddots & & \\ \vdots & \ddots & \ddots & \\ m_{n1} & \dots & m_{n,n-1} & 1 \end{bmatrix}$$

4. 다중 자원의 성능 분석을 통한 클라우드 가상머신 할당 기법

본 논문에서 제안하는 자원의 성능 분석을 통한 가상머신 할당 기법은 노드 컨트롤러에 등록된 노드들의 성능 분석 결과를 바탕으로 자원을 요구하는 가상머신을 최적의 환경을 가진 노드를 선택하여 가상머신을 할당한다. 본 논문에서 제안하는 가상머신 할당 기법의 시스템은 (그림 2)와 같다.



(그림 2) 자원의 성능 분석을 통한 가상머신 할당 기법

Virtual Machine Allocation System은 Node Performance Analyzer와 Cloud Scheduler를 갖는다. Node Performance Analyzer는 시스템이 시작할 때와 노드의 정보가 변경되었을 때 새롭게 노드들의 성능을 분석한다. 분석된 정보는 Node Performance DB에 저장된다.

클라우드 서비스 시스템이 시작하면 가상머신 할당 시스템에서는 (표 1)의 Node Performance Analysis 메소드를 호출한다. Node Performance Analysis 메소드는 노드들의 성능을 평가하고 DB에 저장하는 기능을 한다. 전체 노드의 개수를 N이라고 했을 때 노드 i의 Operation_Processing() 처리 속도 α 를 메모리와 스토리지량과 처리 속도 β , 네트워크 상태 γ 에 각각 0.4, 0.4, 0.2의 가중치가 부여된 노드 평가 결과 Perfor_Anal()의

```

[Node Performance Analysis]
for ( i = 1 to N )
{
    Start_Time = Times();
    result = Operation_Processing(i);
    Finish_Time = Times();
    Processing_Time = Finish_time
                    - Start_Time
    i.result = Perfor_Anal(Processing_Time);
    DB.add(i.result);
}
    
```

(표 1) Node Performance Analysis 알고리즘 반환값을 DB에 저장한다. Operation_Processing()에서는 3장에서 언급한 LU Decomposition을 이용한 행렬의 역행렬을 구하는 방법으로 CPU의 처리 속도와 메모리 적재능력을 평가한다.

Cloud Scheduler는 자원의 요청이 들어오면 Node Performance DB에 저장된 정보와 노드들의 현재 상태를 바탕으로 요청된 자원을 효율적으로 제공할 수 있는 노드를 선택해 가상머신 할당 요청을 한다.

```

[Node Information Search]
for ( i = 1 to N )
{
    if N's Performance
        >= Request Resource than
        ANList.add()
}
Sort ANList
return ANList
    
```

(표 2) Node Information Search 알고리즘

자원의 요청이 들어오면 (표 2)의 Node Information Search 메소드를 통하여 요구된 자원량을 수락할 수 있는 충분한 자원량을 가진 노드를 찾아 가상머신의 할당이 가능한 노드 리스트 ANList에 추가한다. N만큼 반복이 끝난 후 ANList를 정렬하여 반환한다.

```

[Node Select]
for ( i = 1 to M )
{
    if i.result >= ANode.result than
        ANode = i;
}
    
```

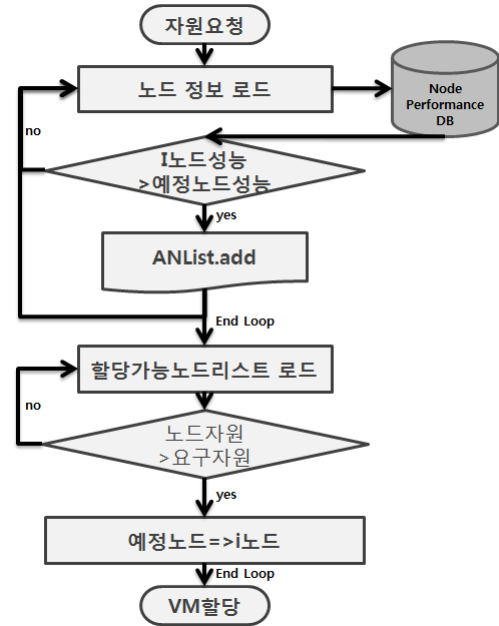
(표 3) Node Select 알고리즘

반환된 ANList는 (표 3)의 Node Select 메소드에서 받아 노드의 성능 분석 결과를 반영한 최적의 노드를 선택한다. 최적의 노드를 찾기 위해 할당 가능한 노드 개수 M만큼 반복한다. 임의 노드 i의 성능 평가 결과가 할당이 예정된 노드 ANode의 성능 평가 결과보다 좋으면 할당 예정 노드를 i로 변경한다.

