Avaliação de Serviços de Computação na Nuvem para Aplicações de e-Ciência

Fábio Jorge Almeida Morais[1], **Francisco Vilar Brasileiro**[1]

[1] Universidade Federal de Campina Grande
Departamento de Sistemas e Computação
Laboratório de Sistemas Distribuídos
58.429-900, Campina Grande, PB

{fabio,fubica}@lsd.ufcg.edu.br

Abstract. *The cloud computing market has experienced an enormous growth in the last few years. Representative companies started offering computational resources on-demand and with no customer's long-term commitments, the so called Infrastructure-as-a-Service elastic solution. The different pricing models used by these companies enable a combinational relationship between price and quality of service. Furthermore, the increasing demand for high-performance computing applications has boosted the problem of how to choose the provider and/or solution which suits better for this type of customer. In this work we analyze empirically and analytically the main solutions offered by the cloud computing market, considering performance issues and the cost of e-Science applications, and taking into account variations in the price-QoS relationship. By the obtained results, we conclude that the best cloud computing solution for e-Science applications is provided by the Amazon AWS's spot-market model.*

Resumo. *O mercado de computação na nuvem obteve um grande salto nos últimos anos. Grandes empresas passaram a vender recursos computacionais sob demanda, de forma elástica e sem comprometimento de longo prazo por parte do cliente, as chamadas infraestruturas como serviço. Modelos de negócio utilizados por essas empresas para a oferta e tarifação possibilitam variações na relação entre fatores de preço e nível de serviço. Somando-se isso à crescente demanda de aplicações de alto desempenho e vazão computacional temos o problema de seleção do provedor e do tipo de serviço que melhor se adapta às necessidades desse perfil de cliente. Este trabalho realiza avaliações experimental e analítica entre os principais serviços do mercado de computação na nuvem, com relação aos fatores desempenho e custo para aplicações de e-ciência, considerando variações na relação entre preço e nível de serviço. Os resultados sugerem que a melhor opção de serviços de computação na nuvem para aplicações de e-ciência é provida pelo modelo spot da Amazon AWS.*

1. Introdução

O mercado de serviços públicos de "*Computação na Nuvem*"(do inglês *cloud computing*) obteve elevado crescimento nos últimos anos. Este mercado oferta infraestrutura computacional como serviço. Diversas companhias compõe o mercado de computação na nuvem, como a Amazon, Google, Microsoft e Rackspace.

Uma parcela destas oferece o serviço de IaaS (do inglês *infrastructure-as-a-service*) [Stanoevska-Slabeva and Wozniak 2010], onde o usuário adquire um recurso computacional como poder de processamento, memória e disco de um provedor de IaaS e utiliza esse recurso, de forma escalável e flexível, para implantar e executar suas aplicações [Sriram and Khajeh-Hosseini 2010].

O crescimento do mercado de computação na nuvem é decorrente das vantagens em termos de custo e confiabilidade na utilização de serviços de computação na nuvem em comparação aos serviços tradicionais, que consistem na aquisição de infraestruturas próprias [Armbrust et al. 2009]. Uma vantagem se dá na redução dos custos computacionais e de operações, devido ao baixo custo de manutenção e de investimento em infraestruturas de Tecnologia da Informação (TI) [Zardari and Bahsoon 2011]. Outra vantagem é baseada na redução do custo de capital, decorrente da utilização do modelo "pague conforme utilização"(do inglês *pay-as-you-go*), onde a medição e a tarifação são baseadas na utilização real do serviço, independente do período do tempo no qual ocorre a utilização [Armbrust et al. 2010], ou seja, o usuário paga apenas pelos recursos consumidos.

A relação de negócio estabelecida entre o provedor do serviço de computação na nuvem e o consumidor é mediada através de um SLA (do inglês *service level agreement*) que é responsável por referir as expectativas do cliente e a prestação de serviço do provedor de computação na nuvem [Buyya et al. 2008]. O SLA possui associado a si um modelo de tarifação responsável por estabelecer os custos de utilização do serviço provido.

No entanto, a relação entre os fatores de preço e nível de serviço (SLA) oferecido é muitas vezes desconhecida e variável. Um caso exemplar é o provedor Amazon AWS [Amazon 2011] que oferece serviços a partir de um modelo *"best-effort"* (*spot model*) que possibilita a compra de instâncias a um custo inferior ao praticado no modelo dedicado (*on-demand model*), mas sem garantias de disponibilidade do serviço, ou seja, o cliente pode vir a perder a instância a qualquer momento.

Ortogonalmente, a comunidade científica enxerga os benefícios advindos desse novo paradigma, como escalabilidade e flexibilidade de serviços e recursos, como um atrativo para a utilização dessas infraestruturas para a execução de uma classe de aplicações que torna-se cada dia mais popular no contexto da computação no mundo. Essa classe de aplicações é chamada de e-ciência (do inglês *e-science*), que é constituída de aplicações altamente paralelizáveis e que exigem alto desempenho e vazão computacional, tal como as aplicações de perfis HPC e HTP (do inglês *high-performance computing* e *high-throughput computing*) [Litzkow et al. 1988], MTC (do inglês *many-task computing*) [Raicu et al. 2008] e BoT (do inglês *bag-of-tasks*) [Cirne et al. 2003].

