

Comparação de Desempenho dos Sistemas de Virtualização KVM e VMware ESXi para Computação de Alto Desempenho*

Willian Soares da Silva, Claudio Schepke, Raul Dias Leiria

¹Laboratório de Estudos Avançados em Computação (LEA)
Universidade Federal do Pampa (UNIPAMPA) – Alegrete – RS – Brasil

{dasilva.williansoares, rdleiria, schepke}@gmail.com

Abstract. *Hypervisors are fundamental technologies for virtualization, since they are responsible for the abstraction of hardware and the provision of virtualized resources for guests. This activity results in overhead. As a result, performance can vary significantly depending on the technology chosen for virtualized environments. This work aims to compare the performance of the KVM and VMware ESXi virtualizers, checking discrepancies and evaluating the results with the performance of the native environment. To carry out the experiments and contribute to the reproducibility of this work, a script was developed. The experiments indicate that KVM has better performance than VMware ESXi.*

Resumo. *Hypervisors são tecnologias fundamentais para a virtualização, pois são os responsáveis pela abstração do hardware e fornecimento de recursos virtualizados para os guests. Esta atividade resulta em overhead. Com isso, o desempenho pode variar significativamente dependendo da tecnologia escolhida para os ambientes virtualizados. O presente trabalho tem como objetivo comparar o desempenho dos virtualizadores KVM e VMware ESXi, verificando discrepâncias e avaliando os resultados com o desempenho do ambiente nativo. Para realizar os experimentos e contribuir com a reprodutibilidade deste trabalho, foi desenvolvido um script. Os experimentos indicam que KVM possui melhor desempenho do que VMware ESXi.*

1. Introdução

A Computação de Alto Desempenho (em inglês, *High Performance Computing* (HPC)) tem por objetivo reduzir o tempo de execução de aplicações que demandam alto poder de processamento [Buyya et al. 2013]. Essas aplicações simulam eventos naturais tais como a previsão do tempo, dinâmica dos fluidos, sequenciamento genético ou sensoriamento remoto [Sterling et al. 2017]. HPC também pode ser utilizado com outras tecnologias, como a virtualização [Radiskhlebova et al. 2019].

A virtualização permite que usuários executem diferentes tipos de sistemas operacionais, em diferentes VMs, em um mesmo servidor físico. Uma das principais características da virtualização é que *workloads* de uma máquina física podem ser divididos em partes para serem executados em vários outros *guests* de (*Virtual Machines*). O *hypervisor*, primeira camada mais perto do *hardware*, é responsável pela virtualização, acrescenta uma sobrecarga, pelo fato de ter que adicionar uma camada de abstração entre o ambiente

*O presente trabalho foi desenvolvido com apoio do PDA 2018 - Iniciação Científica da UNIPAMPA.

virtualizado e os recursos físicos do computador que ele utiliza [Maliszewski et al. 2018]. Vários aspectos do software e sistema operacional podem afetar o desempenho dos virtualizadores e VMs como, por exemplo, a maneira que o hypervisor escala os recursos de CPU, memória, disco, e rede [Hwang et al. 2013].

Existem diferentes *hypervisors* como Oracle VM, Xen, Hyper-V, KVM e VMWare. Cada um deles possui suas próprias características. Cada tipo de tecnologia pode realizar o gerenciamento dos recursos dos computadores de maneira diferente. Essa variedade de *hypervisors* resulta em novos desafios para compreender qual escolher, para o tipo de aplicação que se deseja executar no ambiente virtualizado [Hwang et al. 2013].

De acordo com a literatura, estudos foram desenvolvidos para compreender e avaliar o desempenho das tecnologias de virtualização para HPC, como os *hypervisors* KVM [KVM 2016] e ESXi [VMware 2018]. Contudo, nos trabalhos que avaliam esses *hypervisors*, a reprodutibilidade dos experimentos é desafiadora. Nos trabalhos que comparam esses dois *hypervisors*, alguns experimentos foram realizados em ambientes específicos. Outros trabalhos não comentam sobre a reprodutibilidade dos experimentos. Isso torna esses trabalhos difíceis de serem reproduzidos por outros pesquisadores.

