

# XenFlow: Um Sistema de Processamento de Fluxos Robusto e Eficiente para Migração em Redes Virtuais\*

Diogo Menezes Ferrazani Mattos, Natalia Castro Fernandes,  
Otto Carlos Muniz Bandeira Duarte

<sup>1</sup>Grupo de Teleinformática e Automação  
Universidade Federal do Rio de Janeiro (UFRJ)  
Rio de Janeiro – RJ – Brasil

{menezes, natalia, otto}@gta.ufrj.br

**Abstract.** *Virtual topology migration provides high flexibility to virtual networks, but it is a great challenge. This paper proposes a flow processing system for virtual networks, which provides a simplified virtual router and virtual link migration, without packet losses. The system uses a flow based forwarding mechanism, that makes the migration primitive flexible, simple and robust. The system enables the virtual router migration, and also provides a simple mapping of one virtual link over one or more physical links, without creating tunnels. The key idea is based on a hybrid virtualization technique that combines the Xen and OpenFlow virtualization techniques, providing a flexible data plane and a distributed network control. We developed a prototype and results show that the proposed system presents a higher performance, when it is compared to the conventional Xen virtual machine migration, and it also provides virtual topology migration inside wider spatial scenarios.*

**Resumo.** *A migração de topologias virtuais oferece uma flexibilidade significativa para as redes virtuais, mas ainda representa um grande desafio. Este artigo propõe um sistema de processamento de fluxos para redes virtuais que permite uma migração sem perda de pacotes tanto de roteadores quanto de enlaces virtuais. O sistema utiliza um mecanismo de encaminhamento de pacotes baseado em fluxos, o que torna a funcionalidade de migração flexível, simples e robusta. O sistema permite a migração de roteadores virtuais, assim como o mapeamento simplificado de um enlace lógico sobre um ou mais enlaces físicos sem a utilização de túneis. A ideia chave baseia-se em uma técnica de virtualização híbrida que combina as técnicas de virtualização das plataformas Xen e OpenFlow, disponibilizando, ao mesmo tempo, um plano de dados flexível e um controle distribuído da rede. Foi desenvolvido um protótipo e os resultados obtidos mostram que o novo sistema apresenta um desempenho superior, quando comparado com a migração convencional de uma máquina Xen, além de permitir a migração de topologias virtuais em cenários espacialmente mais amplos.*

## 1. Introdução

A implantação de novos protocolos e serviços no núcleo da Internet sofre a rejeição de grande parte dos provedores de serviços devido ao grande risco que essas

---

\*Este trabalho foi realizado com recursos da FINEP, FUNTTEL, CNPq, CAPES e FAPERJ.

mudanças representam para o bom funcionamento da rede. Uma das propostas para conciliar o desenvolvimento de inovações e tráfego de produção é a virtualização de redes [Feamster et al. 2007, Ratnasamy et al. 2005]. A técnica de virtualização de redes provê diferentes redes, isoladas entre si, em um mesmo substrato físico. Duas importantes técnicas de virtualização são oferecidas pelas plataformas Xen [Egi et al. 2008] e OpenFlow [McKeown et al. 2008]. O Xen é uma plataforma de virtualização de recursos de computadores pessoais com muito sucesso na consolidação de servidores. O OpenFlow é uma plataforma de virtualização de redes baseado na definição e comutação de fluxos. Um comutador OpenFlow realiza a comutação de pacotes baseado em regras definidas por um nó controlador centralizado. Uma das principais propostas para controlador OpenFlow é o sistema Nox [Gude et al. 2008] que implementa uma interface entre aplicações de controle e a rede OpenFlow.

A virtualização de redes introduz uma nova primitiva de gerenciamento, a migração de redes virtuais [Wang et al. 2008]. A primitiva de migração pode ser usada em diversos contextos, desde a realocação da topologia lógica para a manutenção de nós da rede, até no provimento de uma rede verde. Operações de manutenção de nós de rede frequentemente requerem o seu desligamento, o que gera a quebra de conexão e, no caso da manutenção de roteadores, pode gerar a perda das adjacências e a necessidade de nova convergência do algoritmo de roteamento. O uso da migração em redes verdes se dá através da reorganização de nós virtuais, considerando a demanda de uso e o consumo energético [Bolla et al. 2009]. A migração pode, ainda, ser usada para evitar danos no caso de ataques de negação de serviço (DoS – *Denial of Service*). Nessa situação, as redes virtuais, que compartilham o mesmo substrato com a rede sob ataque, são migradas para outros nós físicos fora da região do ataque. Contudo, a migração de topologias virtuais apresenta grandes desafios, como o remanejamento dos enlaces virtuais e a redução dos danos causados pelo período de suspensão dos serviços prestados pelos nós migrados, durante a migração.

