

# Avaliação do Desempenho do Protocolo Bonding em Máquinas Virtuais LXC e KVM

Maikel Jean Klein Christ<sup>1</sup>, Anderson Mattheus Maliszewski<sup>1</sup>, Dalvan Griebler<sup>1,2</sup>

<sup>1</sup> Laboratório de Pesquisas Avançadas para Computação em Nuvem (LARCC)  
Faculdade Três de Maio (SETREM) – Três de Maio – RS – Brasil

<sup>2</sup> Programa de Pós-Graduação em Ciência da Computação  
Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)  
Porto Alegre – RS – Brasil

maikelchrist@hotmail.com, andersonmaliszewski@gmail.com

**Resumo.** *O processamento de grandes volumes de dados (Big Data) e seu armazenamento distribuído vem aumentando gradualmente o uso da rede. Com isso, torna-se necessário o uso de tecnologias para otimizar a largura de banda. Uma das soluções de baixo custo e fácil implementação é a agregação de link. Além disso, a virtualização, usada como base na computação em nuvem, oferece vários benefícios utilizados no Big Data. O objetivo deste trabalho é avaliar o desempenho de rede usando a agregação de link com o protocolo bonding em máquinas virtuais LXC e KVM. Os resultados mostram que o protocolo bonding tem comportamento similar com ambos tipos de virtualização.*

## 1. Introdução

Nos paradigmas atuais, novas tendências tecnológicas surgem com o objetivo de prover serviços com melhores benefícios e desempenho superior. O *Big Data*, um paradigma disruptivo, representa uma nova era de exploração de dados que são utilizados nas mais variadas áreas, a fim de otimizar soluções [Mayer-Schönberger and Cukier 2013]. Esta tecnologia requer *clusters* com alta velocidade de comunicação para o processamento de grandes volumes de dados, além de alta disponibilidade e processamento [INTEL 2013]. Outros requisitos necessários para o *Big Data* são a flexibilidade e elasticidade, fornecidas pela computação em nuvem, permitindo que os servidores possam ampliar ou reduzir suas VMs.

Estima-se que o tráfego IP global em 2015 alcançou 245 terabytes por segundo [CISCO 2013]. A partir desses dados, percebe-se que a conexão entre as computadores será cada vez mais explorada, fazendo-se necessárias a criação de otimizações da rede. Existem várias técnicas voltadas para explorar o potencial da rede, sendo a agregação de link que utiliza o padrão IEEE 802.3ad uma delas e consiste em uma solução que combina múltiplas conexões de rede paralelas em um único link lógico. Seus benefícios incluem o aumento da largura de banda e oferece redundância a falhas em determinado link [Aust et al. 2006].

Um dos protocolos utilizados para criar a agregação de link é o *bonding*, que busca a combinação de uma série de portas físicas para criar um único link lógico de dados. Dependendo da quantidade e capacidade dos links físicos, este protocolo consegue oferecer

um link de alta largura de banda, compartilhando a carga de tráfego entre as portas delegadas. Neste trabalho, utilizamos o protocolo *bonding* para realizar a agregação de link, máquinas virtuais baseadas em KVM (*Kernel-based Virtual Machine*) e LXC (*Linux Containers*) que representam a camada de virtualização, utilizada como base na computação em nuvem e o *benchmark* NetPIPE, para avaliação da rede.

O trabalho tem como objetivo realizar uma avaliação de desempenho em agregação de *link* utilizando o protocolo *bonding* em VMs KVM e LXC, e dar continuidade ao trabalho de Rista et al., o qual aborda o alto desempenho do ambiente em nuvem, implantando instâncias baseadas em contêiner, nas quais aplicações de *Big Data* foram executadas [Rista et al. 2017]. As principais contribuições desse trabalho estão sumarizadas abaixo:

- Avaliação de desempenho do protocolo *bonding* na agregação das interfaces físicas.
- Avaliação de máquinas virtuais usando virtualizadores LXC e KVM.

