Seminário de Iniciação Científica
Universidade do Estado de Santa Catarina
25° SIC UDESC

Explorando a Elasticidade para Otimizar Serviços Hospedados em Nuvens IaaS

Devair Dener Darolt [2], Guilherme Piegas Koslovski[1]

[1] Orientador, Departamento de Ciências da Computação – CCT, guilherme.koslovski@udesc.br
[2] Acadêmico do Curso de Bacharelado em Ciência da Computação – BCC – bolsista PROBIC/UDESC

Palavras-chave: Elasticidade, CloudSim, Nuvens Computacionais, Alocação

Com o surgimento das nuvens computacionais, algumas facilidades de gerenciamento foram introduzidas, como agilidade na alocação de recursos, segurança, escalabilidade e elasticidade. Essas características atraem clientes que necessitam alocar infraestruturas virtuais para o fornecimento de serviços capazes de suportar um grande número de acessos. Tais serviços exploram a elasticidade oferecida pelos provedores de nuvem para adaptar seus recursos computacionais e de comunicação, mantendo a qualidade dos serviços oferecidos e simultaneamente visando a minimização do custo para o fornecimento desses serviços.

Usualmente, serviços hospedados na nuvem são decompostos em diversas camadas. Neste cenário, um balanceador de carga recebe requisições e repassa para os servidores web e servidores de banco de dados distribuídos. Em cada camada, o número de recursos pode aumentar ou diminuir, caracterizando a elasticidade computacional. Neste caso o cliente terá um custo baixo quando o sistema estiver com poucas requisições. Quando um pico de processamento ocorre, novos recursos podem ser adicionados sob demanda.

Dentre os serviços ofertados por provedores de nuvens computacionais, o presente trabalho foca em Infraestrutura como Serviço (*Infrastructure as a Service – IaaS*). Este modelo trata-se do fornecimento dos recursos (processamento, capacidade de armazenamento, etc.) em sua forma fundamental através da abstração em máquinas virtuais. As aplicações n-camadas são hospedadas em infraestruturas virtuais. Dessa forma, nosso estudo explora a elasticidade dos recursos que compõem uma infraestrutura virtual para adaptar a carga de processamento das aplicações. Fornecer elasticidade a infraestruturas é uma tarefa complexa, portanto, elaboramos um algoritmo que monitora o tempo de resposta de cada requisição. Usando essa métrica, o algoritmo decide se máquinas virtuais (MVs) devem ser criadas ou destruídas.

O algoritmo proposto analisa o tempo médio de todas as requisições, que é comparado com um limiar de criação e um limiar de destruição de MVs. Assim, se essa média tiver uma variação de tempo que em percentual é maior que o limiar de criação, novas MVs são alocadas, diminuindo a sobrecarga das outras. Da mesma forma, quando a variação percentual da média é maior que o limiar de destruição, algumas MVs são destruídas evitando que elas fiquem gerando custo quando estão praticamente ociosas. Para destruir MVs, é verificado quais são os serviços que estão executando nesta VM, e então iniciado o processo de migração para outra MV. A criação de MVs é semelhante, porém neste processo é criado uma nova MV que receberá os serviços de outra MV existente, balanceando a carga computacional.

Para analisar o algoritmo proposto, foi estendido simulador NetworkCloudSim. O simulador, desenvolvido em Java, possibilita a representação de nuvens computacionais com diferentes configurações e utiliza sistemas orientados a eventos discretos para simular o funcionamento das entidades como usuários, hospedeiros, máquinas virtuais, *switches*, *cloudlets* e objetos que gerenciam políticas de alocação de recursos e comunicação da nuvem. Essas classes foram estendidas para atender os requisitos das

simulações. Através dessa simulação determinou-se a aplicabilidade da solução proposta, analisando o quanto de elasticidade é necessário para atender os requisitos com o menor custo possível.

