

KỸ THUẬT CUNG CẤP TÀI NGUYÊN CHO LỚP HẠ TẦNG (IAAS)

TECHNICAL RESOURCES PROVISION FOR INFRASTRUCTURE

Nguyễn Hà Huy Cường¹, Lê Văn Sơn²

¹Trường Đại học Quảng Nam; Email: nguyenhahuycuong@gmail.com

²Trường Đại học Sư Phạm, Đại học Đà Nẵng; Email: levansupham2004@yahoo.com

Tóm tắt - Bài báo này chúng tôi nghiên cứu các vấn đề về tối ưu các chức năng tiện ích của điện toán đám mây, tùy thuộc vào nguồn tài nguyên giới hạn tại tầng tài nguyên IaaS. Tối ưu cung cấp tài nguyên tại lớp hạ tầng là vấn đề có thể được chia ra thành ba bài toán: Thứ nhất, đó là bài toán cung cấp tài nguyên điện toán đám mây; Thứ hai, là vấn đề tối ưu về thời gian cho việc phân bổ lại các máy chủ ảo ở tại các trung tâm dữ liệu khác nhau, và cuối cùng là sử dụng tối đa được chất lượng cung cấp dịch vụ QoS. Thuật toán đề xuất đã phân tích được các vấn đề tổng quát về cung cấp tài nguyên cho máy chủ ảo tại lớp hạ tầng. Các thí nghiệm được tiến hành để kiểm tra hiệu quả của thuật toán với việc thay đổi nhiều thông số từ môi trường nghiên cứu thực nghiệm. Các thí nghiệm cũng so sánh hiệu suất của phương pháp đề xuất với các thuật toán liên quan khác.

Từ khóa - điện toán đám mây; hệ phân tán; tài nguyên; lớp hạ tầng dịch vụ; hệ thống máy chủ ảo.

Abstract - In this paper, we study the problem of optimizing the utility function of the cloud, depending on the limited resources at floor IAAS resources. Optimal resource providers in the infrastructure layer problems can be divided into three problems: the problem of providing cloud computing resources. It is a matter of time for optimal reallocation virtual servers at different data centers, and finally the use of maximum quality of service QoS provided. Algorithm proposed analyzes the general problem of providing resources for the virtual server class infrastructure. The experiments were conducted to test the effectiveness of the algorithm with changing environmental parameters from experimental studies. The experiments also compare the performance of the proposed method with other related algorithms.

Key words - cloud computing; distributed system; resource; IaaS; virtual server system

1. Đặt vấn đề

Vào những năm 50 của thế kỉ XX nhiều nhà khoa học đã nỗ lực để tìm ra các giải pháp cho vấn đề “cung cấp tài nguyên” và cũng từ đó các hệ thống máy tính mainframe bắt đầu chính thức được dùng để chia sẻ tài nguyên cho nhiều người sử dụng. Phương pháp cấp phát tài nguyên điện toán là một trong những vấn đề thực tế kinh điển nhất trong khoa học máy tính và đã có rất nhiều nghiên cứu để đưa ra giải pháp tối ưu nhất. Trong công trình nghiên cứu của Vouk [5], tác giả đã phân loại các hệ thống điện toán đám mây và phân tích việc thực hiện quản lý tài nguyên điện toán đám mây. Tác giả Srikantaiah và cộng sự của mình đã nghiên cứu vấn đề lập kế hoạch yêu cầu nhiều tầng ứng dụng web trong các hệ thống không đồng nhất ảo hóa để giảm thiểu tiêu thụ năng lượng trong khi đáp ứng yêu cầu hiệu suất. Họ đề xuất một heuristic cho vấn đề đóng gói đa chiều như một thuật toán giải quyết khối lượng lớn công việc và sau đó là hợp nhất khối lượng công việc đó lại [6]. Trong khi đó nhóm nghiên cứu của tác giả Garg đã đề xuất các chính sách lập kế hoạch gần tối ưu để khai thác sự không đồng nhất của nhiều trung tâm dữ liệu cho một nhà cung cấp điện toán đám mây [7]. Họ cũng xem xét một số yếu tố hiệu quả về năng lượng (chẳng hạn như chi phí năng lượng, tỷ lệ khí thải carbon, khối lượng công việc và sự khai thác hiệu quả năng lượng của CPU), mà sự thay đổi qua lại giữa các trung tâm dữ liệu khác nhau không phụ thuộc vào vị trí của họ. Nhóm của tác giả Warneke cũng đã thảo luận về những thách thức và cơ hội cho hiệu quả xử lý dữ liệu song song trong các môi trường điện toán đám mây và đưa ra khung xử lý dữ liệu để khai thác dự phòng tài nguyên động được cung cấp bởi IaaS đám mây [9]. Nhóm nghiên cứu của tác giả Wu và cộng sự của mình đã đề xuất một thuật toán phân bổ tài nguyên cho các nhà cung cấp SaaS - những người muốn giảm thiểu chi phí cơ sở hạ tầng [10]. Trong thuật toán đề xuất của họ, nhờ việc xem xét các