Existem propostas [Wang et al. 2007, Pisa et al. 2010] de realizar a migração da topologia lógica de forma transparente para as extremidades que utilizam a rede, sem que haja perda de pacotes ou quebra de conexão. No entanto, os cenários, em que essas propostas são válidas, são limitados. Em [Wang et al. 2007], supõe-se a existência de um mecanismo para a migração de enlaces que é externo ao mecanismo de migração de topologia. Supõe-se ainda que um roteador virtual só pode ser migrado de uma máquina física para outra que tenha as mesmas adjacências ou que seja possível criar túneis entre elas. A migração de [Pisa et al. 2010] supõe que exista uma rede local de controle sobre os elementos migrados e supõe o uso de túneis para a migração de enlaces ou que os roteadores físicos tenham conjuntos de adjacências compatíveis. Já a migração de fluxos na plataforma OpenFlow é fácil. Pisa *et al* apresenta um algoritmo que se baseia na redefinição de um caminho para fluxos na rede OpenFlow [Pisa et al. 2010]. Tal proposta apresenta perda zero de pacotes e baixa sobrecarga na rede. Entretanto, esta proposta de migração para redes OpenFlow não é aplicável à virtualização de roteadores ou a sistemas que executem o processamento de fluxos. A proposta restringe-se a redes comutadas.

Este artigo propõe o XenFlow, um sistema de processamento de fluxos que permite migração de redes virtuais, realizando tanto migração de nós virtuais, quanto a migração de enlaces. O roteador virtual é dividido em dois planos, o plano de controle

e o plano de dados. O plano de controle é responsável pela atualização da tabela de roteamento, a partir do protocolo de roteamento. O plano de dados é responsável pelo encaminhamento dos pacotes, de acordo com as políticas de roteamento definidas com base nas rotas calculadas pelo plano de controle. A principal contribuição deste trabalho consiste na adoção de uma abordagem híbrida de virtualização de redes, que combina as plataformas de virtualização Xen e OpenFlow. O plano de controle é implementado em uma máquina virtual Xen, na qual executam os protocolos de roteamento. As máquinas virtuais Xen, comunicam-se com a rede física através de um comutador OpenFlow, que realiza o encaminhamento dos pacotes e implementa o plano de dados. A principal vantagem da presente técnica de virtualização de redes é que, ao realizar a migração de topologias virtuais, pode-se, agora, mapear um enlace lógico sobre um ou mais enlaces físicos e a migração ocorre sem perdas de pacotes. Essa facilidade permite que o alcance da migração de um roteador virtual seja aumentado em relação a outras propostas na literatura, pois não há a restrição de que o nó físico de destino tenha os mesmos vizinhos que o nó físico de origem. Outra vantagem do sistema proposto é o controle distribuído da rede.

A principal proposta de controle distribuído para a rede OpenFlow é o FlowVisor [Sherwood et al. 2009]. No entanto, o FlowVisor realiza o particionamento da rede segundo políticas e permite que um controlador seja definido para cada partição. Dessa forma, o controle de uma rede virtual é centralizado em um único controlador. Paralelamente, na rede XenFlow, o particionamento da rede física é definido por um conjunto de máquinas virtuais e o controle da rede virtual é definido pelos protocolos que executam nas máquinas virtuais, por exemplo, os protocolos de roteamento.

Um protótipo do sistema XenFlow foi construído para a validação da proposta. Os resultados experimentais mostram que o sistema proposto é robusto à migração, no sentido em que, durante a migração não há perda de pacotes nem a interrupção do serviço de encaminhamento de pacotes. O sistema também é eficiente, na medida em que permite a execução da migração de roteador e enlace virtuais sem que haja a perda da conexão ou atraso no encaminhamento dos pacotes. Quando comparado à migração de máquina virtual nativa do Xen, o sistema XenFlow apresentou perda zero de pacotes, enquanto a migração nativa perdeu uma quantidade significativa de pacotes, além de apresentar um período maior de interrupção da atualização do plano de controle.