Outra particularidade dessa classe de aplicações é que os critérios e a importância dada a cada um destes, na avaliação dos níveis de satisfação com o serviço requisitado, não seguem o padrão convencional. Para aplicações com esse perfil normalmente se prioriza o tempo de execução da aplicação em relação ao custo de execução, ou seja, existe a possibilidade de inversão ou alteração das

relevâncias dos fatores que exercem influência no processo de seleção do serviço de computação na nuvem, visto que para execução de aplicações de e-ciência faz-se necessário altas capacidades computacionais por fatias curtas de tempo.

Por conseguinte, a dinamicidade das relações entre preço e SLA existente em modelos de negócio oferecidos pelos provedores de computação na nuvem somada a utilização destes serviços para execução de aplicações de e-ciência, particularmente aplicações intensivas em computação (do inglês *CPU intensive*) e intensivas em computação e memória (do inglês *CPU-Memory intensive*), faz surgir a problemática da seleção mais adequada de provedores e serviços para a execução desta classe de aplicações, quando considerada a possibilidade de modificação dos critérios de seleção e de suas relevâncias.

Desta problemática emerge a questão: *Quais os serviços de computação na nuvem (IaaS) para execução de aplicações de e-ciência apresentam os melhores níveis de utilidade, quando avaliados e comparados em função de desempenho e custo, considerando a dinamicidade da relação entre preços e contratos de nível de serviço inerente aos modelos de negócio utilizados?*

Alguns autores vem trabalhando em pesquisas neste contexto, através de ferramentas de comparação de serviços de computação na nuvem para modelos dedicados [Li et al. 2010], da análise de redução de custos na utilização de modelos do tipo *"best-effort"* [Yi et al. 2010, Chen et al. 2011] e da avaliação de serviços de computação na nuvem para aplicações de e-ciência [Ostermann et al. 2010, Iosup et al. 2010]. No entanto, estes trabalhos apresentam limitações de escopo, seja por não levarem em consideração a relação entre contrapartida (do inglês *trade-off*) de desempenho e custo dos recursos, seja por não tratarem de uma variedade de modelos de tarifação ou por avaliarem os serviços sem considerar possíveis oscilações de preço e disponibilidade de serviço pertencentes aos modelos de negócio praticados no mercado de computação na nuvem.

Este trabalho tem o objetivo de avaliar e comparar, experimental e analiticamente, serviços de computação na nuvem (IaaS), quanto ao desempenho e custo, para aplicações intensivas em computação e em computação e memória, considerando as possíveis variações na relação entre preço e SLA.

2. Trabalhos Relacionados

Trabalhos vem sendo desenvolvidos quanto a comparação sistemática de provedores de computação na nuvem [Li et al. 2010], como a ferramenta *CloudCmp* [Li et al. 2011] que se propõe a avaliar o desempenho e o custo dos serviços oferecidos para classes de problemas em computação sob demanda, mas limita-se a utilizar apenas o modelo dedicado de serviço como prática de negócio no estudo.

Iosup *et al*. [Iosup et al. 2010] e Ostermanm *et al*. [Ostermann et al. 2010] realizam uma análise comparativa de desempenho de uma classe de aplicações de e-ciência, com o objetivo de avaliar o rendimento dos serviços de computação na nuvem. Conclui-se que os atuais serviços oferecidos são uma alternativa para atividades que necessitam de recursos instantaneamente ou temporariamente.

O trabalho de Yi *et al*. [Yi et al. 2010] investiga a possibilidade de redução

dos custo de execução de aplicações no modelo *spot* da Amazon AWS através de políticas de pontos de controle (do inglês *checkpoints*). Enquanto que o trabalho de Chen *et al*. [Chen et al. 2011] concentra-se no estudo de algoritmos de escalonamento de recursos virtuais através de modelos de utilidade para avaliar a interação entre a satisfação do usuário e o lucro do serviço. Estes trabalhos fazem uso do histórico de preços do medelo *spot* da Amazon AWS em formato de séries temporais para simular e avaliar as soluções propostas.

Uma abordagem diferente foi utilizada por Andrzejak *et al*. [Andrzejak et al. 2010], propondo um modelo probabilístico para otimização de custo, desempenho e confiabilidade para a execução de aplicações do tipo BoT através do modelo *spot*. Todavia, o modelo descrito baseia-se na estimativa do lance necessário para manter os níveis de qualidade pretendidos, diferente da análise realizada neste estudo que é dirigida ao nível de utilidade obtido considerando o *trade-off* entre desempenho e custo.

De acordo com o levantamento bibliográfico realizado, não há conhecimento de trabalhos que consideram os efeitos da variação da relevância dos fatores desempenho e custo no processo de seleção de serviços de computação na nuvem para aplicações de e-ciência, quando considerada a utilização dos diferentes modelos de negócio existentes no mercado.

3. Diferentes Aplicações Levam a Diferentes Escolhas

A avaliação e comparação dos serviços de computação na nuvem, mais especificamente de serviços de IaaS, foi realizada inicialmente através de uma análise empírica das infraestruturas como serviço disponibilizadas pelo principal provedor de computação na nuvem do mercado, Amazon AWS. Os tipos de instância, infraestruturas de execução, selecionados e suas características gerais podem ser observados na Tabela 1.