Este artigo está dividido em 5 seções. A Seção 2 apresenta os estudos relacionados. Na Seção 3 é apresentado o *background* sobre agregação de *link*, LACP, *bonding* e virtualização. Os resultados de desempenho são descritos na seção 4. Por fim, na Seção 5, apresenta-se a conclusão e os trabalhos futuros.

## 2. Trabalhos Relacionados

Nesta seção serão discutidos os trabalhos relacionados que utilizaram modelos ou metodologias semelhantes a este trabalho, além de enfatizar as contribuições contrastes.

O artigo de Rista et al. executou aplicações de *Big Data* na rede, utilizando instâncias de nuvem baseadas em contêineres. Os resultados demonstraram uma redução no tempo de execução das aplicações de 33,73% [Rista et al. 2017]. O estudo deste trabalho relaciona-se com o presente trabalho na busca de atender a necessidade da utilização das experimentações utilizando os protocolos IEEE 802.3ad, testando seu desempenho em ambiente real e virtualizadores. Entretanto, este artigo difere-se na avaliação em máquinas virtuais utilizando o KVM.

Matteussi et al. realizaram uma avaliação de desempenho em agregações de *link* utilizando o algoritmo *Round Robin*, o qual possibilita agregar interfaces de rede transmitindo os pacotes em ordem sequencial. Os resultados demonstraram que quando foram realizadas agregações de 8 *links*, houve a perda de 18% de *throughput*, fazendo-se necessário a realização de ajustes diretamente no *kernel* do sistema, para então, otimizar o *throughput* em 25% [Matteussi et al. 2014]. O artigo trouxe como contribuição a necessidade de modificação de alguns parâmetros do *kernel*, tais como alteração do *buffer*. Por outro lado, este artigo difere-se na avaliação em que engloba os ambientes de máquinas virtuais KVM e LXC.

Vogel et al. apresentaram uma avaliação de desempenho para otimizações em cargas de trabalho com uso intensivo de rede. Os resultados apontaram uma degradação de desempenho relacionados a plataforma de gerenciamento de nuvem CloudStack usando o KVM. Por outro lado, com a utilização do LXC, obteve-se resultados próximos ao ambiente nativo [Vogel et al. 2017]. A contribuição do artigo está na apresentação de uma avaliação de desempenho de rede e uma comparação em relação ao *throughput* de

TCP, latência e número de conexões por segundo. Em contraste, este artigo está focado na utilização de agregação de *link* utilizando o protocolo *bonding*.

Em Imaizumi et al. foi proposto um mecanismo de economia de energia baseado em agregação de *link* 802.3ad, o qual reduziu o consumo energético em 25,4%. Sua contribuição é representada na agregação de dois meios para otimizações de *throughput* e redução do consumo energético [Imaizumi et al. 2009]. Diferentemente, neste artigo o foco baseia-se na realização e avaliação da agregação de *link* utilizando o *bonding* para máquinas virtuais KVM e LXC.

Avaliando uma conexão ponta a ponta, Brassil et al. realizou uma análise dos pacotes TCP, os quais foram transmitidos entre vários *links*. O modo de configuração *Round Robin* foi responsável por direcionar os pacotes a cada interface, porém não foram obtidos os resultados esperados, pois o *throughput* alcançado foi de apenas 141 Mbps em duas placas de rede *Megabit* [Brassil 2005]. Em contrapartida, este trabalho é focado na avaliação da agregação de *links* com máquinas virtuais KVM e LXC.

### 3. Background

Nesta seção serão descritos a agregação de *link*, o LACP - *Link Agregation Control Protocol*, protocolo *bonding* e a virtualização baseada em *kernel* e em contêiner.

#### 3.1. Agregação de Link

A agregação de *link* implementa vários métodos para combinar múltiplas conexões de rede. Pode ser usado tanto com a finalidade de aumentar o rendimento, quanto para fornecer redundância em caso de falha em algum *link* ou ambos. Um exemplo é o método aplicado pelo protocolo *bonding*, que busca a combinação de uma série de portas físicas para criar um único *link* lógico de dados. Dependendo da quantidade e capacidade dos *links* físicos, este protocolo consegue oferecer um *link* de alta largura de banda, compartilhando a carga de tráfego entre as portas delegadas. Na Figura 1 é exibida a visão geral da agregação de *link*.

