Um Modelo de Consumo de Energia para Ambientes de Nuvem com Elasticidade

Gustavo Rostirolla[1], **Vinicius Facco Rodrigues**[1], **Rodrigo da Rosa Righi**[1]

[1]Prog. Interdisciplinar de Pós-Graduação em Computação Aplicada, Unisinos – Brasil

grostirolla1@gmail.com, viniciusfacco@live.com, rrrighi@unisinos.br

Resumo. *Uma das principais características da computação em nuvem é a elasticidade, que se refere à capacidade de alterar a quantidade de recursos em tempo real a fim de otimizar a execução de uma tarefa. Um dos principais desafios é como medir a sua eficácia, utilizando elasticidade em aplicações HPC é possível reduzir o tempo de aplicação, mas consumindo uma grande quantidade de recursos e/ou energia para concluir a tarefa. Particularmente, observa-se que o estado da arte não apresenta um modelo de consumo de energia que contempla um número maleável de recursos, mas apenas um número fixo e predefinido deles. Neste contexto, propõe-se um modelo de consumo de energia elástico. Os resultados revelaram uma acurácia média de 97,15%.*

Abstract. *One of the main characteristics of cloud computing is elasticity, which refers to the capacity of on-the-fly changing the number of resources to support the execution of a task. One of the main challenges in this scope is how to measure its effectiveness, because of elasticity enables high performance computing by reducing the application time, but an infeasible amount of resource and/or energy can be paid to accomplish this. Particularly, the state-of-the-art does not present an energy consumption model that fits a malleable number of resources, but only a fixed and predefined number of them. In this context, this article proposes an elastic energy consumption model. The results revealed a median accuracy of 97.15%.*

1. Introdução

Uma das principais características da computação em nuvem é a elasticidade, na qual os usuários podem escalar seus recursos computacionais a qualquer momento, de acordo com a demanda ou o tempo de resposta desejado [Lorido-Botran et al. 2014]. Considerando uma aplicação paralela de longa execução, um usuário pode querer aumentar o número de instâncias para tentar reduzir o tempo de conclusão da tarefa. Logicamente, o sucesso deste processo vai depender tanto do grão quanto da modelagem da aplicação. Por outro lado, se a tarefa não escala de forma linear ou perto de uma forma linear, e se o utilizador é flexível com respeito ao tempo de conclusão, o número de instâncias pode ser reduzida. Isso resulta em uma menor quantidade nós $\times$ horas, e portanto, em um custo mais baixo e melhor uso da energia. Graças aos avanços na área de virtualização [Petrides et al. 2012], a elasticidade em computação em nuvem pode ser uma alternativa viável para obter economia de custo significativa quando comparado com o método tradicional de manter uma infra-estrutura de TI baseada em *cluster*. Normalmente, neste último caso, há um dimensionamento para o uso de pico, sendo subutilizada quando observamos toda a execução do aplicativo ou ainda, ao analisar o uso real da infra-estrutura.

Elasticidade pode ser uma faca de dois gumes envolvendo desempenho e o consumo de energia. Ambos são diretamente relacionados ao consumo de recursos, o que também pode ajudar a medir a qualidade elasticidade. Embora elasticidade permita que os aplicativos aloquem e liberem recursos de forma dinâmica, ajustando às demandas da aplicação, estabelecer limites apropriados, medir o desempenho e consumo de energia com precisão neste ambiente não são tarefas fáceis [Lorido-Botran et al. 2014]. Desta forma, um utilizador pode conseguir um bom desempenho considerando o tempo para executar a sua aplicação, mas utilizando uma grande quantidade de recursos, resultando em um desperdício de energia. A ideia de apenas obter um melhor desempenho da aplicação com uma execução elástica, em alguns casos, não é suficiente para usuários e administradores da nuvem. Os usuários acabam pagando por um maior número de recursos, não efetivamente utilizados, de acordo com o paradigma *pay-as-you-go*. A medição do consumo de energia de tais sistemas elásticos não é uma tarefa fácil. Muitos trabalhos se concentram em medição e como estimar o consumo de energia em *data centers*, no entanto, essas tarefas são desafios ao considerar sistemas elásticos.