Tabela 1. Tipos de instâncias selecionados e características gerais

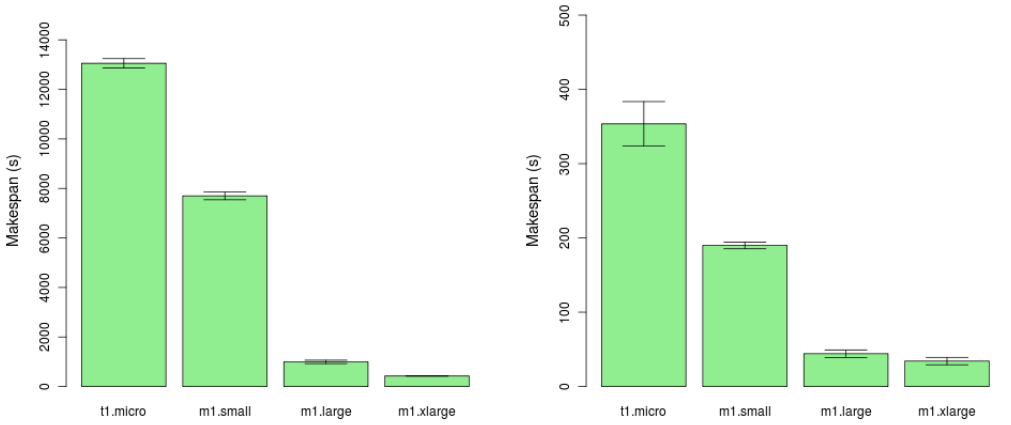
Instância	CPU	Memória	Preço (*on-demand*)
Micro (t1.micro)	Até 2 ECUs[1]	613 MB	US$ 0.02/hora
Small (m1.small)	1 ECU[1]	1700 MB	US$ 0.085/hora
Large (m1.large)	4 ECUs[1]	7500 MB	US$ 0.34/hora
Extra Large (m1.xlarge)	8 ECUs[1]	15000 MB	US$ 0.68/hora

[1] *EC2 Compute Unit*

O experimento consiste na avaliação de desempenho e custo de instâncias segundo o tempo entre a submissão e a conclusão (*makespan*) da execução de aplicações de referência (*benchmarks*) paralelizáveis, intensivas em computação (CPU) e em computação e memória (CPU e Memória), que caracterizam-se como aplicações de e-ciência. Os *benchmarks* selecionados, obedecendo aos perfis desejados, foram respectivamente: FBench [Walker 2011] e CloudCmp [Li et al. 2011]. Cada *benchmark* foi adaptado para referenciar apenas um perfil desejado de aplicação.

Foram executadas 10 repetições para cada par ⟨Instância, *Benchmark*⟩ no experimento, devido à baixa variabilidade dos resultados entre as repetições. As

comparações utilizam intervalos de confiança com coeficientes de confiança de 95%.



(a) *Makespan* do *benchmark* FBench (b) *Makespan* do *benchmark* CloudCmp

Figura 1. Tempo de execução dos *benchmarks* para as instâncias avaliadas

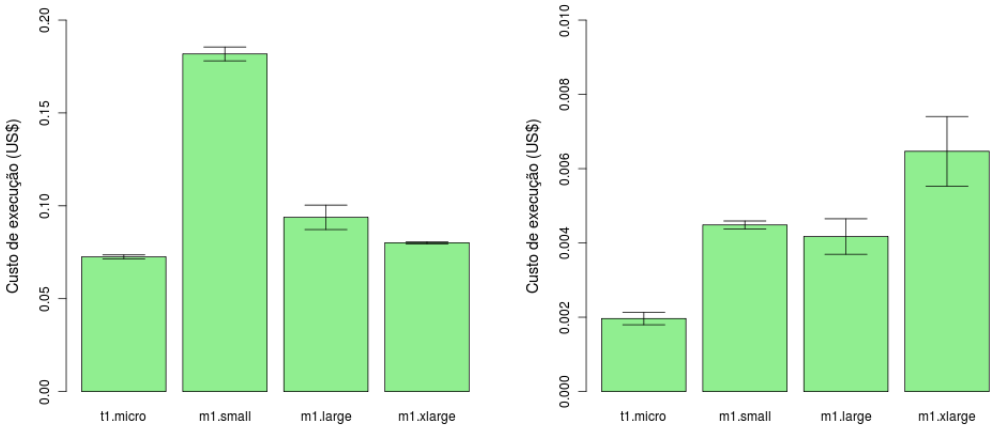
Os resultados de *makespan* obtidos da experimentação, ou seja, o desempenho dos quatro tipos de instância para aplicações segundo os perfis dos *benchmarks* selecionados, podem ser visualizados na Figura 1.

Para avaliar os serviços quanto ao custo de execução é necessário considerar um modelo de tarifação para as instâncias utilizadas no experimento. O modelo *on-demand* da Amazon AWS é o mais utilizado por oferecer instâncias dedicadas por tempo indeterminado e por realizar a tarifação por hora de utilização, e por tais motivos foi o escolhido para esta avaliação inicial. Contudo, o custo de execução foi estimado a partir de uma função de tarifação proporcional para tarifação em segundos, dado que os *makespans* obtidos são inferiores a uma hora. A limitação do tempo de execução dos *benchmarks* foi necessária pelo alto custo do experimento, que poderia vir a comprometer a realização deste estudo.