De acordo com [Barker 2016], reprodutibilidade de experimentos é uma das principais formas de se verificar a eficácia/eficiência de contribuições científicas. Neste trabalho foi desenvolvido um *script* que realiza a avaliação dos *hypervisors* KVM e ESXi com o auxílio da suíte de *benchmarks* NAS Parallel Benchmark. O objetivo, é identificar qual *hypervisor* pode ser considerado mais adequado para aplicações científicas.

O presente trabalho está organizado da seguinte maneira. Na Seção 2, são apresentados os trabalhos utilizados como referência para o desenvolvimento deste trabalho. Na Seção 3, é apresentada a metodologia que foi utilizada na avaliação dos experimentos, bem como as partes automatizadas pelo *script* nos experimentos e *parsing* dos resultados. Na Seção 4 são discutidos os resultados obtidos neste trabalho. Por fim, na Seção 5, são discutidas as considerações finais e as direções de trabalhos futuros.

2. Trabalhos Relacionados

Segundo [Hwang et al. 2013], o avanço do desempenho dos processadores e o uso de técnicas de virtualização tem ajudado a reduzir o custo computacional. Porém, ainda ocorre *overhead*, principalmente quando múltiplas VMs estão competindo por recursos.

Para obter o melhor desempenho das aplicações, nos ambientes virtualizados, é importante escolher qual *hypervisor* será utilizado para determinadas aplicações, pois é ele que provê o ambiente de *hardware* virtualizado para possibilitar alocar múltiplas VMs a partir de um computador físico. Conforme [Elsayed and Abdelbaki 2013], o *hypervisor* utilizado influencia no desempenho do ambiente de virtualização.

A proposta deste trabalho é tornar a avaliação do desempenho do ambiente de virtualização reprodutível, ao desenvolver um *script* para automatizar a avaliação dos *hypervisors* KVM e ESXi. Apesar da literatura contemplar trabalhos com esses *hypervisors* que mencionam a reprodutibilidade de experimentos [Guzek et al. 2014] e [Varrette et al. 2013], eles foram executados em ambientes específicos. Outros, trabalhos que realizam a comparação desses *hypervisors* não mencionam a reprodutibilidade, [Reddy and Rajamani 2014] [Manik and Arora 2016].

Tabela 1. Comparativo dos Trabalhos Relacionados.

Autores	Plataformas	Workloads	Automatização
RADISKHLEBOVA, A. A. <i>et al.</i> 2019	LXC, Docker, vSphere	NC	NC
MALISZEWSKI, A. M. <i>et al.</i> 2018	KVM, LXC	NPB-OMP	NC
VOGEL, A. <i>et al.</i> 2016	KVM	NPB-33, MPI	NC
GRANISZEWSKI, W. <i>et al.</i> 2016	Hyper-V, ESXi, OVM, VirtualBox, Xen	NBENCH, Netperf, Filebench, Ramspeed	NC
MORABITO, R. <i>et al.</i> 2015	KVM, LXC, Docker, OSv	Aplicação real, Y-cruncher, NBENCH, Linpack, Bonnie++, Netperf	NC
HWANG, J. <i>et al.</i> 2013	Hyper-V, vSphere, Xen, KVM,	Aplicação real, Bytemark, Ramspeed, Bonnie++, FileBench, Netperf	NC
ELSAYED, A. <i>et al.</i> 2013	VMware ESXi5, Citrix Xen Server 602, Hyper-V2008R2	Aplicação real, Aplicações sintéticas, PRTG <i>network monitor tools</i>	NC
Este artigo	Linux KVM, VMWare ESXi	NPB-MPI	BASH, Python

A Tabela 1 apresenta os artigos utilizados como referência neste trabalho. Para cada artigo são apresentadas as plataformas avaliadas com seus respectivos *workloads* e se há algum tipo de automatização. Dos trabalhos relacionados, 4 trabalhos realizam a avaliação de desempenho exclusivamente de *hypervisors*: [Elsayed and Abdelbaki 2013], [Graniszewski and Arciszewski 2016], [Hwang et al. 2013] e [Vogel et al. 2016]; e 3 trabalhos entre *hypervisors* e a tecnologia de virtualização leve (*lightweight*) contêineres: [Radiskhlebova et al. 2019], [Maliszewski et al. 2018] e [Morabito et al. 2015]. Em relação às plataformas escolhidas, pode-se observar um conjunto bem distinto sendo avaliado pelos diferentes trabalhos. Já no quesito *workloads*, há um conjunto distinto também, com predomínio de aplicações reais e benchmark consolidados. Nenhum dos trabalhos avaliados apresenta algum recurso de automatização dos testes.