Desta forma, este artigo apresenta um modelo de consumo de energia pra ambientes elásticos que fornece dados sobre a energia consumida durante a execução de aplicações HPC *High Performance Computing* em ambientes de nuvem elásticos. Particularmente, o modelo proposto extrai dados de consumo de energia em uma infra-estrutura maleável (que permite variação do número de nós em tempo de execução), permitindo estabelecer relações entre o consumo de energia, consumo de recursos e desempenho. Com o objetivo de analisar o modelo de energia proposto, utilizou-se um trabalho anterior chamado AutoElastic [Righi et al. 2015], que consiste em um *middleware* que prove elasticidade reativa e gerencia os recursos da nuvem de acordo com a demanda de uma aplicação HPC. Assim, o modelo de energia atua como um complemento para o AutoElastic, salvando dados de energia durante o tempo de execução do aplicativo. Os resultados com uma aplicação de uso intensivo da CPU foram realizados em diferentes cenários: variando valores dos *thresholds* inferior e superior e variando as cargas de trabalho de entrada (Crescente, Descendente, Constante e Onda). A contribuição científica do artigo consiste no modelo de energia, incluindo equações e procedimentos de captura de dados, para infraestruturas de nuvem elásticas. Este modelo pode ser utilizado para medir a qualidade ($i.e.$ eficácia), do *middleware* que prove elasticidade principalmente quando utilizados em conjunto com funções de custo.

O restante deste artigo irá apresentar primeiramente modelo de consumo energético proposto na Seção 2. A metodologia de avaliação e discussão dos resultados estão descritos na Seção 3. A Seção 4 apresentar os trabalhos relacionados. Por fim, a Seção 5 apresenta as considerações finais, destacando as contribuições com dados quantitativos e a direção dos trabalhos futuros.

2. Modelo de consumo de energia para ambientes elásticos

Esta seção apresenta um modelo de consumo de energia para extrair dados sobre o consumo, explorando as relações entre o consumo de energia, consumo de recursos e desempenho. O modelo apresentado leva em consideração uma das principais características da computação em nuvem, a elasticidade, onde a quantidade de recursos muda durante o tempo de execução, assim como o consumo de energia.

A implantação de sensores de corrente ou Wattímetros pode ser caro se não for feito no momento em que toda a infraestrutura (*i.e.*, *cluster* ou *data center*) é instalada, além de ser custosa tanto em questões financeiras como em tempo conforme a infraestrutura cresce. Uma solução alternativa e menos dispendiosa é a utilização de modelos de energia para estimar o consumo de componentes ou de um *data center* inteiro [Orgerie et al. 2014]. Bons modelos devem ser leves (em relação ao consumo de recursos computacionais) e não interferir no consumo de energia que eles tentam estimar. Tendo em vista estes requisitos, o modelo proposto explora dados de energia capturados em um pequeno conjunto de nós, a fim de formular uma equação que estende os resultados para um conjunto arbitrário de nós homogêneos. Mais precisamente, a metodologia utilizada é similar a de Luo et al. [Luo et al. 2013] que consiste em três etapas:

(i) Coletar amostras de uso de recursos, bem como o consumo de energia da máquina utilizando um medidor de consumo. Neste caso, utilizou-se um medidor Minipa ET-4090 que coletou mais de 8000 amostras usando uma carga composta que pode consumir diversos tipos de recursos dos nós, a fim de representar aplicações reais em ambiente de nuvem [Chen et al. 2014];
(ii) Executar métodos de regressão para gerar o modelo de energia a ser utilizada posteriormente;
(iii) Testar o modelo em um conjunto diferente de dados, coletados com o medidor de diferentes máquinas homogêneas, a fim de validar se o modelo representa corretamente o consumo de energia das demais máquinas.

A fim de analisar a precisão do modelo gerado foram coletados dados de CPU, memória principal e consumo de energia instantâneos, aplicando posteriormente PCR (Regressão de Componentes Principais) em mais de 8.000 amostras obtidas a partir de um único nó. Os dados recolhidos estão alinhados com estudos anteriores [Orgerie et al. 2014], que apresentam a CPU como o principal vilão do consumo de energia. Após a geração deste modelo foi realizada a predição da mesma quantidade de amostras de energia baseada em amostras coletadas de CPU e memória de outro nó com mesma configuração de hardware. Comparando estas amostras geradas pela predição de consumo, com as amostras coletadas com o medidor, obteve-se uma precisão média e mediana de 97,15% e 97,72% respectivamente, como pode ser visto na Figura 1.