nhà cung cấp dịch vụ SaaS quản lý sự thay đổi năng động của khách hàng, nhóm tác giả đã lập được một biểu đồ các yêu cầu của khách hàng với các thông số phụ thuộc vào tùy mức độ của cơ sở hạ tầng và xử lý sự không đồng nhất của các máy ảo. Các bài báo mà nhóm nghiên cứu chúng tôi đã công bố trước đây tại các hội nghị và tạp chí khoa học [1,2,3], có những vấn đề về giải pháp cung cấp tài nguyên hiệu quả dựa trên cơ sở của các thuật toán phát hiện, ngăn chặn và tránh được những bế tắc tài nguyên trong cung cấp tại hệ thống máy chủ ảo. Qua việc tiếp cận nghiên cứu, phân tích so sánh và đánh giá, chúng tôi nhận thấy rằng các thuật toán phát hiện bế tắc dựa trên nền tảng của các thuật toán phát hiện bế tắc trong cung cấp tài nguyên ở trong hệ phân tán vẫn còn nguyên vẹn tính thời sự của nó. Và hướng tiếp cận nghiên cứu cải tiến dựa trên nền các thuật toán của các tác giả nghiên cứu trước đây vẫn mang lại một số kết quả đáng tin cậy. Trong bài báo này chúng tôi đề xuất thuật toán cung cấp tài nguyên phân tán hiệu quả cho các máy chủ ảo tại lớp hạ tầng IaaS.

2. Đề xuất thuật toán cung cấp tài nguyên cho hệ thống hạ tầng máy chủ ảo

Hiện nay, cung cấp tài nguyên cho máy chủ ảo VS dựa vào tài nguyên của máy chủ vật lý tại thời điểm cụ thể là vấn đề còn khá mới mẻ. Để xác định được trung tâm dữ liệu nào có khả năng phân phối tài nguyên cho máy chủ ảo VS dựa trên thông tin yêu cầu của một hợp đồng (lease), thì thuật toán đề xuất sẽ tiến hành theo các bước sau:

Thuật toán: Cung cấp máy chủ ảo VS_j bất kỳ từ hợp đồng (lease) trong môi trường cục bộ.

Input: Ký hiệu P là tập hợp các tiến trình yêu cầu tài nguyên phân cứng gồm: $P_{i,j(cpu)}^*$, $P_{i,j(ram)}^*$, $P_{i,j(HDD)}^*$ tới lớp hạ tầng IaaS.

Output: tài nguyên được cung cấp một cách hiệu quả r_j^{cpu} , r_j^{ram} , r_j^{HDD}

Bảng 1. Bảng các kí hiệu sử dụng trong thuật toán

STT	Kí hiệu	Ý nghĩa các kí hiệu
1	$x_i^{j(cpu)*}$	CPU yêu cầu để tạo máy ảo VS_i từ IaaS cung cấp j
2	$x_i^{j(ram)*}$	RAM yêu cầu để tạo máy ảo VS_i từ IaaS cung cấp j
3	$x_i^{j(HDD)*}$	HDD yêu cầu để tạo máy ảo VS_i từ IaaS cung cấp j
4	$C_j^{cpu}, C_j^{ram}, C_j^{HDD}$	Khả năng tối đa của CPU, RAM, HDD lớp IaaS cung cấp j
5	$r_j^{cpu}, r_j^{ram}, r_j^{HDD}$	Giá trị tài nguyên mới của IaaS cung cấp j