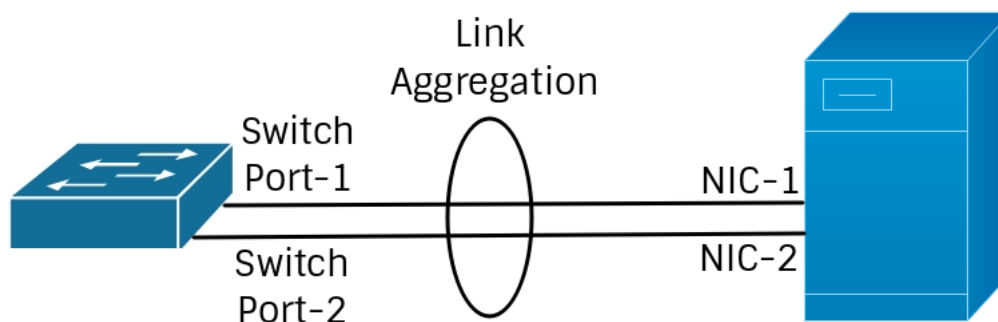


Figura 1. Visão geral da agregação de *link*.

#### 3.2. LACP - *Link Agregation Control Protocol*

Para controlar a agregação de *link*, o LACP permite que um dispositivo de rede controle o agrupamento automático de *links* enviando pacotes LACP para o dispositivo onde está implementado este protocolo.

A definição de LACP (*Link Aggregation Control Protocol*) consiste em um algoritmo constituído por máquinas de estados e funções, que torna possível através de trocas de pacotes, combinar diversos enlaces físicos em um único enlace lógico. Vários benefícios compõem esta técnica, tais como aumento na largura de banda, redundância e balanceamento de carga nos enlaces físicos [Bondan et al. 2012].

Este protocolo apresenta algumas limitações que afetam seu funcionamento, como por exemplo; somente pode ser executado em agregações ponto-a-ponto; não suporta MACs fora do padrão IEEE 802.3; os dados somente trafegam em um sentido no canal; todos os dados devem somente operar na mesma taxa de transferência. A funcionalidade do LACP restringe-se a *switches* e servidores, onde os dois lados da conexão devem conter suporte ao protocolo.

O LACP faz parte da especificação IEEE 802.3ad que permite o agrupamento de várias portas físicas para formar um único canal lógico [Sarabando 2008]. Quando alterado o número de portas agrupadas ativas em um canal, os padrões de tráfego refletem o estado reequilibrado ao *link*, suportando a criação automática de canais com portas Gigabit Ethernet para então, realizar a troca de pacotes entre estas.

### 3.3. Protocolo *Bonding*

*Bonding* é um protocolo de agregação de *link* que tem como principal característica agregar diversas interfaces físicas em uma única interface lógica, aumentando a largura de banda, oferecendo redundância e disponibilidade do *link*. Este protocolo atua no controle de camada 2 e pode ser utilizado para detectar automaticamente, configurar, e gerir uma ligação lógica única com várias ligações físicas entre os dois dispositivos adjacentes. Assim, agregação de *link* fornece maior disponibilidade e capacidade, enquanto melhorias de desempenho de rede são obtidos utilizando o *hardware* existente [Coutinho et al. 2013].

No nível de *hardware*, o tráfego é distribuído em paralelo entre portas correspondentes, atuando logicamente como um único *link*, somando a largura de banda de todas as interfaces. Atualmente, permite-se conexões entre 2 e 16 canais físicos em um único canal virtual para aumentar a largura de banda. Os dispositivos que compõem o núcleo são interligados entre múltiplos canais, permitindo melhor roteamento e alta disponibilidade. Quando algum *link* redundante falha, os pacotes podem ser enviados para os *links* restantes compostos pela agregação de forma transparente para as demais interfaces da rede.