O restante deste artigo está organizado da seguinte forma. A Seção ?? discute os trabalhos relacionados. A Seção 2 apresenta as tecnologias usadas na definição da plataforma híbrida de virtualização de redes. A definição do sistema proposto, sua arquitetura e principais componentes são apresentados na Seção 3. A Seção 4 apresenta a análise dos resultados experimentais. As conclusões e trabalhos futuros são apresentados na Seção 5

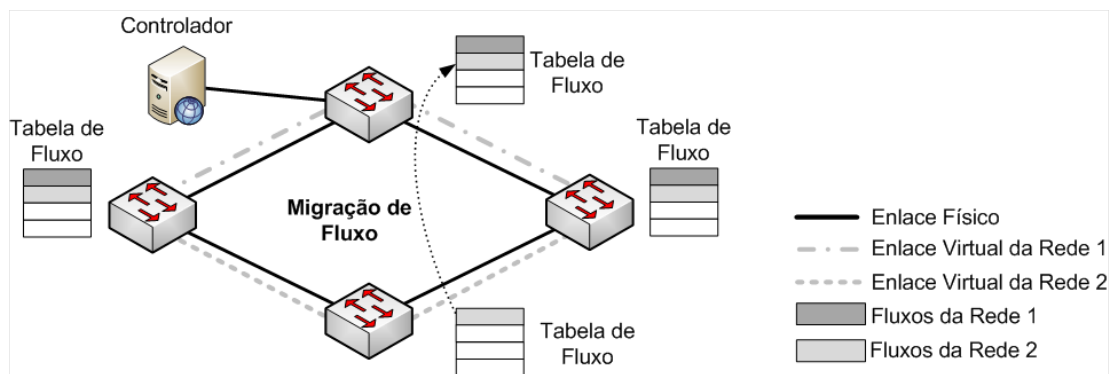
## **2. Plataformas de Virtualização Xen e OpenFlow**

XenFlow é um sistema de processamento de fluxos, baseado na plataforma Xen e na técnica de comutação da plataforma OpenFlow, que provê uma eficiente funcionalidade de migração de topologias virtuais.

O OpenFlow é uma tecnologia de comutação que permite a programação do encaminhamento dos pacotes através de ações associadas a fluxos. Um fluxo é definido como um conjunto de doze características da comunicação entre dois nós na rede. As características de um fluxo OpenFlow não se restringem às características dos protocolos

de camada de enlace, mas agregam informação da camada de rede e da camada de transporte [McKeown et al. 2008]. A tabela de encaminhamento de um comutador OpenFlow é a Tabela de Fluxo, a qual relaciona um fluxo com uma determinada porta de saída do comutador através de ações. A função de decidir as ações a serem tomadas para cada fluxo é de um controlador centralizado, que processa o primeiro pacote de um fluxo e define as ações. O Nox [Gude et al. 2008] é um controlador OpenFlow que age como uma interface entre aplicações de controle e a rede OpenFlow. Quando um pacote chega ao comutador OpenFlow, o comutador verifica se o pacote se adéqua a algum fluxo já existente. Em caso positivo, as ações definidas para aquele fluxo são aplicadas ao pacote. Em caso negativo, o pacote é enviado ao controlador, que extrai as características do fluxo, a partir do pacote, e cria um novo fluxo introduzindo-o na tabela de fluxos do comutador OpenFlow.

No OpenFlow a migração de redes virtuais corresponde à migração de fluxos e, portanto, é bastante simples, pois basta reconfigurar as tabelas de fluxo, como mostrado na Figura 1. Isso é possível, pois o controlador centralizado tem acesso e controla a configuração das tabelas de todos os comutadores na rede virtual. Assim, o algoritmo de migração consiste em adicionar uma nova entrada para o fluxo na tabela de fluxos de cada um dos comutadores no novo caminho, com exceção do primeiro comutador de origem, o qual é comum ao caminho antigo e ao novo. Após essa primeira etapa, o controlador modifica o fluxo definido no primeiro comutador, de modo que a antiga porta de saída seja modificada para uma nova porta, que redireciona o fluxo ao restante do caminho previamente definido.