Các bước thực hiện thuật toán như sau:

Bước 1: Tính toán hiệu quả cung cấp tài nguyên cho máy chủ tùy theo nhu cầu của khách hàng.

$$x_i^{j(cpu)*}, x_i^{j(ram)*}, x_i^{j(HDD)*} = \text{Max}\{U_{IaaS}\};$$

Bước 2: Tính toán lại giá trị tài nguyên tại thời điểm t
If

$$C_j^{cpu} \geq \sum_i x_i^{j(cpu)}, C_j^{ram} \geq \sum_i x_i^{j(ram)}, C_j^{HDD} \geq \sum_i x_i^{j(HDD)}$$

// Kiểm tra khả năng tài nguyên tại các trung tâm cung cấp tài nguyên có lớn hơn nhu cầu của khách hàng tại thời điểm t. Với $C_j^{cpu}, C_j^{ram}, C_j^{HDD}$ là khả năng tài nguyên lớn nhất của lớp hạ tầng.

Then

$$r_j^{cpu(n+1)} = \max\left\{\varepsilon, r_j^{cpu(n)} + \eta\left(\sum_i x_i^{j(cpu)} - C_j^{cpu}\right)\right\};$$

// Kiểm tra tài nguyên hiệu quả cung cấp cho phần cứng CPU. Trong đó ε, η là những hằng số.

$$r_j^{ram(n+1)} = \max\left\{\varepsilon, r_j^{ram(n)} + \eta\left(\sum_i x_i^{j(ram)} - C_j^{ram}\right)\right\};$$

// Kiểm tra tài nguyên hiệu quả cung cấp cho phần cứng RAM.

$$r_j^{hdd(n+1)} = \max\left\{\varepsilon, r_j^{hdd(n)} + \eta\left(\sum_i x_i^{j(hdd)} - C_j^{hdd}\right)\right\};$$

// Kiểm tra tài nguyên hiệu quả cung cấp cho phần cứng HDD.

Bước 3: Return giá trị tài nguyên cho thuê mới tại thời điểm t cho tất cả các hợp đồng.

$$r_j^{cpu(n+1)}; r_j^{ram(n+1)}; r_j^{hdd(n+1)}$$

Else Return về câu lệnh liền kề câu lệnh gọi;

Ví dụ 1: Khi có một hợp đồng thuê tài nguyên để tạo máy ảo với yêu cầu tài nguyên CPU; RAM và ổ cứng HDD.

Trường hợp thứ nhất khả năng của tối đa của lớp IaaS là:

$$C_j^{cpu} = 4\text{GHz}, C_j^{ram} = 2048\text{MB}, C_j^{HDD} = 40\text{GB}$$

CPU yêu cầu để tạo máy ảo là $x_i^{j(cpu)*} = 1\text{GHz}$; với ram

là $x_i^{j(ram)*} = 512\text{MB}$ và HDD là $x_i^{j(HDD)*} = 10\text{GB}$. Với $j=5$ thì ta có $\sum_i x_i^{5(cpu)} = 5\text{GHz}$; $\sum_i x_i^{5(ram)} = 2560\text{MB}$; $\sum_i x_i^{5(HDD)} = 50\text{GB}$. Tổng của 5 yêu cầu tạo máy ảo lớn hơn khả năng của tài nguyên lớp IaaS cung cấp cho $j=5$ khi có hợp đồng khác thì thông báo Null.

Trường hợp thứ 2 tăng lên gấp đôi khả năng của lớp hạ tầng IaaS.

$$C_j^{cpu} = 8\text{GHz}, C_j^{ram} = 4096\text{MB}, C_j^{HDD} = 80\text{GB}$$

Cũng như trong trường hợp thứ nhất CPU yêu cầu để tạo máy ảo là $x_i^{j(cpu)*} = 1\text{GHz}$; với ram là $x_i^{j(ram)*} = 512\text{MB}$

và HDD là $x_i^{j(HDD)*} = 10\text{GB}$. Với $j=5$ thì ta có $\sum_i x_i^{5(cpu)} = 5\text{GHz}$; $\sum_i x_i^{5(ram)} = 2560\text{MB}$;