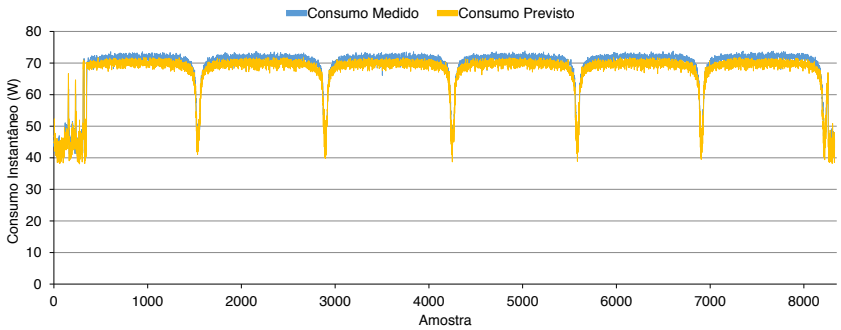


Figura 1. Comparativo do consumo instantâneo entre o consumo previsto e o consumo medido.

Após a execução da aplicação, os dados de CPU e memória principal são utilizados como entrada no modelo gerado, a fim de se obter o consumo de energia instantânea,

medido em Watts (W). A grande vantagem deste modelo é o fato de considerar a elasticidade da nuvem, em outras palavras, o modelo leva em conta apenas o consumo de energia dos recursos que foram efetivamente utilizados, e não o consumo total do *data center*, ou um nó específico. O uso de recursos é coletados de todos os nós durante o tempo de execução da aplicação, e através de um arquivo de log que informa o intervalo de tempo que cada máquina é utilizada, apenas as amostras relativas a execução da aplicação são consideradas para o cálculo do consumo de energia. Este processamento de registro é executado *post-mortem* e permite uma análise mais precisa do consumo de energia da aplicação, e não apenas o consumo de energia de toda a infraestrutura. Este granularidade mais fina permite a utilização de funções de custo, por exemplo, a fim de determinar a viabilidade da utilização da elasticidade em nuvem para executar uma determinada aplicação.

O modelo de consumo também poderia ser empregado para avaliar os ambientes de computação em nuvem heterogêneos, uma vez que é baseado em um modelo já consolidado [Orgerie et al. 2014] apresentado na Equação 1 onde α representa um consumo de energia quando o nó está ocioso e β e δ representam o consumo de energia variável determinado pela quantidade de recursos utilizados (neste caso de CPU e de memória) e retornando o consumo de energia instantâneo em Watts. A única adaptação necessária para contemplar ambientes heterogêneos seria a criação de modelos de consumo de energia distintos para cada tipo de máquina presente no *data center*.

Para complementar esta análise, apresenta-se um conjunto de equações que permitem o cálculo do consumo de energia em ambientes elásticos e também a quantidade de energia gasta por um determinado número de nós. A Equação 2 resulta no consumo de energia de uma máquina m de acordo com o valor de CPU e memória registrados em um instante i, utilizando a Equação 1 como base. A Equação 3 é utilizada para calcular o consumo total de energia de todas as máquinas alocadas em um instante t, ou seja, levando em conta a elasticidade, retornando o consumo em Watts. A Equação 4 calcula o consumo de energia total de um instante 0 a um instante t onde intervalos de tempo são calculados em segundos e utilizando a Equação 3 mencionada anteriormente que já considera a questão elasticidade, este cálculo resulta no consumo de energia em Joules ($W \times segundo$). Finalmente, a Equação 5 apresenta o consumo de energia da aplicação quando utilizando uma quantidade específica de nós representados por z. Este cálculo resulta no consumo total de energia, também representada em Joules, gasto quando utilizando esta quantidade específica de nós.

$$f(CPU, Memoria) = \alpha + \beta \times CPU + \delta \times Memoria \tag{1}$$

$$MC(m,i) = f(CPU(m,i), Memoria(m,i)) \tag{2}$$

$$ETC(t) = \sum_{i=0}^{Maquina} MC(i,t) \times x \begin{cases} x = 0 & \text{se a máquina } i \text{ não está ativa no instante } t; \\ x = 1 & \text{se a máquina } i \text{ está ativa no instante } t. \end{cases} \tag{3}$$

$$TC(t) = \sum_{i=0}^{t} ETC(i) \{ \ 0 \leq \ t \ \leq TempoTotalAplicacao \tag{4}$$

$$NEC(z) = \sum_{i=0}^{TempoApp} ETC(i) \times y \begin{cases} y = 0 & \text{se no instante } i \text{ o total de máquinas ativas} \neq z; \\ y = 1 & \text{se no instante } i \text{ o total de máquinas ativas} = z. \end{cases} \tag{5}$$