Em nosso trabalho propomos a avaliação de desempenho dos *hypervisors* KVM e ESXi, a fim de verificar o *overhead* causado por essas tecnologias no ambiente de HPC. Para isso, foi proposto o desenvolvimento de um *script* para automatizar os experimentos.

3. Metodologia

De acordo com os artigos utilizados como referência neste trabalho não constam experimentos comparando os *hypervisors* GNU/Linux KVM e o VMware (ESXi), com foco na reprodutibilidade de experimentos. A identificação do *hypervisor* mais adequado para aplicações científicas em HPC contribuirá com a literatura. Outra motivação da escolhas dessas tecnologias é que KVM é *opensource*, enquanto VMware não.

Para a avaliação de desempenho foi escolhida a suíte de *benchmarks* NAS Parallel Benchmark 3.3.1 (NAS-PB). Um *script* de automatização foi desenvolvido para auxiliar na fase de realização dos experimentos e para contribuir com a reprodutibilidade do trabalho. Por fim, é possível gerar os gráficos a partir dos resultados obtidos com a execução do *script*.

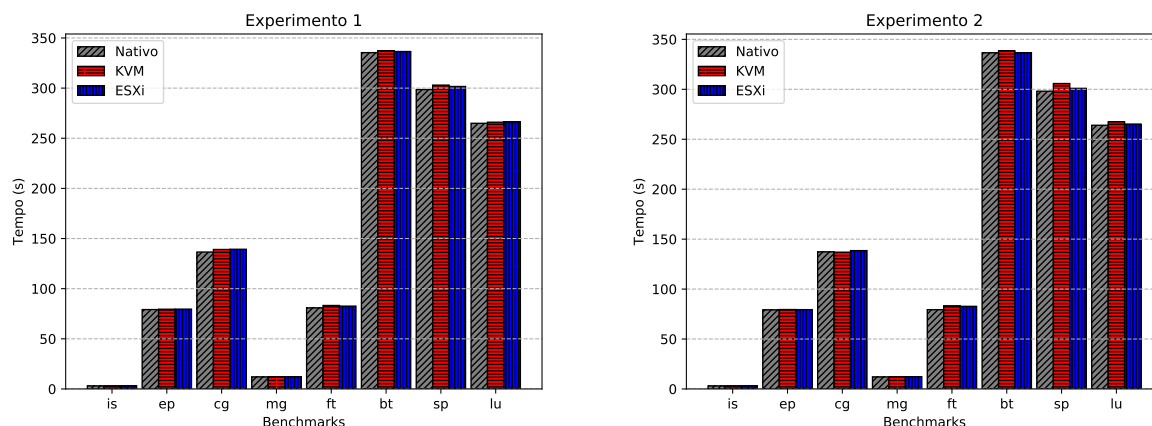


Figura 1. (a) Experimento 1 - 1 Nó, 1 CPU, 4 GB RAM. (b) Experimento 2 - 1 Nó, 1 CPU, 16 GB RAM.

Durante o desenvolvimento deste trabalho foram utilizados diferentes tipos de tecnologias, como: linguagens de programação, *benchmarks* e sistemas operacionais. Foi criado um repositório textual na plataforma GitHub (github.com/RDLeiria/WillianSoares). No repositório encontram-se informações sobre como este trabalho foi desenvolvido e a disponibilização dos *scripts* de testes.

O ambiente de execução utilizado é um servidor físico executando Ubuntu Server 18.04.2 como sistema operacional de base. O servidor é composto por um processador Intel Xeon E5-2650 com 6 cores, operando em 1.9 GHz de frequência, com suporte a *Hyper-Threading* desabilitado intencionalmente para a realização dos experimentos. A memória principal é de 16 GB DDR4. Os discos rígidos possuem capacidade máxima de armazenamento de 500 GB e 7200 RPM, sendo os modelos e fabricante os mesmos.