Os custos de execução obtidos da aplicação do modelo *on-demand*, através dos valores presentes na Tabela 1, para os *makespans* resultantes do experimento inicial encontram-se expostos na Figura 2.

Estes resultados, tanto de *makespan* quanto de custo de execução, apontam para a problemática da escolha do serviço de computação na nuvem que melhor se adéqua ao perfil da aplicação que utilizará a infraestrutura para sua execução.

Fica explícito que dependendo do fator de escolha, *makespan* ou custo, que for priorizado e do perfil da aplicação executada os resultados das escolhas são diferentes. Isto é, existe um *trade-off* entre *makespan* e custo no processo de seleção de serviços de computação na nuvem para execução de aplicações de e-ciência.



(a) Custo de execução do *benchmark* FBench

(b) Custo de execução do *benchmark* CloudCmp

Figura 2. Custo de execução dos *benchmarks* para as instâncias avaliadas

4. Modelo Bi-Critério de Utilidade de Serviços

4.1. Definição do Modelo

A problemática da escolha de serviços de computação na nuvem exposta anteriormente pode ser formalizada através da modelagem de uma função bi-critério de utilidade, que considera custo e *makespan* como fatores influenciantes do processo de decisão. Para tal, propomos a utilização de uma função Cobb-Douglas de utilidade [Goldberger 1968].

Funções deste tipo são amplamente utilizadas em economia para representar a relação de utilidade entre bens consumidos. Nesse trabalho a função Cobb-Douglas modela adequadamente as preferências do decisor com relação ao *trade-off* entre custo e *makespan*, pois os fatores envolvidos na decisão são inversamente proporcionais, ou seja, crescem em direções opostas.

Um exemplo, como visto na seção anterior, desse comportamento acontece para aplicações intensivas em computação e memória, pois a priorização da redução de *makespan* leva à seleção de instâncias de maior capacidade a custos maiores, enquanto que a priorização da redução do custo favorece a escolha de instâncias a custos menores mas com níveis elevados de *makespan*.

Para o problema em questão temos como objetivo maximizar a função utilidade sujeita aos níveis de relevância, através da redução dos valores dos fatores custo e *makespan*, para a execução de uma carga de trabalho do tipo w composta por aplicações do tipo BoT com n_w tarefas e tarefas com tempo médio de execução dado por $\bar{m}_w$.

Na utilização do modelo de negócio dedicado, *on-demand*, são garantidos os níveis de qualidade de serviço estabelecidos. Desta forma, é desconsiderado qualquer fator externo, não inerente à aplicação, como agente influenciador dos valores de custo e *makespan* na execução da carga de trabalho.

A função de custo (Equação 1) é modelada a partir dos conceitos de associatividade de custo (do inglês *cost associativity*) [Armbrust et al. 2010], considerando apenas o valor V_i cobrado pela instância do tipo i, o tamanho da carga de trabalho, dado pelo produto de n_w e $\bar{m}_w$, e o fator de serviço $K_{i,w}$.

$$C_{i,w} = V_i \times n_w \times \bar{m}_w \times K_{i,w} \tag{1}$$

O fator multiplicador de serviço $K_{i,w}$ é derivado da experimentação realizada na Seção 3 através da relação entre a média $\bar{M}^e_{i,w}$ e a média mínima $\bar{M}^e_{w,min}$ experimental do *makespan* de execução dos dois perfis de carga de trabalho, e consiste em:

$$K_{i,w} = \frac{\bar{M}^e_{i,w} - \bar{M}^e_{w,min}}{\bar{M}^e_{w,min}} + 1 \tag{2}$$

A função de *makespan* (Equação 3) é função do tempo despendido para a execução da carga de trabalho e do número máximo de instâncias que podem ser adquiridas simultaneamente L, onde a divisão pelo menor valor dentre L e n_w garante nível ótimo de particionamento das tarefas. Esse valor L é tipicamente restringido pelo provedor [Costa et al. 2011].

$$M_{i,w} = \frac{K_{i,w} \times n_w \times \bar{m}_w}{min(L, n_w)} \tag{3}$$

No entanto, para deixar os valores do custo e do *makespan* em uma mesma escala é necessário a normalização, para uma escala entre 0 e 1, destes valores da seguinte forma:

$$C_{i,w,norm} = \frac{C_{i,w} - C_{min,w}}{C_{max,w} - C_{min,w}} \tag{4}$$

$$M_{i,w,norm} = \frac{M_{i,w} - M_{min,w}}{M_{max,w} - M_{min,w}} \tag{5}$$

Finalmente, a maximização dos níveis de utilidade consiste na redução dos valores do custo e do *makespan*. Para tal é realizada uma normalização e inversão da função de utilidade (Equação 6), onde α é o nível de importância dado ao custo e $1 - \alpha$ é o nível de importância dado ao *makespan* (em porcentagem).

$$U_{i,w,norm} = 1 - \left(C^{\alpha}_{i,w,norm} \times M^{(1-\alpha)}_{i,w,norm}\right) \tag{6}$$

4.2. Implementação e Execução do Modelo

O modelo proposto anteriormente foi implementado através da ferramenta R de análise estatística [Chambers 2011], de tal forma que fosse possível a exercitação e execução do modelo produzido.