3. Análise dos resultados

Os experimentos foram conduzidos utilizando a nuvem privada OpenNebula[1] com 6 nós(1 FrontEnd e 5 nós). As máquinas utilizadas possuem processadores dual-core de 2,9 GHz com 4 GB de memória RAM e uma rede de interconexão de 100 Mbps. Um total de quatro padrões de carga (Crescente, Descendente, Constante e Onda) foram utilizados com o *middleware* AutoElastic [Righi et al. 2015] com e sem o recurso de elasticidade. No caso em que a elasticidade é ativa, os *thresholds* utilizados foram 70% e 90% para o limite superior e 30% e 50% para o limite inferior. Como resultado de uma combinação simples, todas as cargas foram testadas 4 vezes utilizando elasticidade, onde 4 é o número de combinações de *thresholds* superior e inferior selecionadas. Todas as execuções iniciaram a partir do mesmo cenário que consiste em um único nó com duas máquinas virtuais (igual ao número de núcleos da máquina). Nas execuções elásticas, a nuvem pode dimensionar para um limite de cinco nós (10 máquinas virtuais) definida por uma SLA (*Service Level Agreement*). O *middleware* AutoElastic registra quais nós foram utilizados e intervalo de tempo a fim de analisar o consumo de energia de forma elástica posteriormente.

A Figura 2 ilustra o consumo de energia em Watts de acordo com o modelo apresentado quando as ações de elasticidade estão desativadas. Neste contexto, um único nó com duas VMs está sendo utilizado para hospedar os processos escravos. Aqui, podemos observar que o simples fato de ligar o nó computacional (executando o sistema operacional Ubuntu e o *middleware* AutoElastic) consome cerca de 40 Watts. Qualquer computação realizada provoca uma elevação desse índice para o intervalo entre 40 e 71 Watts. Embora a função Crescente cresça lentamente, o consumo de energia aumenta rapidamente para o limite superior do intervalo. O mesmo comportamento aparece nas funções Decrescente e Onda.

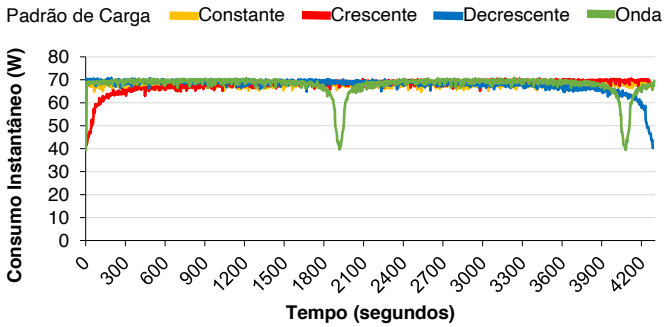
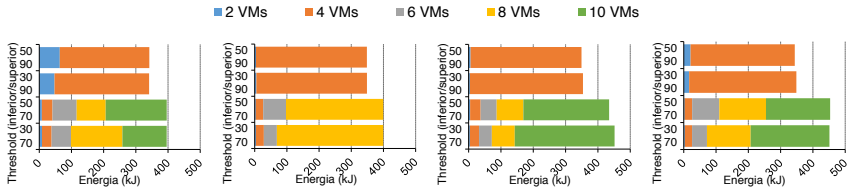


Figura 2. Consumo de energia instantâneo das diferentes cargas sem elasticidade.

Quando a elasticidade é ativada há uma variação elevada na quantidade de VMs utilizada durante a execução da aplicação. A Figura 3 apresenta o perfil de consumo de energia da aplicação de acordo com o número de VMs empregada para resolver o problema e sua contribuição no consumo de energia total (energia consumida até o termino da aplicação) de acordo com a Equação 5. Na Figura 3, o resultado da Equação 5 foi traduzido para VMs considerando que os nós são homogêneos e possuem dois núcleos onde cada VM foi mapeada em um núcleo.