### 3.4. Virtualização em *Kernel* e em Contêiner

A virtualização baseada em *Kernel* (KVM) tem como característica o código aberto, trabalhando com o *hardware* disponível e oferece um ambiente isolado para cada VM. Quando um módulo KVM é instalado, o *kernel* do Linux é assumido como um *hypervisor* [Chiramal et al. 2016]. Na Figura 2 temos uma visão geral em alto nível sobre as estruturas do KVM e LXC.

Na primeira camada temos os componentes físicos, responsáveis por fornecer os recursos para as camadas superiores (tanto para a virtualização de *kernel* como para virtualização em contêiner). Na segunda camada do KVM, apresenta-se o Libvirt, que é uma coleção de *softwares* responsáveis pelo fornecimento do gerenciamento das máquinas

virtuais de uma forma sutil aos recursos de virtualização, tais como armazenamento e gerenciamento de comandos [Hat 2018]. Além disso, temos os *drivers* e módulos do KVM, as *bridges* responsáveis pela rede, o VirtIO que é utilizado para reduzir a sobrecarga associada ao QEMU e melhorar o processamento com grande desempenho e o QEMU responsável por auxiliar o *kernel* do KVM a criar máquinas virtuais, sendo possível ser utilizado como um emulador de máquina, no qual executa os diferentes sistemas operacionais em diferentes máquinas.

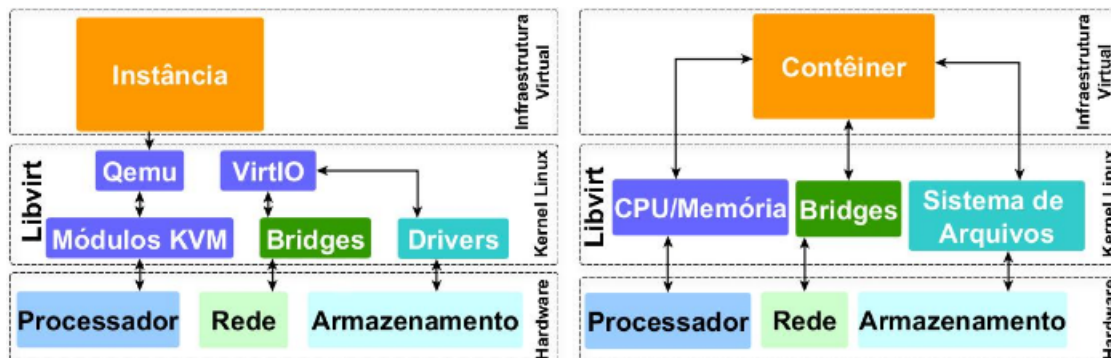


Figura 2. KVM e LXC Visão Geral [Vogel et al. 2017].

A virtualização em contêiner (LXC), por outro lado, compartilha o mesmo núcleo que o *kernel* da máquina, para criar diferentes contêineres usando *scripts* na inicialização de sistemas específicos. Sua estrutura apresentada na Figura 2, mostra-se simplificada em relação ao KVM. Este método de virtualização atua em nível de sistema operacional permitindo executar múltiplos sistemas denominados *contêineres*, que são separados do SO através de *cgroups*, subsistemas de grupos de controle do *Linux* usados para agrupar processos e gerenciar seu consumo agregado de recursos, e *namespaces*, usados para criar e isolar os objetos dentro dos contêineres. Além disso, a conexão de rede geralmente utiliza *bridges* ou a interface nativa.

