

MODELAGEM E AVALIAÇÃO DE COMPORTAMENTO DE APLICAÇÕES DO TIPO *BAG OF TASKS* EM UMA NUVEM GERIDA PELO *OPENSTACK*

FERNANDO ANGELIN¹; GERSON GERALDO H. CAVALHEIRO²

¹UFPel – Universidade Federal de Pelotas – fangelin@inf.ufpel.edu.br

²UFPel – Universidade Federal de Pelotas – gerson.cavalheiro@inf.ufpel.edu.br

1. INTRODUÇÃO

A computação na nuvem revolucionou a indústria de Tecnologia da Informação e Comunicação possibilitando o provisionamento sob demanda de recursos computacionais elásticos com uma política pague pelo que utilizar (BELOGLAZOV, 2013), desta forma uma organização pode terceirizar suas necessidades computacionais para a nuvem e assim, evitar investimentos em infraestrutura computacional privada. A computação em nuvem vem crescendo nos últimos anos, oferecendo acesso a computadores de uma forma eficiente e remota. Neste tipo de modelo computacional, o uso dos recursos pode ser otimizado dinamicamente conforme as demandas de seus usuários. As mudanças na alocação dos recursos podem ser feitas em tempo real permitindo assim sistemas mais eficientes, evitando o super e/ou subdimensionamento dos recursos (ARMBRUST et al., 2010).

Muitas aplicações do nosso cotidiano necessitam de altas cargas de processamento. Porém existem certos períodos de tempo em que todo este processamento não está sendo realizado. Caso um usuário conte com recursos de processamento privado, é provável que se observe uma subutilização de seus recursos. Preocupados em incluir nos seus ambientes de computação, soluções que visam a diminuição dessa ociosidade de processamento, e o consequente gasto financeiro, os administradores de redes tem utilizado a técnica de virtualização. O uso da virtualização representa a ilusão de várias máquinas virtuais independentes, cada uma rodando uma instância de um sistema operacional virtualizado (SMITH; NAIR, 2005).

O cenário proposto para este trabalho é composto por uma nuvem oferecendo o serviço de processamento para computação intensiva, atendendo aplicações que necessitem altas demandas de processamento durante a execução. Muitas aplicações podem usufruir deste cenário, dentre elas estão as aplicações do tipo *Bag of Tasks* (*BoT*). Estas aplicações são compostas por uma coleção de tarefas independentes entre si, podendo ser executadas em qualquer ordem. Para isso, neste trabalho é proposto um modelo para este tipo de aplicação para que seja possível avaliar sua execução em uma nuvem computacional. Daremos preferência a uma nuvem gerida pelo *OpenStack*, um sistema operacional que visa gerenciar uma nuvem computacional de forma simplificada e eficiente (OPENSTACK, 2014).

Para a validação desta modelagem foi utilizado um simulador de cargas (aplicação que efetua uma série de cálculos gerando cargas sintéticas de processamento no CPU) que foi desenvolvido para a validação do *Neat*. Este componente também é utilizado, permitindo que o *OpenStack*, realize a consolidação dinâmica de máquinas virtuais utilizando a migração destas máquinas virtuais em tempo real.

Diferentes estratégias de escalonamento podem ser usadas para diferentes tipos de perfis de usuários e, neste caso, cabe avaliar o impacto de cada estratégia de escalonamento escolhida e aplicada às máquinas virtuais para essa classe de aplicações.

2. METODOLOGIA

Para a realização deste trabalho, foi estudado o *OpenStack*. Sua finalidade, seu funcionamento e arquitetura. Ele é entendido como um sistema operacional para gerenciamento de uma nuvem e que pode controlar grandes centros de computação, armazenamento e recursos de rede em um *datacenter* (OPENSTACK, 2014). Uma nuvem gerida pelo *OpenStack* provê uma solução de *Infrastructure as a Service* (IAAS). Tem a arquitetura modular e seus principais componentes são: *Nova*, *Swift*, *Cinder*, *Glance*, *Keystone*, *Horizon* e *Neutron*. Por se tratar de um projeto *open-source*, pode-se criar, adicionar ou remover componentes, promovendo uma maior liberdade na utilização dos componentes que melhor satisfaçam as necessidades do provedor do serviço em nuvem.

Posteriormente, foi estudado o componente que não está na distribuição oficial, mas que pode ser instalada para suprir a necessidade do administrador de uma nuvem, o *Neat*. Ele se trata de uma extensão para o *OpenStack* que implementa a consolidação dinâmica de máquinas virtuais utilizando a migração em tempo real. O principal objetivo desta consolidação dinâmica de máquinas virtuais (MVs) é melhorar a utilização dos recursos físicos disponíveis e reduzir o consumo energético por meio de realocação de MVs entre os nós físicos (NEAT, 2014). Foi dado ênfase aos algoritmos de detecção de sobrecarga, especialmente ao *Local Regression Robust* (*LRR*), que calcula as estimativas dos parâmetros de *Loess* e, usa-as para prever o futuro de utilização de CPU no próximo passo de tempo, tendo em conta também, o tempo de migração da máquina virtual.

Também foi estudado o padrão *Bag of Tasks* (*BoT*), que são descritas por um conjunto de tarefas independentes umas das outras (ANDREWS, 1999) e, não raro, possuem alto custo computacional. De forma oportunista, podem ser submetidas à nuvem para realizar o seu processamento, para aproveitar a característica paralela deste tipo de aplicações. Uma modelagem deste tipo de tarefas foi proposta, em formato de um conjunto de n triplas, que é descrito a seguir.