4. Resultados Experimentais

Com o objetivo de identificar qual *hypervisor* pode ser considerado mais adequado para ambientes de HPC foram gerados gráficos que comparam os resultados obtidos. Os gráficos apresentam o tempo médio de execução dos ambientes (Nativo, KVM e ESXi) com diferentes cenários de execuções. Para identificar os ambientes avaliados, variou-se a quantidade de nós, CPUs e memória RAM. Nos resultados dos gráficos para o ambiente nativo foi utilizada a cor cinza, a cor vermelha para o KVM e a cor azul para o ESXi.

No Experimento 1 e Experimento 2 foram utilizados apenas 1 nó nos virtualizadores e 1 CPU. A memória RAM foi configurada para 4 GB no Experimento 1 e 16 GB para o Experimento 2. Os gráficos estão representados pela Figura 1 e demonstram o tempo médio das execuções dos *benchmarks*.

Os resultados obtidos no Experimento 1 indicam que o *hypervisor* KVM teve melhor desempenho nas execuções dos *benchmarks* EP, CG e LU. Estes utilizam menos memória ao executar quando comparado aos demais *benchmarks*. O *hypervisor* ESXi teve melhor desempenho que KVM ao executar os *benchmarks* IS, FT, BT e SP. Esses *benchmarks* utilizam maior quantidade de memória para comunicação. O Experimento 1 demonstra que o ESXi tem melhor desempenho.

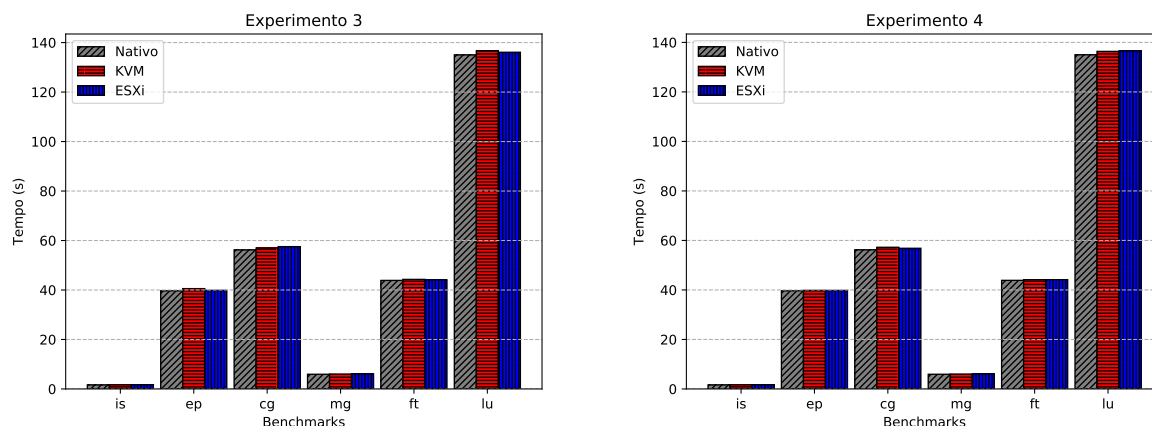


Figura 2. (a) Experimento 3 - 1 Nó, 2 CPUs, 4 GB RAM. (b) Experimento 4 - 1 Nó, 2 CPUs, 16 GB RAM.

De acordo com os resultados obtidos no Experimento 2 KVM teve melhor desempenho na execução do *benchmark* MG. Nos demais casos, executando os *benchmarks* IS, EP, FT, BT, SP e LU, ESXi teve desempenho superior ao KVM. Nesse cenário, o nó com o ESXi também foi melhor na utilização de CPU e memória. Foi possível também identificar uma anomalia ao avaliar o *hypervisor* KVM. O tempo médio de execução do *benchmark* CG foi menor que tempo obtido no ambiente nativo. Isso pode ocorrer porque esse *benchmark* necessita de uma quantidade baixa de memória [Subhlok et al. 2001].

O Experimento 3 e Experimento 4 foram executados utilizando 1 nó com 2 CPUs. O tamanho da memória RAM foi configurado para 4 GB no Experimento 3 e 16 GB no Experimento 4. Alguns *benchmarks* (SP e BT) da suíte NAS-PB não executam com uma determinada quantidade de CPUs pois necessitam um número quadrático de CPUs para executar [Bailey et al. 1995]. Por esse motivo, o tempo médio de execução desses *benchmarks* não está presente nos gráficos do Experimento 3 e Experimento 4. A Figura 2 representa o tempo médio das execuções de cada *benchmark* nestes 2 cenários.