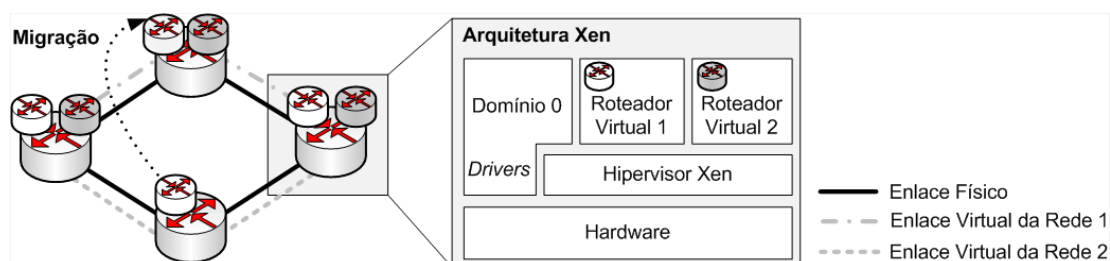


**Figura 1. Virtualização de redes usando OpenFlow. Cada rede virtual é representada por um conjunto de fluxos. A migração da rede virtual é a redefinição dos fluxos referentes àquela rede em outro comutador.**

O Xen é uma plataforma de virtualização de computadores pessoais bastante empregada na consolidação de servidores e cuja arquitetura é baseada em uma camada de virtualização, denominada Monitor de Máquina Virtual (VMM) ou hipervisor. Os ambientes virtuais Xen são chamados de máquinas virtuais, ou domínios, e apresentam recursos próprios, como CPU, memória, acesso a disco e a rede. Cada ambiente virtual está isolado dos demais. Há, ainda, um ambiente virtual privilegiado, denominado Domínio 0, que detém o acesso aos dispositivos físicos, provê acesso às operações de Entrada/Saída dos demais domínios e executa operações de gerência no hipervisor.

A Figura 2 mostra uma rede virtualizada com base no Xen. A rede é composta por

máquinas virtuais Xen executando a função de roteador [Fernandes et al. 2010]. Nesse cenário, migrar um roteador virtual equivale a migrar uma máquina virtual. Como um roteador executa um serviço em tempo real, a migração de um roteador virtual demanda que o tempo de interrupção do serviço de encaminhamento de pacotes seja o mínimo possível. Sendo assim, existem duas formas distintas de migrar uma rede virtual no Xen.



**Figura 2. Virtualização de redes usando Xen. Cada máquina virtual tem um sistema operacional próprio e executa aplicações de roteamento isoladas das demais. Os enlaces físicos são compartilhados pelas redes virtuais. A migração de uma rede virtual equivale é aplicada através da migração de um roteador virtual.**

A migração nativa do Xen [Clark et al. 2005] baseia-se na migração ao vivo de máquinas virtuais, que consiste em copiar a memória da máquina virtual de uma máquina física de origem para a de destino. Como as páginas de memória da máquina virtual, na origem, são alteradas durante o procedimento de cópia, utiliza-se um mecanismo de cópias iterativas de páginas de memória, no qual as páginas alteradas são marcadas e copiadas na iteração seguinte. Isso se repete até que o número de páginas de memória alteradas desde a última rodada seja suficientemente pequeno. Nesse momento, a execução da máquina virtual é suspensa na máquina física de origem, as últimas páginas de memória alteradas são copiadas para o destino, e a máquina virtual é, então, restaurada na máquina física de destino. Uma desvantagem dessa proposta para a migração de roteadores virtuais é a perda de pacotes durante o tempo em que a máquina virtual fica indisponível, entre a suspensão e a restauração. Este mecanismo está limitado a uma rede local, pois essa abordagem supõe a existência de um disco rígido compartilhado pela rede e a migração de enlaces é realizada através do envio de pacotes de *ARP Reply*.

Para evitar a perda de pacotes na migração, propôs-se a migração com separação de planos. A separação de planos consiste em dividir a tarefa de roteamento em dois planos distintos, um de controle e outro de dados. O processo de controle, definido no plano de controle, é responsável pela execução do protocolo de roteamento e pela atualização da tabela de rotas. Já o processo de encaminhamento é responsável por rotear os pacotes nas interfaces de saída corretas, de acordo com as rotas definidas no plano de controle. Pisa *et al.* propõem a mudança do plano de dados de todos os roteadores virtuais para o Domínio 0 dos roteadores físicos [Pisa et al. 2010]. Dessa forma, a migração do plano de dados ocorre sem afetar o encaminhamento dos pacotes. No entanto, a solução é restritiva, no sentido de que um roteador virtual só pode ser migrado para outro roteador físico que apresente os mesmos vizinhos que o de origem. Essa solução limita o alcance da migração a um salto a partir do roteador de origem.