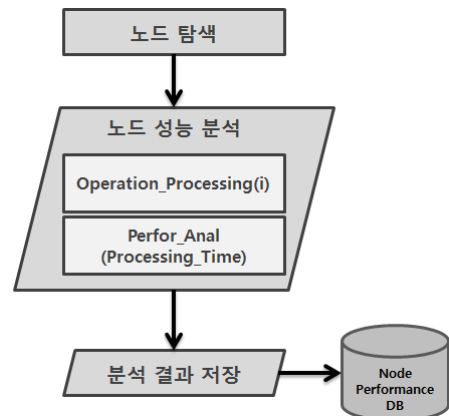
위의 세 알고리즘을 활용한 자원의 성능 분석을 통한 가상머신 할당 기법의 흐름도는 (그림 3,4)과 같다.

(그림 3)은 클라우드 서비스 시스템이 시작했을 때의 동작이다. Node Performance Analyzer에서 Node Performance Analysis 메소드를 활용하여 노드의 성능을 분석하고 분석 결과를 DB에 저장한다.

(그림 4)는 자원의 요청이 들어왔을 때 Cloud Scheduler에서 Node Information Search 메소드와 Node Select 메소드를 활용하여 노드의 성능을 반영한 가상머신 할당을 유도한다.



(그림 3) 자원의 성능 분석을 통한 가상머신 할당 기법 흐름도 1



(그림 4) 자원의 성능 분석을 통한 가상머신 할당 기법 흐름도 2

5. 결론 및 향후 연구과제

클라우드 컴퓨팅은 클라우드 서버의 자원을 활용하여 사용자의 환경에 제약을 받지 않고 고사양의 컴퓨팅 환경

에서 작업할 수 있도록 제공해주는 서비스이다. 본 논문은 고성능의 노드와 저성능의 노드를 성능에 맞게 효율적으로 사용할 수 있으며 사용자 비용에 맞는 서비스 제공도 할 수 있도록 노드들의 성능을 현실적으로 반영하였을 때 최적의 노드에 가상머신을 할당하는 기법을 제안하였다.

향후 연구과제로 QoS(Quality of Service)를 반영한 사용자 요구사항 충족에 대한 구체적인 연구와 노드의 성능을 분석하는데 있어 추가적으로 살펴봐야할 그래픽 처리 능력, 네트워크의 능력을 객관적으로 평가할 수 있는 네트워크 분석 등이 적용된 알고리즘의 연구가 필요하다.

참고문헌

[1] 클라우드 컴퓨팅 동향, 정보통신산업진흥원 주간기술동향, 2010.

[2] Sempolinski, P., Thain, D., A Comparison and Critique of Eucalyptus, OpenNebula and Nimbus, Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on, pp.417-426, 2010.

[3] 김창환, 클라우드 컴퓨팅 기술 시장 동향, 정보통신산업진흥원, 2010.

[4] Johannes Fürnkranz, Round robin classification, The Journal of Machine Learning Research, Vol.2, pp.721-747, 2002

[5] Thomas H. Cormen, Charles E. Leiserson, Ronald Lorin Rivest, Clifford Stein, "Introduction to Algorithms - 16. Greedy Algorithm" MIT Press and McGraw-Hill, 2ng edition, 2001.

[6] P Patel, AK Singh, A Survey On Resource Allocation Algorithms In Cloud Computing Environment : Golden Research Thoughts, 2012.

[7] Ching-Chi Lin, Pangfeng Liu, Jan-Jan Wu, Energy-Aware Virtual Machine Dynamic Provision and Scheduling for Cloud Computing, Cloud Computing (CLOUD), 2011 IEEE International Conference on, pp.736-737, 2011.

[8] Xu, J., Zhao, M., Fortes, J., Carpenter, R., and Yousif, M., Autonomic Resource Management in Virtualized Data Centers Using Fuzzy Logic-Based Approaches, P_ceedings of Cluster Computing, Vol.11 No.3, pp.213-227, 2008.

[9] X. Zhu, D. Young, B. J. Watson, Z. Wang, J. Rolia, S. Sigal, B. McKee, C. Hyser, D. Gmach, R. Gardner, T. Christian, and L. Cherkasova, 1000 Islands: Integrated Capacity and Workload Management for the Next Generation Data Center, Proceedings of International Conference on Auto-nomic Computing (ICAC '08), 2008.

[10] 양은지, 최현식, 한세영, 박성용, Xen 환경에서 스케

줄링 지연을 고려한 가상머신 우선순위 할당 기법, 정보과학회논문지 : 시스템 및 이론 제 27권, pp.195-355, 2010.

[11] 클라우드 컴퓨팅에서 CPU 사용률을 고려한 가상머신 할당 기법, 한국해양정보통신학회논문지 Vol.15, No.3, , 2011.

[12] 김은재, 인영숙, 송기용, 2D Mesh구조에서의 알고리즘 기반 결합허용 IU 분해, 한국정보과학회 가을 학술발표논문집 Vol.24, No.2, 1997.

[13] 채송화, 이화민, 클라우드에서 다중 자원의 성능을 고려한 가상머신 할당 기법, 한국정보처리학회 추계학술발표대회 논문집 Vol.19, No.1, 2012.