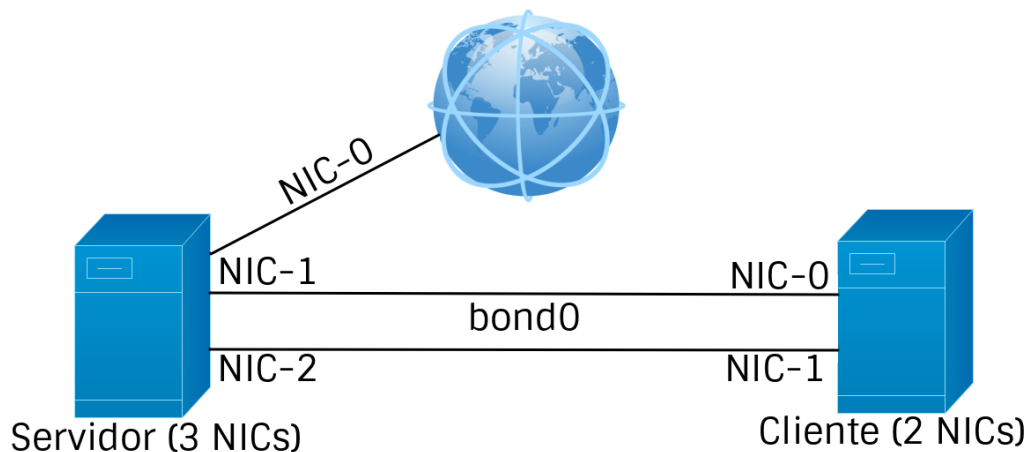
## 4. Avaliação de Desempenho

No decorrer deste capítulo serão discutidos os resultados obtidos, o ambiente de testes, as metodologias seguidas e o *benchmark* utilizado na realização dos testes.

### 4.1. Ambiente de Testes

Com o objetivo de avaliar o comportamento da agregação de *link* do protocolo *bonding*, este trabalho implantou o protocolo com modo 0 ou *Balance-rr (Round-robin)* em ambiente nativo, posteriormente em ambientes virtualizados com LXC e KVM, com a finalidade de reproduzir o cenário ideal para correta coleta de métricas através da utilização do *benchmark* NetPipe. Os testes aplicaram-se sobre uma topologia simplificada interligando dois *hosts*. Na Figura 3 mostra-se a metodologia utilizada para realização dos testes.

Utilizou-se duas máquinas, uma configurada como servidor e uma configurada como cliente. A máquina configurada como servidor dispõe de três interfaces de rede, cada interface sendo Gigabit. Uma das interfaces está configurada para acessar a Internet, e as restantes para a utilização do protocolo. Já a máquina configurada como cliente



**Figura 3. Metodologia utilizada na realização dos testes.**

dispõe de duas interfaces de rede, ambas destinadas a utilização do protocolo *bonding*. O sistema operacional utilizado para esta pesquisa é o Ubuntu Server 16.04. Na Tabela 1 abaixo apresenta-se a especificação do *hardware* de cada *host*.

	Processador	Placas de Rede	Memória RAM
Servidor	Intel® Core™ i5-650 Processor - 3.20 GHz	3 x Gigabit Ethernet	4 GB
Cliente	Intel® Core™ i5-650 Processor - 3.20 GHz	2 x Gigabit Ethernet	4 GB