As configurações utilizadas para as simulações foram: Cada hospedeiro foi modelado de forma padronizada com 16 GB de memória RAM, 16 CPUs, interconectados em uma topologia Fat-Tree, com 8 PODs, e largura de banda de 1 Gbps entre *edge switches* e *aggregation switches,* e 10 Gbps entre *aggregation switches* e *root switches.* As configurações das máquinas virtuais alocadas para hospedar os balanceadores de carga, servidores web e bancos de dados seguiram uma distribuição uniforme entre configurações pré determinadas. A RAM foi selecionada entre 4 GB, 2 GB e 1 GB, enquanto a vCPU foi selecionada entre 8, 4, 2 e 1 (numero de núcleos alocados). A alocação do enlace foi selecionado entre 50%, 25% e 12% da capacidade de 1 Gbps, padrão usada na modelagem da topologia interna do *datacenter*.

Três cenários de testes foram analisados variando o número máximo de recursos elásticos (em percentual): i) sem elasticidade; ii) 100% de elasticidade; e iii) e o terceiro com 200%. Para cada cenário, a infraestrutura virtual era inicialmente composta por 5 máquinas virtuais (1 balanceador de carga, 3 servidores web e um banco de dados). Inicialmente, 100 requisições foram submetidas ao sistema. Para simular picos de execução, foram adicionadas mais 100 requisições aleatoriamente até um total de 500 requisições. Para o cenário sem elasticidade, as MVs não podem ser destruídas e novas VMs não podem ser criadas, já nos demais cenários, MVs são dinamicamente provisionadas e removidas (quando ociosas). Dessa forma é agregado ao sistemas novos servidores web, balanceadores de cargas e bancos de dados de forma a minimizar o tempo das requisições.

Resultados obtidos:

Fig. 1 *Custo total das VMs*

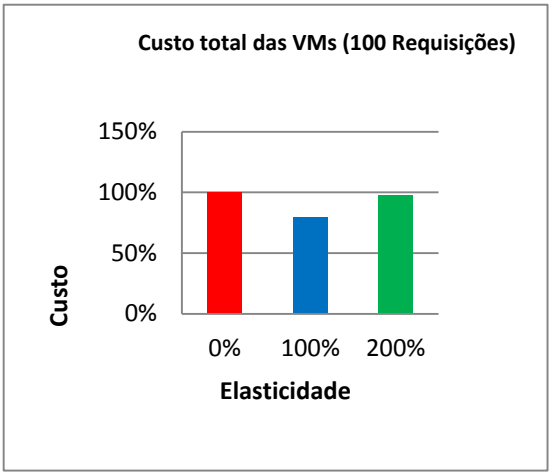
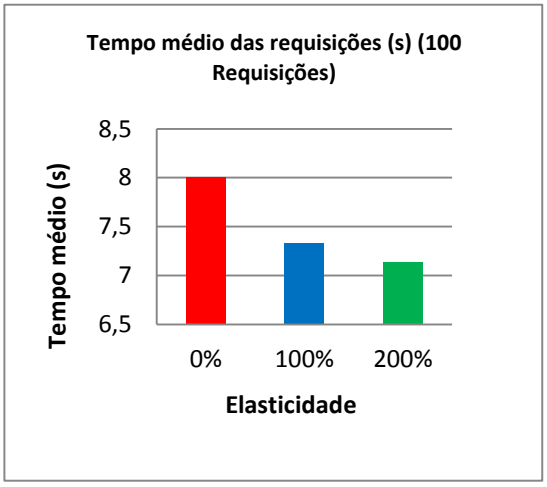


Fig. 2 *Tempo médio das requisições*



Através da simulação, foram obtidos resultados no qual foi possível observar que com o aumento de requisições ao sistema há um aumento no custo total de alocação das MVs, assim como o aumento no tempo médio das requisições. Ao analisarmos o gráfico para 100 requisições percebemos que na situação de 100% de elasticidade o custo é menor, enquanto o tempo médio das requisições é menor para o caso de 200% de elasticidade, pois com mais MVs escalonadas ocorre uma sobrecarga menor nos servidores. Ainda, podemos perceber que o modelo com 100% de elasticidade possui uma melhora de 20% no custo e 8% no tempo médio de execução, em relação ao modelo sem elasticidade. Já o cenário de 200% de elasticidade possui uma melhora de apenas 3% no custo e 11% no tempo médio de execução. Com isso pode se concluir que 100% de elasticidade possui custos mais favoráveis em relação ao tempo médio de execução.