A execução do modelo cobriu diversos cenários do *trade-off* da seleção de serviços de computação na nuvem, inclusive excedendo os limites utilizados no experimento inicial. A faixa de valores atribuída a n_w tem o intuito de cobrir desde situações simplistas, uma tarefa por carga de trabalho, até situações realistas, quando o número de tarefas ultrapassa os limite L. Os níveis assumidos por fator e variável pertencentes ao modelo são apresentados na Tabela 2.

Tabela 2. Parâmetros utilizados na execução do modelo de utilidade (*on-demand*)

Parâmetro	Valor
n_w	{1; 5; 10; 20; 50}
$\bar{m}_w$	1
L	20
α	0 a 1 com passos de 0.05
Tipo de instância	{micro; small; large; xlarge}
Perfil da aplicação	{CPU; CPU e Memória}

4.3. Resultados e Discussão

Tendo como referência cargas de trabalho intensas em computação, temos que a instância *m1.xlarge* é a que apresenta os maiores índices de utilidade para quase todos os níveis de prioridade α (Figura 3). Isso só não é verdade quando o valor de α está muito próximo de 1, ou seja, quase toda a prioridade consiste na redução do custo de execução.

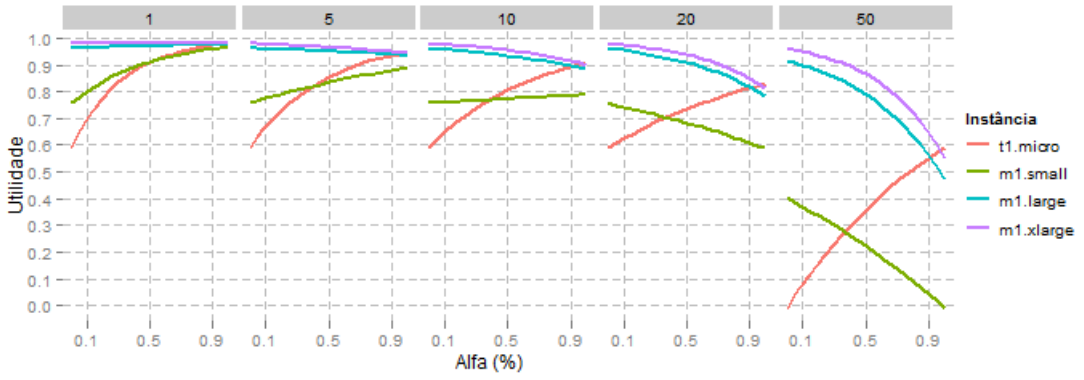


Figura 3. Níveis de utilidade para *workloads* intensivos em CPU

Estes resultados apresentam o mesmo comportamento do experimento realizado na Seção 3. Ficando evidente que para cargas de trabalho intensivas em computação a instância *m1.xlarge* se sobressai dentre as demais, possivelmente por apresentar alta capacidade computacional e por possuir um incremento de capacidade computacional proporcionalmente inferior ao incremento de custo quando comparado com as demais instâncias.

Por outro lado, para cargas de trabalho intensivas em computação e memória os melhores índices de utilidade são compartilhados entre as instâncias *m1.xlarge*, *m1.large* e *t1.micro*. A variação de prioridade, da redução do *makespan* à

redução do custo, elege gradativamente as instâncias *m1.xlarge*, *m1.large* e *t1.micro* como as que possuem os melhores utilidades (Figura 4).

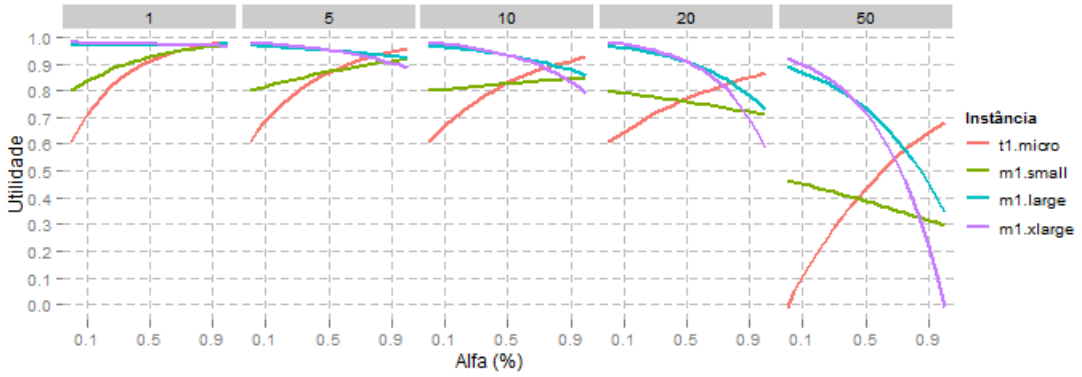


Figura 4. Níveis de utilidade para *workloads* intensivos em CPU e Memória

No entanto, esta distinção entre os diferentes tipos de instância fica mais evidente com o aumento do número de tarefas da carga de trabalho. Também é percebido uma diminuição dos valores de utilidade quando o número de tarefas da carga de trabalho supera o valor de L, pois neste caso a quantidade de carga de trabalho (número de tarefas) a ser executada em cada instância é incrementada.