Os resultados obtidos no Experimento 3 indicam que KVM teve melhor desempenho ao executar os *benchmarks* MG, FT e LU, que realizam a comunicação *ring pattern*. O *hypervisor* ESXi teve melhor desempenho ao executar os *benchmarks* IS, EP e CG. IS realiza a comunicação *all-all*, no EP a comunicação é considerada insignificante e no CG a comunicação é unidirecional [Subhlok et al. 2001].

Nos resultados obtidos no Experimento 4, KVM teve melhor desempenho nas execuções dos *benchmarks* MG, FT, e LU. Nas execuções dos outros *benchmarks* IS, EP e CG o *hypervisor* ESXi teve melhor desempenho. O Experimento 3 e Experimento 4 não sofrem influência do tamanho da memória principal, pois KVM teve melhor desempenho nos *benchmarks* que realizam a comunicação *ring pattern* e ESXi nos demais casos.

A Figura 3 representa a média dos tempos de execução dos *benchmarks* obtidos no Experimento 5 e Experimento 6.

O nó utilizado no Experimento 5 e Experimento 6 foi configurado com 4 CPUs. Nos resultados obtidos no Experimento 5, o *hypervisor* KVM teve melhor desempenho nas execuções dos *benchmarks* IS, EP, CG, SP e LU. Todos esses *benchmarks* mantêm

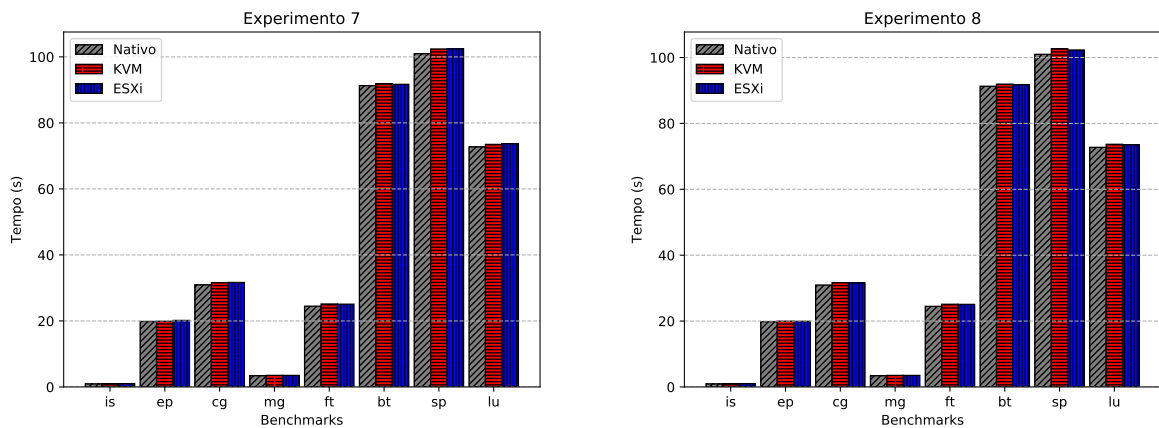


Figura 3. (a) Experimento 5 - 1 Nó, 4 CPUs, 4 GB RAM. (b) Experimento 6 - 1 Nó, 4 CPUs, 16 GB RAM.