**Tabela 1. Especificações de Hardware.**

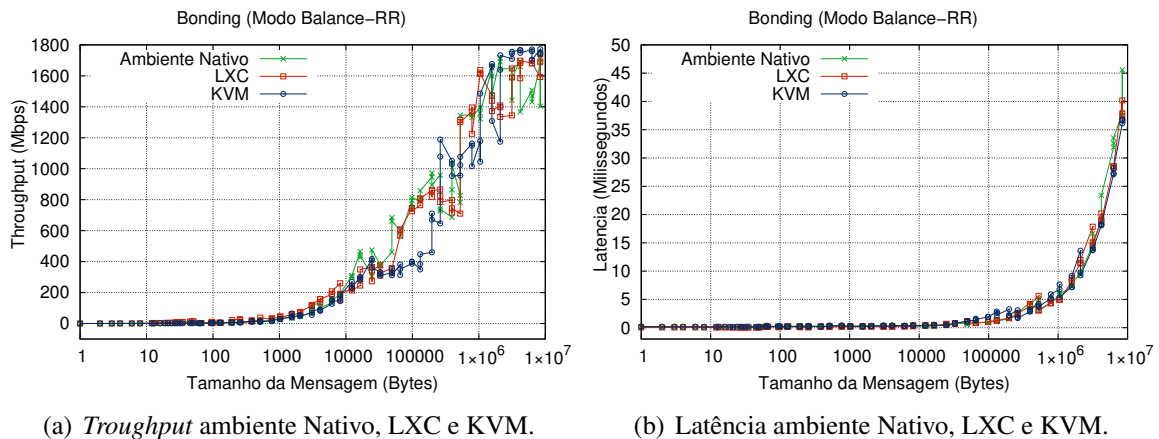
#### 4.1.1. Comparação do Desempenho

A comparação dos resultados se inicia com a Figura 4(a), a qual contém o *throughput* do ambiente nativo, KVM e LXC, onde todos os ambientes alcançaram aproximadamente o mesmo *throughput* até quando o tamanho da mensagem atingiu 10000 Bytes. Todos os ambientes após a agregação de duas placas Gigabit atingiram 1800 Mbps de *throughput*. Notou-se uma queda brusca no desempenho a partir de 400 Mbps de *throughput*, resultantes a contadores TCP. De acordo com Matteussi et al. há gargalos ocorridos devido aos parâmetros *Socket Buffers* (*rmem\_size*, *wmem\_size*) e *Transmit Queue Length* (*txqueuelen*) [Matteussi et al. 2014]. O ambiente KVM apresentou *overhead* de tráfego para tamanho de pacotes de 100000 Bytes. Segundo Trindade e Costa, esse *overhead* se dá ao fato de que o KVM possui sobrecargas em relação a troca de processos entre a máquina virtual e o *host* nativo [Trindade and Costa 2018]. Assim, ocasionando números elevados de ciclos de processamento.

Relacionado a latência, a Figura 4(b) demonstra um resultado similar para os três ambientes, onde tanto a virtualização baseada em *kernel* quando a virtualização baseada em contêiner obtiveram resultados praticamente iguais ao ambiente nativo.

## 5. Conclusões

A realização da pesquisa feita por este trabalho buscou avaliar o desempenho do protocolo *bonding* em virtualização com LXC e KVM em comparação ao ambiente nativo. Este tra-



**Figura 4. Bonding Throughput x Latência.**

balho propiciou a realização de testes com *benchmark* específico de análise do ambiente de rede, o que possibilitou a avaliação dos componentes utilizados. Em relação ao protocolo *bonding*, houveram divergência no *throughput* principalmente no ambiente KVM, resultantes aos contadores TCP e gargalos endereçados aos parâmetros *Socket Buffers* (*rmem\_size*, *wmem\_size*) e *Transmit Queue Length* (*txqueuelen*) [Matteussi et al. 2014]. Na avaliação de latência, nenhum *overhead* foi observado.

Através dos resultados conclui-se que o *bonding* é um método válido e adequado para melhorar a capacidade da rede, onde as placas de rede atingiram 90% (1800Mbps) de sua capacidade total. As contribuições deste trabalho foram principalmente em demonstrar o desempenho do protocolo *bonding* na agregação das interfaces físicas aumentando o *throughput* e diminuindo a latência da rede. Ao aplicar o protocolo *bonding* em virtualizações com LXC e KVM o desempenho da rede aproxima-se ao ambiente nativo.

Como trabalho futuro planeja-se explorar os outros modos de operação do protocolo *bonding* assim como realizar a comparação com o Multipath TCP. Também planeja-se utilizar outros ambientes de virtualização como VMware ou Xen.

## Agradecimentos

Todos os testes e resultados presentes nesta pesquisa foram realizados na infraestrutura do LARCC<sup>1</sup>. Os autores agradecem ao apoio do projeto HiPerfCloud<sup>2</sup> bem como da Sociedade Educacional Três de Maio (SETREM)<sup>3</sup>.