5. Utilizando Recursos Voláteis

Uma das características mais importantes de aplicações de e-ciência é a não exigência de uma rígida garantia de qualidade de serviço por parte da infraestrutura de execução [Fraga et al. 2011].

Alguns modelos de negócio aplicados no mercado possuem uma relação não estática entre qualidade de serviço e custo. Por exemplo, no modelo *spot* da Amazon AWS, o usuário pode ter o custo de tarifação elevado e até ter a perda do recurso durante a utilização, ato de preempção.

Tabela 3. Mediana dos preços do modelo *spot* por tipo de instância

Instância	Preço (US$/hora)	Economia
Micro	0.013	35%
Small	0.031	64%
Large	0.12	65%
Extra Large	0.24	65%

Todavia, este modelo consiste, teoricamente, no modelo em que se pode atingir os menores custos de execução, como pode ser observado na taxa de economia obtida com relação aos valores praticados no modelo *on-demand* (Tabela 3). Os valores representativos do modelo de tarifação *spot* no último ano (Tabela 3) foram obtidos através da mediana do histórico de valores de cada tipo de instância nesse período [Timetric 2011]. Desta forma, a utilização deste modelo de negócio para a execução de aplicações de e-ciência pode ser considerada.

Ao utilizar um modelo de negócio como o *spot*, faz-se necessário o desenvolvimento de técnicas que garantam que as cargas de trabalho chegarão ao seu término, mesmo que ocorra a preempção da instância alocada. Para tal, é utilizado o conceito de *checkpoint* para que em caso de preempção da instância seja necessária a reexecução de apenas parte da carga de trabalho.

Contudo, para se aplicar esta técnica é necessário modificar as funções de custo e *makespan* descritas anteriormente. Onde a função custo corresponde à soma do custo de execução da carga de trabalho e do custo de realização dos *checkpoints* (Equação 7). A função *makespan*, por sua vez, corresponde ao somatório dos tempos associados à execução da carga de trabalho, ou seja, o *makespan* de execução da carga em si somado ao tempo de realização dos *checkpoints* e aos possíveis tempos de reexecução de tarefas (Equação 8). Considerando o pior caso em que a preempção ocorreu no instante anterior à realização do *checkpoint*, ou seja, faz-se necessário a reexecução de uma fatia de tempo aproximadamente igual ao tempo inter-*checkpoints*.

$$C^s_{i,w} = V_i \times n_w \times \bar{m}_w \times K_{i,w} \times \left(1 + \frac{to_w}{\Delta tc_w}\right) \tag{7}$$

$$M^s_{i,w} = \frac{n_w \times \bar{m}_w \times K_{i,w} \times \left(1 + \frac{to_w}{\Delta tc_w}\right)}{min(L, n_w)} + q_i \times \Delta tc_w \tag{8}$$

Onde:

- Δtc_w: intervalo inter-*checkpoints* para uma carga de trabalho do tipo w;
- to_w: tempo despendido para realização de um *checkpoint* para uma carga de trabalho do tipo w;
- q_i: número de ocorrências de preempção da instância i.

O valor de q pode ser estimado a partir de uma distribuição de probabilidade binomial, $binom(N, \rho)$, que retorna o número de ocorrências de sucesso dentre N tentativas, com parâmetros N igual ao número total de ocorrências e ρ igual a probabilidade de sucesso.

No contexto do modelo *spot* uma ocorrência é equivalente a uma mudança no valor da instância, enquanto que ρ corresponde à probabilidade de preempção dado que o usuário estabeleceu um valor V_b como lance. Para extrairmos os valores de N e ρ utilizamos o histórico da Amazon AWS, de um ano, de variação do preço no modelo *spot* das instâncias consideradas neste estudo.

O valor de N é dado pela frequência de ocorrência de mudanças de preço para uma dada instância multiplicada pelo tempo total de execução da carga de trabalho considerando a realização de *checkpoints* (Equação 9), ou seja:

$$N_{i,w} = F_i \times \frac{\left(n_w \times \bar{m}_w \times K_{i,w} \times \left(1 + \frac{to_w}{\Delta tc_w}\right)\right)}{min(L, n_w)} \tag{9}$$

Os valores de F correspondem às frequências de mudança do preço das instâncias avaliadas segundo o histórico de mudanças de preço do último ano. Enquanto que o valor de ρ corresponde à probabilidade de preempção de um tipo de instância para um lance de valor V_b considerando o histórico de preços do último ano.

Desta forma, a função *makespan* obtida é:

$$M^s_{i,w} = \frac{n_w \times \bar{m}_w \times K_{i,w} \times \left(1 + \frac{to_w}{\Delta tc_w}\right)}{min(L, n_w)} + binom(\lceil N_{i,w} \rceil, \rho_i) \times \Delta tc_w \quad (10)$$

Não obstante, para que seja realizada a maximização dos níveis de utilidade é necessário a normalização das funções custo (Equação 4) e *makespan* (Equação 5) e a normalização e inversão da função utilidade, como mostrado na Equação 6, pelos mesmos motivos apresentados anteriormente (Seção 4).

5.1. Implementação e Execução do Modelo

Da mesma forma que na Seção 4 o modelo de utilidade foi implementado segundo as características do modelo de negócio *spot*. Novos parâmetros e níveis de atribuição foram adicionados ao estudo nesse momento, conforme exposto na Tabela 4.