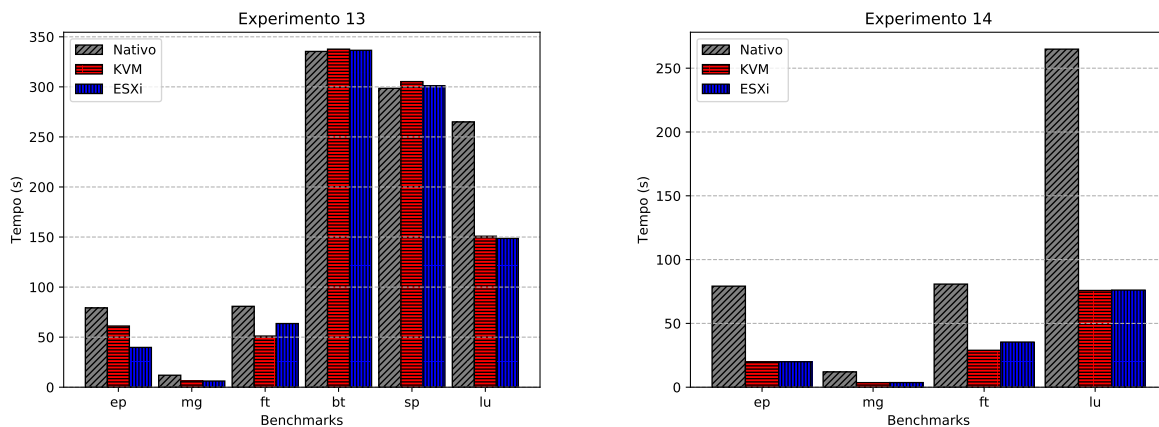


Figura 4. (a) Experimento 7 - 2 Nós, 1 CPU (cada nó), 4 GB RAM. (b) Experimento 8 - 2 Nós, 2 CPUs (cada nó), 4 GB RAM.

a execução da CPU perto do 100% e utilizam uma quantidade menor de memória que os *benchmarks* MG e BT [Subhlok et al. 2001]. ESXi teve melhor desempenho ao executar os *benchmarks* MG, FT e BT. Neste experimento, KVM tem melhor desempenho quando é necessário utilizar a CPU e ESXi quando é necessário utilizar a memória. Nos resultados obtidos no Experimento 6, o *hypervisor* ESXi teve melhor desempenho ao executar todos os *benchmarks* da suíte NAS-PB. A adição de memória RAM teve impacto no desempenho do *hypervisor* ESXi, melhorando seu desempenho em todos os *benchmarks*.

No Experimento 7 e Experimento 8 os resultados estão representados pela Figura 4. No Experimento 7 foram criados dois nós com configurações de *hardware* iguais: 1 CPU e 4 GB de memória RAM. O *hypervisor* que obteve melhor desempenho foi o ESXi, comparado com o ambiente nativo. O *overhead* também foi menor nas execuções dos *benchmarks* BT e SP, comparados ao KVM. KVM teve melhor desempenho que ESXi ao executar o *benchmark* FT. O experimento revelou que o ESXi tem melhor desempenho com múltiplos nós, com a utilização eficiente dos recursos.

No Experimento 8 o *hypervisor* KVM teve melhor desempenho ao executar os *benchmarks* MG, BT, SP e LU. Neste experimento, KVM superou os resultados obtidos pelo ESXi, que teve desempenho superior ao executar os *benchmarks* EP e FT. O aumento no número de cores nos nós melhora o desempenho do *hypervisor* KVM.

5. Conclusão

O presente trabalho teve como objetivo realizar a comparação dos *hypervisors* GNU/Linux KVM e VMware ESXi. Apesar da literatura abordar trabalhos que realizam a comparação de desempenho de diferentes tecnologias de virtualização, não foram encontrados trabalhos que comparem esses dois *hypervisors* e que sejam reprodutíveis. Os *hypervisors* KVM e ESXi foram avaliados no ambiente de HPC, para identificar qual tecnologia pode ser considerada mais adequada na execução de aplicações científicas. A avaliação foi realizada com o auxílio de um *script*, desenvolvido para automatizar a fase de realização dos experimentos e contribuir com a reprodutibilidade do trabalho.

Algumas etapas dos experimentos presentes neste trabalho não ficaram totalmente reprodutíveis. Aspectos como a criação e configuração de cenários (n VMs) e criação automatizada dos gráficos precisam ser revisadas em trabalhos futuros. Por outro lado, foram descritos detalhadamente os processos de desenvolvimento deste trabalho para que seja possível compreender as metodologias utilizadas e como os resultados foram obtidos.

A partir dos experimentos realizados nos cenários, ambas tecnologias podem ser consideradas para utilização para virtualização em HPC. Algumas diferenças de tempos de execuções obtidas nos experimentos podem parecer irrelevantes. Porém se considerado a utilização em larga escala, pode haver uma diferença significativa na escolha do *hypervisor*. Para os testes realizados neste trabalho, os experimentos demonstraram que KVM foi o virtualizador que teve melhor desempenho devido as suas características de virtualização.