## Referências

- [Aust et al. 2006] Aust, S., Kim, J.-O., Davis, P., Yamaguchi, A., and Obana, S. (2006). Evaluation of linux bonding features. In *Communication Technology, 2006. ICCT06. International Conference on*, pages 1–6. IEEE.
- [Bondan et al. 2012] Bondan, L., Gobbi, R. C., and Fochi, V. M. (2012). O protocolo de agregação de enlaces lacp. Undergraduate Thesis.

<sup>1</sup><https://larcc.setrem.com.br/>

<sup>2</sup><https://hiperfcloud.setrem.com.br/>

<sup>3</sup><https://www.setrem.com.br/>

- [Brassil 2005] Brassil, J. (2005). An empirical study of bonded heterogeneous wan access links. In *Local and Metropolitan Area Networks, 2005. LANMAN 2005. The 14th IEEE Workshop on*, pages 6–pp. IEEE.
- [Chirammal et al. 2016] Chirammal, H. D., Mukhedkar, P., and Vettathu, A. (2016). *Mastering KVM Virtualization*. Packt Publishing Ltd.
- [CISCO 2013] CISCO (2013). Global internet traffic projected to quadruple by 2015 <<https://newsroom.cisco.com/press-release-content?type=webcontent&articleid=324003>>. Último acesso Setembro de 2018.
- [Coutinho et al. 2013] Coutinho, E., Sousa, F. R., Gomes, D. G., and Souza, J. (2013). Elasticidade em computação na nuvem: Uma abordagem sistemática. *XXXI Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC 2013)-Minicursos*, pages 1–44.
- [Hat 2018] Hat, R. (2018). Libvirt virtualization api <<https://libvirt.org/index.html/>>. Último acesso Junho de 2018.
- [Imaizumi et al. 2009] Imaizumi, H., Nagata, T., Kunito, G., Yamazaki, K., and Morikawa, H. (2009). Power saving mechanism based on simple moving average for 802.3 ad link aggregation. In *GLOBECOM Workshops, 2009 IEEE*, pages 1–6. IEEE.
- [INTEL 2013] INTEL (2013). Guia de planejamento - saiba mais sobre big data <<https://www.intel.com.br/content/dam/www/public/lar/br/pt/documents/articles/90318386-1-por.pdf>>. Último acesso Setembro de 2018.
- [Matteussi et al. 2014] Matteussi, K. J., Xavier, B. G., Sparremberger, A. S., Rose, C. d., and Ferreto, T. C. (2014). Avaliação de desempenho sobre agregações de links em modo round robin. In *12th Escola Regional de Redes de Computadores (ERRC)*, pages 1–6. Sociedade Brasileira de Computação.
- [Mayer-Schönberger and Cukier 2013] Mayer-Schönberger, V. and Cukier, K. (2013). *Big Data: A Revolution that Will Transform how We Live, Work, and Think*. An Eamon Dolan book. Houghton Mifflin Harcourt.
- [Rista et al. 2017] Rista, C., Griebler, D., Maron, C. A., and Fernandes, L. G. (2017). Improving the network performance of a container-based cloud environment for hadoop systems. In *High Performance Computing & Simulation (HPCS), 2017 International Conference on*, pages 619–626. IEEE.
- [Sarabando 2008] Sarabando, N. G. (2008). Validation of "triple-play" services in the access node. Master's thesis, Universidade de Aveiro.
- [Trindade and Costa 2018] Trindade, L. V. P. and Costa, L. H. M. (2018). Análise do desempenho da virtualização leve para ambientes com edge computing baseada em nfv. In *Simpósio Brasileiro de Redes de Computadores (SBRC)*, volume 36.
- [Vogel et al. 2017] Vogel, A., Griebler, D., Schepke, C., and Fernandes, L. G. (2017). An intra-cloud networking performance evaluation on cloudstack environment. In *Parallel, Distributed and Network-based Processing (PDP), 2017 25th Euromicro International Conference on*, pages 468–472. IEEE.