$\sum_i x_i^{5(HDD)} = 50\text{GB}$. Tổng của 5 yêu cầu tạo máy ảo không

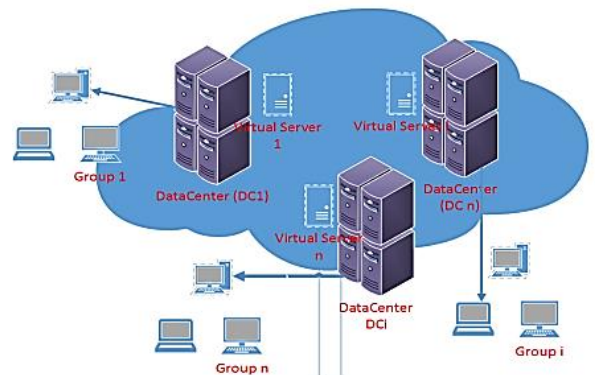
lớn hơn khả năng của tài nguyên lớp IaaS cung cấp cho $j=5$ thì trong trường hợp này giá trị tài nguyên mới có thể cho thuê tại thời điểm t là giả sử $\varepsilon_{cpu} = C_j^{cpu} - \sum_i x_i^{j(cpu)}$ và $\eta=1$ tương tự cho ε_{ram} ; ε_{HDD} .

$$\text{Thì giá trị } r_j^{cpu(n+1)} = 3\text{GHz}; r_j^{ram(n+1)} = 1536\text{MB}$$

$$r_j^{HDD(n+1)} = 30\text{GB}$$

3. Mô phỏng và bàn luận kết quả nghiên cứu

Trong báo cáo này, chúng tôi tiến hành thử nghiệm thuật toán với 3 kịch bản. Các kịch bản được thử nghiệm trên máy tính Intel Core™i3-2350M CPU(2.3 GHz, và 4GB Ram). Chúng tôi sử dụng phần mềm mô phỏng CloudSim (Cloud Simulation) Toolkit for Modeling and Simulation, phiên bản 3.0.3 năm 2009 của trường Đại học Melbourne, Australia để thử nghiệm đánh giá. Trong các kịch bản của mình các máy ảo VM được tạo với cấu hình như sau: CPU = 1Ghz; HDD = 10GB; RAM = 512MB;



Hình 1. Mô hình mạng mô phỏng máy chủ ảo Virtual Server trong Cloud Computing trong ca thử nghiệm trong môi trường phân tán

Trong kịch bản thử nghiệm đầu tiên, chúng tôi yêu cầu tạo 4 máy ảo trên một trung tâm dữ liệu với kết quả thu được như hình bên dưới:

===== OUTPUT =====

STT	Cloudlet ID	STATUS	Data center ID	VM ID	Time	Start Time	Finish Time
0	3	THÀNH CÔNG	2	4	62.5	0.1	62.6
1	2	THÀNH CÔNG	2	3	83.33	0.1	83.43
2	1	THÀNH CÔNG	2	2	125	0.1	125.1
3	0	THÀNH CÔNG	2	1	250	0.1	250.1

Trong kịch bản này máy ảo có VM ID 4 có tổng thời gian hoàn thành là 62.5s, trong khi máy ảo có VM ID 1 có tổng thời gian hoàn thành là 250s.

Kịch bản thử nghiệm thứ 2, chúng tôi yêu cầu tạo 6 máy ảo trên hai trung tâm dữ liệu với kết quả thu được như hình bên dưới:

===== OUTPUT =====

STT	Cloudlet ID	STATUS	Data center ID	VM ID	Time	Start Time	Finish Time
0	4	THÀNH CÔNG	3	5	50	0.2	50.2
1	3	THÀNH CÔNG	2	4	62.5	0.2	62.7
2	2	THÀNH CÔNG	2	3	83.33	0.2	83.53
3	5	THÀNH CÔNG	3	6	83.33	0.2	83.53
4	1	THÀNH CÔNG	2	2	125	0.2	125.2
5	0	THÀNH CÔNG	2	1	500	0.2	500.2

Kịch bản này máy ảo có VM ID 5 có tổng thời gian hoàn thành là 50 s, trong khi máy ảo có VM ID 1 có tổng thời gian hoàn thành là 500s.