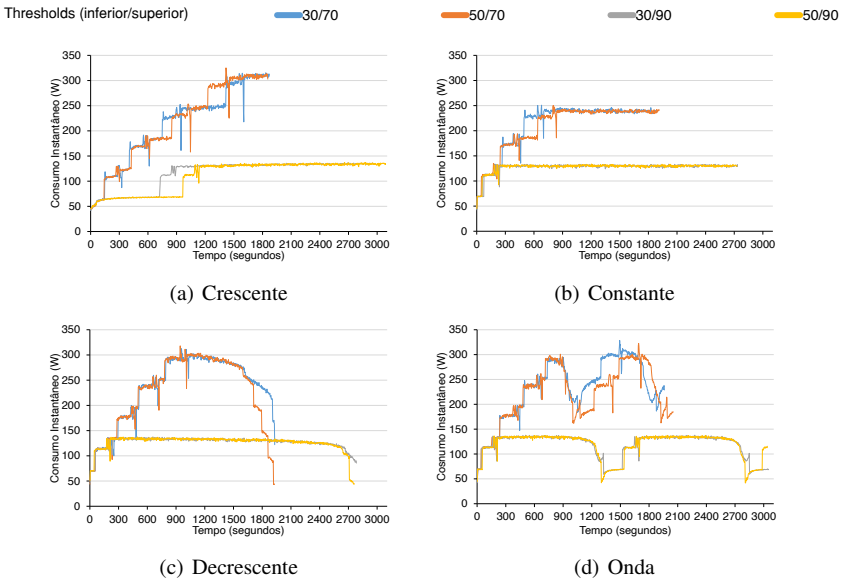
[1]http://opennebula.org/



(a) Crescente (b) Constante (c) Decrescente (d) Onda

Figura 3. Consumo energético para diferentes quantidades de máquinas virtuais e cargas de trabalho variando os thresholds inferior e superior.

Considerando um modelo de consumo de energia que desconsidera a elasticidade o limite inferior do consumo instantâneo seria de 200 W uma vez que o consumo de energia de cada máquina ociosa é de 40 W conforme apresentado na Figura 2 e destacado pelo α na Equação 1. A Figura 4 apresenta o gráfico de execução destacando picos e quedas bruscas de consumo de energia quando se analisa o consumo de energia de forma elástica, utilizando a Equação 3 durante o tempo total de execução da aplicação. Neste gráfico podemos observar alocação e desalocação de *hosts*, além de oscilações durante a inicialização das VMs. Estes gráficos apresentam as vantagens em analisar a aplicação utilizando um modelo elástico, pois considera apenas o consumo de energia das máquinas que executam computação, e representa de forma mais fiel o consumo energético de uma aplicação que faz uso da elasticidade.



(a) Crescente (b) Constante

(c) Decrescente (d) Onda

Figura 4. Comportamento do consumo energético das diferentes cargas de trabalho variando os thresholds inferior e superior.

4. Trabalhos relacionados

Alguns trabalhos concentram-se em modelos para estimar o consumo de energia em ambientes de nuvem, no entanto, estas obras não levam em conta a elasticidade de tais sistemas. Luo et al. [Luo et al. 2013] apresenta um algoritmo de gestão de recursos que considera tanto requisitos de consumo de energia como QoS (Qualidade de Serviço). O artigo apresenta um modelo para prever o consumo de energia dentro de uma única máquina, além de uma estrutura simulada para avaliar algoritmos de escalonamento de recursos que leva em consideração o consumo de energia. Os autores afirmam que na maioria dos estudos de energia de computação em nuvem existentes são utilizados modelos lineares para estimar o consumo de energia, descrevendo a relação entre consumo de energia e utilização de recursos. Garg et al. [Garg et al. 2011] apresenta um modelo de energia do *data center* com base nos dados de CPU. O modelo apresentado considera todas as CPUs no *data center* sem considerar a variação dos recursos disponíveis para a aplicação. Com relação a métricas específicas para estimar o consumo de energia, em [Zikos and Karatza 2011], os autores utilizam a seguinte equação para medir a energia: $E = P \times T$. A quantidade de energia utilizada depende da potência e o tempo no qual é utilizada. Assim, E, P e T, denotam energia, potência e tempo, respectivamente. A unidade padrão para a energia é o joule (J), assumindo que a energia é medida em watts (W) e o tempo em segundos (s).

Considerando a análise do consumo de energia, algumas obras focam em definir perfis de energia [Chen et al. 2014], avaliação de custo e desempenho energético [Tesfatsion et al. 2014]. Feifei et al. [Chen et al. 2014] propõe a StressCloud: uma ferramenta de análise de desempenho e consumo de energia e análise de sistemas em nuvem. Os resultados experimentais demonstram a relação entre o desempenho e o consumo de energia dos sistemas de nuvem com diferentes estratégias de alocação de recursos e cargas de trabalho. No entanto, os autores não abordam nem aplicações paralelas nem elasticidade em nuvem.