### 3. Sistema XenFlow Proposto

O sistema proposto combina as vantagens da virtualização de redes com controle distribuído da plataforma Xen, com o processamento por fluxo provido pela plataforma OpenFlow. A arquitetura de um elemento de rede XenFlow é mostrada na Figura 3. A base da arquitetura na máquina física é a plataforma de virtualização Xen. Dessa forma, a virtualização de redes é alcançada através da instanciação de máquinas virtuais, que executam funções referentes a um elemento de rede virtual, e pelo conjunto de fluxos definidos por essas máquinas. Devido ao processo de encaminhamento de fluxo ser baseado no OpenFlow, cada elemento de rede virtual pode ser um comutador (nível 2), roteador (nível 3) ou *middle box* (nível maior que 3). A função que cada elemento de rede executa depende da pilha de protocolos que ele implementa e do seu conjunto de aplicação. A comunicação entre dois elementos de rede virtuais, de uma mesma máquina física na qual estão hospedados, e de um elemento de rede com a rede externa à máquina física, é realizada pelo Domínio 0. O sistema XenFlow, para prover maior flexibilidade na definição dos caminhos dos fluxos no Domínio 0, comuta os fluxos por uma matriz de comutação programável que implementa o protocolo OpenFlow. Assim, a funcionalidade de migração de topologia virtual no sistema XenFlow é simples, como na plataforma OpenFlow, e torna possível reorganizar a rede lógica sem que sejam necessárias mudanças na topologia física.

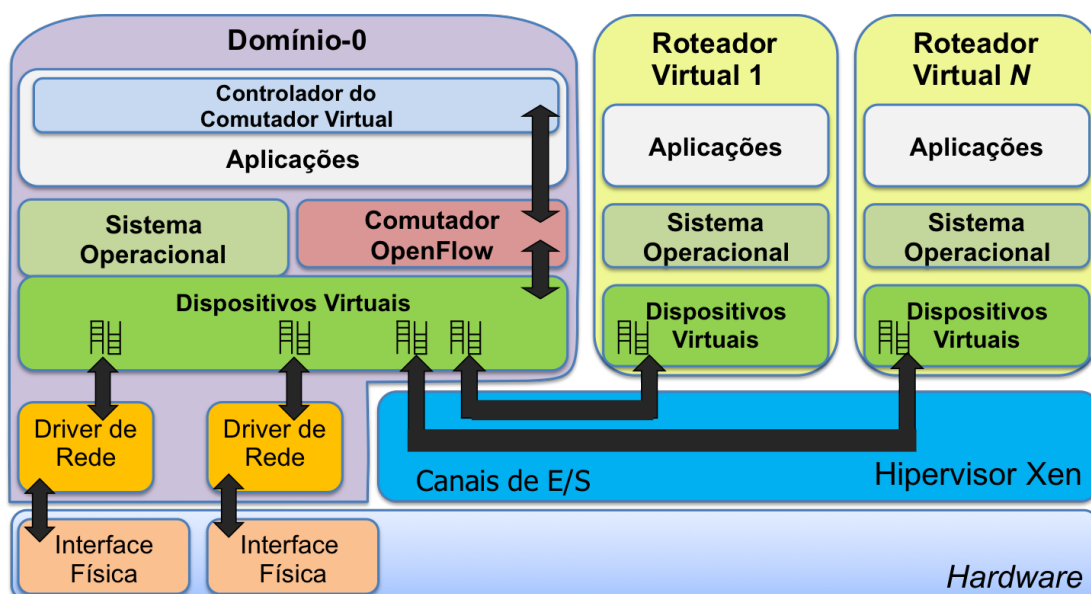
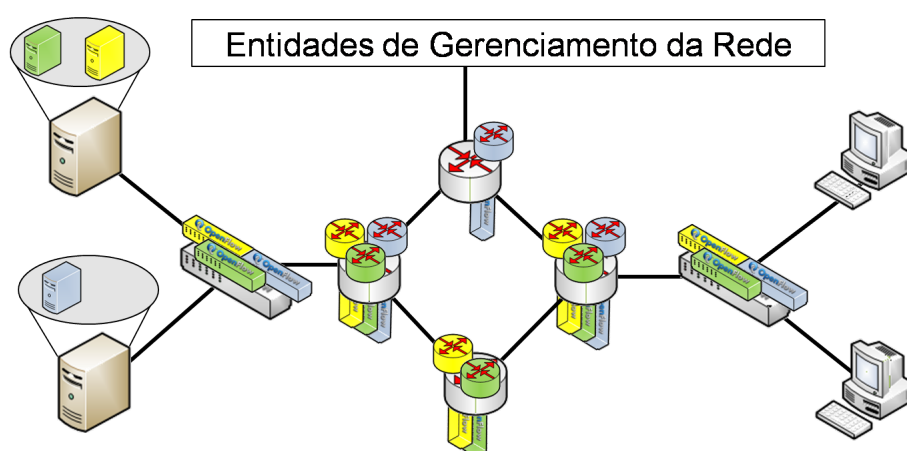