Trong kịch bản thử nghiệm thứ 3, chúng tôi yêu cầu tạo 7 máy ảo trên bốn trung tâm dữ liệu với kết quả thu được như hình bên dưới:

===== OUTPUT =====

STT	Cloudlet ID	STATUS	Data center ID	VM ID	Time	Start Time	Finish Time
0	4	THÀNH CÔNG	3	5	50	0.3	50.3
1	3	THÀNH CÔNG	2	4	62.5	0.3	62.8
2	2	THÀNH CÔNG	2	3	83.33	0.3	83.63
3	5	THÀNH CÔNG	3	6	83.33	0.3	83.63
4	6	THÀNH CÔNG	4	7	83.33	0.3	83.63
5	1	THÀNH CÔNG	2	2	125	0.3	125.3
6	0	THÀNH CÔNG	2	1	500	0.3	500.3

Trong kịch bản thứ 3 này máy ảo có VM ID 5 có tổng thời gian hoàn thành thấp nhất là 50s, trong khi máy ảo có VM ID 1 có tổng thời gian hoàn thành cao nhất là 500s. Qua các kịch bản thử nghiệm ta thấy rằng trung tâm dữ liệu 2 là trung tâm tạo được 4 máy ảo. Trong ca thử nghiệm 2, sau khi 4 máy ảo được tạo ra ở trung tâm thứ 2, tiếp tới trung tâm thứ 3 tạo ra được 2 máy ảo. Trong ca thử nghiệm thứ 3, trung tâm 2 cũng tạo được 4, sau đó trung tâm 3 tạo được 2, trung tâm 4 tạo được 1. Với khả năng có hạn của máy laptop chúng tôi thử nghiệm thì có thể tạo tối đa được 8 máy ảo với 4 trung tâm dữ liệu.

4. Kết luận

Qua kết quả mô phỏng và phân tích đánh giá ưu, khuyết điểm của thuật toán đề xuất này chúng ta nhận thấy rằng hiện tại vấn đề nghiên cứu chính sách cung cấp tài nguyên hiệu quả tại mỗi trung tâm dữ liệu cần có một thông tin về tài nguyên sẵn có, để cung cấp cho mỗi hợp đồng thuê tài nguyên tạo VS. Đồng thời để các nhà môi giới sẽ lựa chọn được các trung tâm dữ liệu phù hợp với yêu cầu tài nguyên tạo ra máy chủ ảo cho mình.

Thuật toán được nêu ở trên xuất phát từ cơ sở cùng một nguyên lí. Đó là sự thiếu chắc chắn của trạng thái các máy

chủ ảo ở xa phát sinh vấn đề lưu trữ một “giới hạn an toàn” nhất định. Điều đó lại ngăn cản việc phát hiện giải quyết bế tắc trong một số trường hợp do thiếu sự chắc chắn các máy chủ ảo ở xa trung tâm dữ liệu.

Kết quả nghiên cứu này có thể hình thành phương pháp cập nhật dữ liệu trong hệ thống thông tin dùng chung thông qua mạng cục bộ trong phạm vi hẹp. Từ đó ta có thể phát triển và áp dụng cho mạng Internet trong môi trường máy chủ ảo ĐTĐM.

Hiện nay trên thế giới tại các trung tâm dữ liệu sử dụng kỹ thuật máy chủ ảo ngày một gia tăng cả trong nghiên cứu và triển khai ứng dụng dịch vụ Điện toán Đám mây. Vì vậy việc nghiên cứu và phát triển các giải thuật cung cấp tài nguyên trong nền tảng ảo hóa là một trong vấn đề nổi cộm lên trên hết. Trong những nghiên cứu tiếp theo chúng tôi sẽ đi sâu nghiên cứu ứng dụng và chứng minh tính chính xác của thuật toán này. Ngoài ra, chúng tôi đề xuất thêm các biện pháp phòng tránh và ngăn chặn bế tắc xảy ra.