Finalmente, Tesfatsion et al. [Tesfatsion et al. 2014] realizar uma análise conjunta de custo e desempenho energético utilizando técnicas como DVFS (*Dynamic Voltage and Frequency Scaling*), a elasticidade horizontal e vertical. Esta abordagem combinada resultou em 34% de economia de energia em comparação com cenários onde cada política é aplicada sozinha.

Em relação ao consumo de energia, o método tradicional que leva em conta o consumo instantâneo e o tempo é normalmente utilizado. Desta forma, destaca-se o seguinte a respeito das métricas de avaliação: (i) a avaliação do consumo de energia, considerando um número maleável de recursos; (ii) em ambientes elásticos, há uma falta de análise conjunta do consumo de energia e a utilização de recursos para definir os valores para os limites de *thresholds* inferiores e superiores.

5. Conclusão

Este artigo apresentou e avaliou um modelo elástico de consumo de energia para *data centers* de computação em nuvem. O modelo proposto estima o consumo de energia com base em amostras de CPU e memória com precisão média e mediana 97,15% e 97,72%, respectivamente. Este modelo foi utilizado em conjunto com o *middleware* AutoElastic, que executa aplicativos HPC, alocando e desalocando recursos de acordo com as demandas dos processos. Os resultados mostraram que os melhores valores para economia de

energia foram obtidos quando se utiliza um limite superior (*threshold*) de cerca de 90%, e os piores valores para essa métrica quando se utiliza 70%. Entretanto, neste último caso obteve-se o melhor desempenho. Focando na reprodutibilidade dos resultados, introduzimos um conjunto de equações que permite que outros pesquisadores possam empregar o modelo energético proposto para medir o consumo de energia em suas aplicações elásticas. Por fim, esta pesquisa deve seguir com a extensão do modelo proposto para incluir máquinas heterogêneas, uma vez que a versão atual assume apenas os nós computacionais e máquinas virtuais com a mesma configuração, e avaliação de consumo energético de *middlewares* para Internet das Coisas.

Referências

Chen, F., Grundy, J., Schneider, J.-G., Yang, Y., and He, Q. (2014). Automated analysis of performance and energy consumption for cloud applications. In *Proceedings of the 5th ACM/SPEC International Conference on Performance Engineering*, ICPE '14, pages 39–50, New York, NY, USA. ACM.

Garg, S. K., Yeo, C. S., Anandasivam, A., and Buyya, R. (2011). Environment-conscious scheduling of hpc applications on distributed cloud-oriented data centers. *J. Parallel Distrib. Comput.*, 71(6):732–749.

Lorido-Botran, T., Miguel-Alonso, J., and Lozano, J. (2014). A review of auto-scaling techniques for elastic applications in cloud environments. *Journal of Grid Computing*, 12(4):559–592.

Luo, L., Wu, W., Tsai, W., Di, D., and Zhang, F. (2013). Simulation of power consumption of cloud data centers. *Simulation Modelling Practice and Theory*, 39(0):152 – 171. S.I.Energy efficiency in grids and clouds.

Orgerie, A.-C., Assuncao, M. D. D., and Lefevre, L. (2014). A survey on techniques for improving the energy efficiency of large-scale distributed systems. *ACM Computing Surveys*, 46(4):1–31.

Petrides, P., Nicolaides, G., and Trancoso, P. (2012). Hpc performance domains on multicore processors with virtualization. In *Proceedings of the 25th International Conference on Architecture of Computing Systems*, ARCS'12, pages 123–134, Berlin, Heidelberg. Springer-Verlag.

Righi, R., Rodrigues, V., Andre daCosta, C., Galante, G., Bona, L., and Ferreto, T. (2015). Autoelastic: Automatic resource elasticity for high performance applications in the cloud. *Cloud Computing, IEEE Transactions on*, PP(99):1–1.

Tesfatsion, S., Wadbro, E., and Tordsson, J. (2014). A combined frequency scaling and application elasticity approach for energy-efficient cloud computing. *Sustainable Computing: Informatics and Systems*, 4(4):205 – 214. Special Issue on Energy Aware Resource Management and Scheduling (EARMS).

Zikos, S. and Karatza, H. D. (2011). Performance and energy aware cluster-level scheduling of compute-intensive jobs with unknown service times. *Simulation Modelling Practice and Theory*, 19(1):239 – 250. Modeling and Performance Analysis of Networking and Collaborative Systems.