Tabela 4. Parâmetros utilizados na execução do modelo de utilidade (*spot*)

Parâmetro	Valor
n_w	{50; 100; 500; 1000}
$\bar{m}_w$	1
L	100
to_w	1/60
Δt_c	{0.25; 0.5; 1; 2} de m_w
α	0 a 1 com passos de 0.05
Tipo de instância	{micro; small; large; xlarge}
Perfil da aplicação	{CPU; CPU e Memória}

A execução exercita o modelo de forma a cobrir uma variedade de tamanhos da carga de trabalho em contrapartida à frequência de realização de *checkpoints*, que é determinada com relação ao valor de $\bar{m}_w$, por exemplo, Δt_c igual a 50% de $\bar{m}_w$. Além de considerar o valor V_b como metade do valor praticado no modelo *on-demand* para cada tipo de instância. Os valores de ρ obtidos são apresentados na Tabela 5.

5.2. Resultados e Discussão

Os resultados obtidos para a utilidade dos serviços de computação na nuvem para cargas de trabalho intensivas em computação e memória, para o modelo de negócio *spot*, são bastante semelhantes aos da utilidade para o modelo *on-demand* (Figura 5). Ou seja, os níveis mais elevados de utilidade são pertencentes às

Tabela 5. Probabilidades de preempção para os tipos instância selecionados

Instância	ρ (V_b = valor *on-demand*)	ρ (V_b = valor *on-demand*/2)
Micro	0.00%	100%
Small	1.04%	7.41%
Large	0.25%	6.72%
Extra Large	0.41%	1.03%

instâncias *m1.xlarge*, *m1.large* e *t1.micro*. Esse comportamento é justificado pela pequena probabilidade de preempção, exceto para a micro, o que implica em uma redução pouco significativa nos valores de utilidade do modelo.

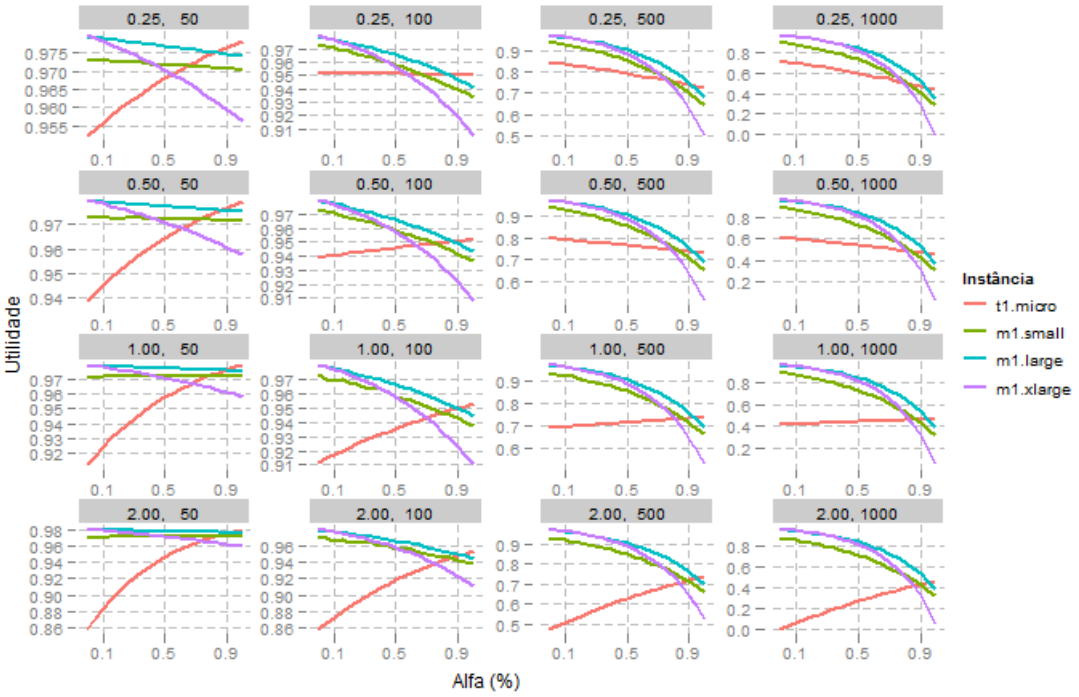


Figura 5. Níveis de utilidade para *workloads* intensivos em CPU e Memória

O estudo de utilidade também foi realizado para cargas de trabalho intensas em CPU, no entanto, por limitações de espaço seus resultados não serão discutidos aqui. Os modelos e dados produzidos neste estudo podem ser encontrados no sítio http://www.lsd.ufcg.edu.br/ fabio/cloud-utility-model.tar

6. Conclusões e Trabalhos Futuros

Neste trabalho foi realizada uma análise comparativa de desempenho e custo de infraestruturas como serviço providas pela Amazon AWS para aplicações de e-ciência, considerando os modelos de negócio *on-demand* e *spot*.

Essa análise foi realizada por meio de experimentação, através de *benchmarks* representativos e publicamente disponíveis, e analiticamente, utilizando um modelo de utilidade que representa a problemática do *trade-off* entre custo e *makespan* para esse perfil de aplicações.