Referências

- [Bailey et al. 1995] Bailey, D., Harris, T., Saphir, W., Van Der Wijngaart, R., Woo, A., and Yarrow, M. (1995). The nas parallel benchmarks 2.0. Technical report, Technical Report NAS-95-020, NASA Ames Research Center.
- [Barker 2016] Barker, M. (2016). 1,500 scientists lift the lid on reproducibility. [Online].
- [Buyya et al. 2013] Buyya, R., Vecchiola, C., and Selvi, S. T. (2013). *Mastering cloud computing: foundations and applications programming*. Newnes.
- [Elsayed and Abdelbaki 2013] Elsayed, A. and Abdelbaki, N. (2013). Performance evaluation and comparison of the top market virtualization hypervisors. In *Computer Engineering & Systems (ICCES), 2013 8th International Conference on*, pages 45–50.
- [Graniszewski and Arciszewski 2016] Graniszewski, W. and Arciszewski, A. (2016). Performance analysis of selected hypervisors (virtual machine monitors-vmms). *International Journal of Electronics and Telecommunications*, 62(3):231–236.
- [Guzek et al. 2014] Guzek, M., Varrette, S., Plugaru, V., Pecero, J. E., and Bouvry, P. (2014). A holistic model of the performance and the energy efficiency of hypervisors in a high-performance computing environment. *Concurrency and Computation: Practice and Experience*, 26(15):2569–2590.

- [Hwang et al. 2013] Hwang, J., Zeng, S., y Wu, F., and Wood, T. (2013). A component-based performance comparison of four hypervisors. In *Integrated Network Management (IM 2013), 2013 IFIP/IEEE International Symposium on*, pages 269–276. IEEE.
- [KVM 2016] KVM (2016). Main Page — KVM, . [Online: https://www.linux-kvm.org/index.php?title=Main_Page&oldid=173792].
- [Maliszewski et al. 2018] Maliszewski, A. M., Griebler, D., Schepke, C., Ditter, A., Fey, D., and Fernandes, L. G. (2018). The nas benchmark kernels for single and multi-tenant cloud instances with lxc/kvm. In *International Conference on High Performance Computing & Simulation (HPCS)*. IEEE.
- [Manik and Arora 2016] Manik, V. K. and Arora, D. (2016). Performance comparison of commercial vmm: Esxi, xen, hyper-v & kvm. In *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 1771–1775.
- [Morabito et al. 2015] Morabito, R., Kjällman, J., and Komu, M. (2015). Hypervisors vs. lightweight virtualization: a performance comparison. In *Cloud Engineering (IC2E), 2015 IEEE International Conference on*, pages 386–393. IEEE.
- [Radishkhebova et al. 2019] Radishkhebova, A. A., Vavrenyuk, A. B., Rusakova, A. S., and Makarov, V. V. (2019). Study of the possibilities of using virtualization tools to optimize the cluster resources management. In *2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, pages 310–314.
- [Reddy and Rajamani 2014] Reddy, P. V. V. and Rajamani, L. (2014). Performance evaluation of hypervisors in the private cloud based on system information using sigar framework and for system workloads using passmark. *International Journal of Advanced Science and Technology*, 70:17–32.
- [Sterling et al. 2017] Sterling, T., Anderson, M., and Brodowicz, M. (2017). *High Performance Computing: Modern Systems and Practices*. Morgan Kaufmann.
- [Subhlok et al. 2001] Subhlok, J., Venkataramaiah, S., and Singh, A. (2001). Characterizing nas benchmark performance on shared heterogeneous networks. In *Proceedings 16th International Parallel and Distributed Processing Symposium*, pages 9–pp. IEEE.
- [Varrette et al. 2013] Varrette, S., Guzek, M., Plugaru, V., Besson, X., and Bouvry, P. (2013). Hpc performance and energy-efficiency of xen, kvm and vmware hypervisors. In *2013 25th International Symposium on Computer Architecture and High Performance Computing*, pages 89–96. IEEE.
- [VMware 2018] VMware (2018). VMware vsphere hypervisor gratuito, virtualização gratuita (esxi). [Online; accessed 5-June-2018].
- [Vogel et al. 2016] Vogel, A., Maron, C. A., Griebler, D., and Schepke, C. (2016). Medindo o desempenho de implantaç oes de openstack, cloudstack e opennebula em aplicações científicas. *16th Escola Regional de Alto Desempenho do Estado do Rio Grande do Sul (ERAD/RS)*, pages 279–282.