Para causar estresse nas MVs, foi utilizado um *script* que gera *Lookbusy*, ou seja, ele aplica cargas sintéticas no CPU, como se fosse um trabalho real. O servidor *controller* lê o *Bag of Tasks* e passa a distribuir para as MVs, que estão hospedadas nos nós físicos.

3. RESULTADOS E DISCUSSÃO

Os experimentos foram realizados em uma plataforma contendo cinco nós com a seguinte configuração: Processador Intel(R) Xeon(R) CPU E5-2420 1.90GHz (6 cores, 12 threads) e 8GB RAM. Destes nós, um é dedicado a realizar as operações de controle do *OpenStack* (*controller*) e os demais a hospedar as máquinas virtuais. Os experimentos realizaram o monitoramento do consumo energético e a migração de máquinas virtuais na execução de uma aplicação *BoT* descrita da seguinte forma: $\{[10, 30, 30], [20, 40, 90], \dots, [55, 7, 10], [12, 140, 95]\}$. Os valores descritos em cada tripla (de tarefas homogêneas), se refere: a quantidade de tarefas, o tempo (em segundos) que cada tarefa demanda de CPU e, a porcentagem de utilização de CPU. Esta descrição é compactada de forma a agrupar as tarefas que maximizem o uso das CPUs disponíveis. No estudo de

caso realizado a aplicação é composta de 2700 tarefas que consumiriam no total cerca de 60 horas em uma execução sequencial monoprocessada.

Foram realizados experimentos considerando a existência de 10, 20 e 40 máquinas virtuais e duas estratégias de detecção de sobrecarga, uma sem controle, não realizando migração de máquinas virtuais, outra migrando máquinas virtuais quando algum nó real é identificado como sobrecarregado por meio do LRR. Os resultados estão sumarizados na Tabela 1. Não utilizando a detecção de sobrecarga, podemos perceber que não ocorreram migrações, diferentemente de quando utilizado o LRR. A economia de energia se refere a quantidade de energia que, na totalidade, os nós deixaram de gastar por se utilizarem do *Neat* com relação a energia que seria gasta na situação em que os nós não usem o referido componente.

Tabela 1. Relação dos Resultados Obtidos

Experimento	Sem detecção de Sobrecarga					
	10 MVs		20 MVs		40 MVs	
	Migrações	Economia (%)	Migrações	Economia (%)	Migrações	Economia (%)
1	0	56,50	0	43,91	0	51,71
2	0	56,87	0	44,20	0	54,66
3	0	56,52	0	43,79	0	51,49
Detecção de Sobrecarga – LRR						
Experimento	10 MVs		20 MVs		40 MVs	
	Migrações	Economia (%)	Migrações	Economia (%)	Migrações	Economia (%)
1	6	55,58	39	47,33	74	50,05
2	18	54,64	39	51,75	76	54,66
3	14	53,28	39	50,61	74	51,21

Neste caso, para este ambiente descrito, ocorre um melhor aproveitamento com 20 MVs e com o LRR habilitado. Para um número menor de MVs, ocorre economia de energia sem detecção de sobrecarga pelo fato do nó físico comportar todas as MVs. Aparentemente ocorre economia de energia nos experimentos sem detecção com 40 MVs, porém, foi percebido que a maioria das MVs ficaram congeladas, não terminando sua computação.

4. CONCLUSÕES

No estudo realizado foi possível perceber que diferentes estratégias de detecção de sobrecarga implicam diferentes padrões de migração de máquinas virtuais. Na sequência, realizar-se-á o estudo de outras estratégias de detecção de sobrecarga, bem como avaliados o padrão de execução considerando diferentes configurações de aplicações *Bag of Tasks*.

Como trabalho futuro, a expectativa é desenvolver uma estratégia de detecção de sobrecarga eficiente para aplicações *Bag of Tasks* sobre uma nuvem gerida pelo *OpenStack*. Também, encontrar um número ótimo de máquinas virtuais que executarão sobre uma determinada nuvem, conforme suas características de hardware, para uma melhor otimização da energia elétrica consumida bem como a qualidade do serviço oferecido.

5. REFERÊNCIAS BIBLIOGRÁFICAS

ANDREWS, G. R. **Foundations of Multithreaded, Parallel, and Distributed Programming**. PEARSON, 1999. Tucson, AZ, USA.

ARMBRUST, M., FOX, A., GRIFFITH, R., JOSEPH, A. D., KATZ, R., KONWINSKI, A., LEE, G., PATTERSON, D., RABKIN, A., STOICA, I., and ZAHARIA, M. **A view of cloud computing**. Commun. ACM, 53(4):50–58, 2013.

BELOGLAZOV, A. **Energy-Efficient Management of Virtual Machines in Data Centers for Cloud Computing**. 2013. *PhD thesis*, THE UNIVERSITY OF MELBOURNE, Melbourne-Australia.

NEAT. **OpenStack Neat: Dynamic Consolidation of Virtual Machines on OpenStack**. Acessado em 6 de dezembro. 2014. Online. Disponível em: <http://openstack-neat.org/>

OPENSTACK. **Overview: OpenStack, The Open Source Cloud Operating System**. Acessado em 5 de dez. 2014. Online. Disponível em: <https://www.openstack.org/software/>

SMITH, J. E; NAIR, R. **The architecture of virtual machines**. Computer, 38(5):32–38, 2005.