Concluímos que o processo de seleção de serviços de computação na nuvem é uma tarefa delicada, onde a priorização de fatores pode influenciar significativamente o nível de satisfação obtido. Além do que, para aplicações de e-ciência é fortemente indicada a utilização do modelo *spot*, devido aos benefícios agregados em termos de custo, mesmo tendo-se em conta a flexibilidade dos níveis de garantia de qualidade de serviço providos.

Como trabalhos futuros podem ser destacados novos estudos utilizando uma maior variedade de tipos de instância e de provedores e a incorporação ao modelo da capacidade de representação da utilidade quando a escolha de serviços considera um conjunto heterogêneo de instâncias.

Referências

Amazon (2011). Amazon web services. `http://aws.amazon.com`. Online; novembro, 2011.

Andrzejak, A., Kondo, D., and Yi, S. (2010). Decision model for cloud computing under sla constraints. In *Modeling, Analysis & Simulation of Computer and Telecommunication Systems (MASCOTS), 2010 IEEE International Symposium on*, pages 257–266. IEEE.

Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., et al. (2009). Above the clouds: A berkeley view of cloud computing. *EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2009-28*.

Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., et al. (2010). A view of cloud computing. *Communications of the ACM*, 53(4):50–58.

Buyya, R., Yeo, C., and Venugopal, S. (2008). Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities. In *High Performance Computing and Communications, 2008. HPCC'08. 10th IEEE International Conference on*, pages 5–13. Ieee.

Chambers, J. (2011). The r project for statistical computing. `http://www.r-project.org/`. Online; junho, 2011.

Chen, J., Wang, C., Zhou, B., Sun, L., Lee, Y., and Zomaya, A. (2011). Tradeoffs between profit and customer satisfaction for service provisioning in the cloud. In *Proceedings of the 20th international symposium on High performance distributed computing*, pages 229–238. ACM.

Cirne, W., Paranhos, D., Costa, L., Santos-Neto, E., Brasileiro, F., Sauvé, J., Silva, F., Barros, C., and Silveira, C. (2003). Running bag-of-tasks applications on computational grids: The mygrid approach. In *Parallel Processing, 2003. Proceedings. 2003 International Conference on*, pages 407–416. Ieee.

Costa, R., Brasileiro, F., Lemos, G., and Mariz, D. (2011). Sobre a amplitude da elasticidade dos provedores atuais de computação na nuvem. In *Anais do XXIX Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC2011)*. Sociedade Brasileira de Computação (SBC).

Fraga, E., Brasileiro, F., and Serey, D. (2011). Estimando o valor de uma grade entre pares para a execução de aplicações do tipo saco de tarefas. In *Anais do XXIX Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC2011)*. Sociedade Brasileira de Computação (SBC).

Goldberger, A. (1968). The interpretation and estimation of cobb-douglas functions. *Econometrica: Journal of the Econometric Society*, pages 464–472.

Iosup, A., Ostermann, S., Yigitbasi, N., Prodan, R., Fahringer, T., and Epema, D. (2010). Performance analysis of cloud computing services for many-tasks scientic computing. *IEEE Trans. on Parallel and Distrib. Sys*.

Li, A., Yang, X., Kandula, S., and Zhang, M. (2010). Cloudcmp: comparing public cloud providers. In *Proceedings of the 10th annual conference on Internet measurement*, pages 1–14. ACM.

Li, A., Yang, X., Kandula, S., and Zhang, M. (2011). Cloudcmp benchmark. `https://github.com/angl/cloudcmp/tarball/v0.1`. Online; novembro, 2011.

Litzkow, M., Livny, M., and Mutka, M. (1988). Condor-a hunter of idle workstations. In *Distributed Computing Systems, 1988., 8th International Conference on*, pages 104–111. IEEE.

Ostermann, S., Iosup, A., Yigitbasi, N., Prodan, R., Fahringer, T., and Epema, D. (2010). A performance analysis of ec2 cloud computing services for scientific computing. *Cloud Computing*, pages 115–131.

Raicu, I., Zhang, Z., Wilde, M., Foster, I., Beckman, P., Iskra, K., and Clifford, B. (2008). Toward loosely coupled programming on petascale systems. In *Proceedings of the 2008 ACM/IEEE conference on Supercomputing*, page 22. IEEE Press.

Sriram, I. and Khajeh-Hosseini, A. (2010). Research agenda in cloud technologies. *Arxiv preprint arXiv:1001.3259*.

Stanoevska-Slabeva, K. and Wozniak, T. (2010). Cloud basics–an introduction to cloud computing. *Grid and Cloud Computing*, pages 47–61.

Timetric (2011). Amazon aws spot price. `http://timetric.com/dataset/amazon-web-services-aws-spot-price`. Online; novembro, 2011.

Walker, J. (2011). Fbench - trigonometry intense floating point benchmark. `http://www.fourmilab.ch/fbench`. Online; junho, 2011.

Yi, S., Kondo, D., and Andrzejak, A. (2010). Reducing costs of spot instances via checkpointing in the amazon elastic compute cloud. In *Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on*, pages 236–243. IEEE.

Zardari, S. and Bahsoon, R. (2011). Cloud adoption: a goal-oriented requirements engineering approach. In *Proceeding of the 2nd international workshop on Software engineering for cloud computing (SECLOUD'11). ACM, New York, NY, USA*, pages 29–35.