TÀI LIỆU THAM KHẢO

- [1] Lê Văn Sơn, Nguyễn Hà Huy Cường, “Giải pháp kỹ thuật cung cấp tài nguyên thông tin dùng chung trong điện toán đám mây”, Kỷ yếu hội thảo một số vấn đề chọn lọc của Công nghệ thông tin và truyền thông, 2012, page 587 – 605.
- [2] Nguyễn Hà Huy Cường, Lê Văn Sơn, Nguyễn Thanh Thủy, “ Ứng dụng thuật toán Kshemkalyani – Singhal phát hiện bế tắc trong cung cấp tài nguyên phân tán cho hệ thống máy chủ ảo”, Kỷ yếu hội thảo Fair VI, 2013, page 602 – 608.
- [3] Nguyễn Hà Huy Cường, “Nghiên cứu giải pháp kỹ thuật ngăn chặn bế tắc trong cung cấp tài nguyên phân tán cho hệ thống máy chủ ảo”, Tạp chí khoa học và Công nghệ, Volume 50, Number 3E, 2012, page 1324 – 1331.
- [4] D. P. Mitchell and M. J. Merritt, “A distributed algorithm for deadlock detection and resolution,” in Proc.ACM Symposium on Principles of Distributed Computing, 1984, pp. 282–284.
- [5] E.Knapp. (1987), Deadlock Detection in Distributed Database Systems, ACM Computing Surveys, Vol.19, No. 4 pp.303-327.
- [6] M.Singhal.(1989), Deadlock detection in distributed systems. IEEE Computer, Vol.22, pp. 37–48.
- [7] A.D.Kshemkalyani, and M.Singhal. (1999), A One-Phase Algorithm to Detect Distributed Deadlocks in Replicated Databases, IEEE Trans. Knowledge and Data Eng., vol. 11, No. 6, pp. 880-895.
- [8] W. Voorsluys, S. Garg, and R. Buyya, “Provisioning spot market cloud resources to create cost-effective virtual clusters,” in Proceedings of the 11th International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP). Los Alamitos, CA, USA: IEEE Comput. Soc., 2011.
- [9] R. Buyya, R. Ranjan, and R. Calheiros, “Modeling and simulation of scalable cloud computing environments and the cloudsim toolkit: Challenges and opportunities,” in Proceeding of the 7th International Conference on High Performance Computing & Simulation (HPCS). Los Alamitos, CA, USA: IEEE Comput. Soc., 2009, pp. 1–11.
- [10] D. Feitelson, “Parallel workloads archive,” <http://www.cs.huji.ac.il/labs/parallel/workload>.
- [11] D. Tsafir, Y. Etsion, and D. G. Feitelson, “Modeling User Runtime Estimates,” in In Processing of the 11th Workshop on Job Scheduling Strategies for Parallel Processing (JSSPP). Springer-Verlag, 2005, pp.1–35.
- [12] Binildas CA, Malhar Barai, Vincenzo Caselli, *Service Oriented Architecture with Java (Using SOA and Web Services to build powefull Java applications)*, PACKT Publishing, 2008.
- [13] Ian Foster, Yong Zhao, Ioan Raicu, Shiyong Lu, *Cloud Computing and Grid Computing 360-Degree Compared*, Grid Computing Environments Workshop, 2008.
- [14] Rajkumar Buyya, Chee Shin Yeo, and Srikumar Venugopal,

- Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities*, International Conference on High Performance Computing, 2008.
- [15] M. Andreolini, S. Casolari, M. Colajani, and M. Messori, “Dynamic load management of virtual machines in a cloud architectures,” in CLOUDCOMP, 2009.
- [16] D. Prangchumpol, S. Sanguansintukul, and P. Tantasanaw, “Analyzing User Behavior from Server Logs for Improved Virtualization Management”, 2009.
- [17] M. Stillwell, D. Schanzenbach, F. Vivien, and H. Casanova, “Resource allocation algorithms for virtualized service hosting platforms,” JPDC, vol. 70, no.9, pp. 962-974, 2010.
- [18] A. Kshemkalyani, M. Singhak, “Deadlock Detection in Distributed systems”, in Distributed Algorithms, 2010 – 2011.

(BBT nhận bài: 24/03/2014, phản biện xong: 17/04/2014)