Figura 3. Arquitetura de um elemento de rede do sistema XenFlow.

No sistema XenFlow, assim como na plataforma Xen, os *drivers* dos dispositivos físicos ficam do Domínio 0 e, portanto, toda comunicação entre máquinas virtuais e dispositivos físicos deve passar pelo Domínio 0. Assim, o Domínio 0 realiza a multiplexação dos pacotes que estão saindo dos elementos de rede virtuais para os dispositivos físicos de rede e a demultiplexação dos pacotes que estão chegando aos dispositivos físicos de rede e vão para os elementos de rede virtuais [Fernandes et al. 2010]. No caso do sistema XenFlow, o processo de multiplexação e demultiplexação é realizado por um comutador OpenFlow. Uma das vantagens de realizar a multiplexação dos pacotes no Domínio 0

em um comutador OpenFlow é que o encaminhamento dos pacotes pode ser programado de acordo com as regras definidas no controlador do comutador OpenFlow, que, por sua vez, é uma aplicação que executa no Domínio 0 e define as regras de encaminhamento dos pacotes. Outra vantagem de se estabelecer o mapeamento de interfaces físicas com interfaces virtuais a partir de um comutador OpenFlow é que se pode estabelecer regras no controlador para garantir uma banda mínima para cada fluxo ou conjunto de fluxo [McKeown et al. 2008]. O comutador OpenFlow agrega as interfaces virtuais, que realizam a comunicação ponto-a-ponto entre domínios virtuais e Domínio 0, e as interfaces físicas. Sendo assim, os fluxos de entrada e saída da máquina física também podem ser programados segundo as políticas do controlador. O controlador também implementa uma interface de gerenciamento dinâmico da rede. Através dessa interface, pode-se inserir novas regras na Tabela de Regras do controlador ou atualizar outras regras.



**Figura 4. Rede XenFlow. A rede é composta por nós que implementam um plano de dados OpenFlow e um plano de controle Xen.**

Embora o OpenFlow apresente um controle centralizado, o controle da rede XenFlow é distribuído em fatias de roteadores físicos que participam da rede virtual. Esta, por sua vez, é representada por um conjunto de roteadores virtuais que implementam uma pilha de protocolos e pelo conjunto de fluxos instanciados para aquela rede. A Figura 4 mostra o sistema XenFlow com três redes virtuais instanciadas. Nessa rede, há nós da rede física que são comutadores e outros que são roteadores. Os comutadores físicos executam o protocolo OpenFlow, o que os permite implementarem comutadores virtuais. Os roteadores físicos implementam o sistema, como descrito na Figura 3. Na Figura 4, há ainda uma entidade de gerenciamento. Essa entidade tem consciência da topologia física da rede, que pode ser obtida através da topologia calculada por protocolos de roteamento de estado de enlace, como o OSPF (*Open Shortest Path First*), que calculam as melhores rotas com base na topologia da rede armazenada localmente. É a partir da entidade de gerência que as migrações são iniciadas. Quando uma migração ocorre, a topologia física da rede é consultada para que o sistema mapear os enlaces lógicos sobre os enlaces físicos.

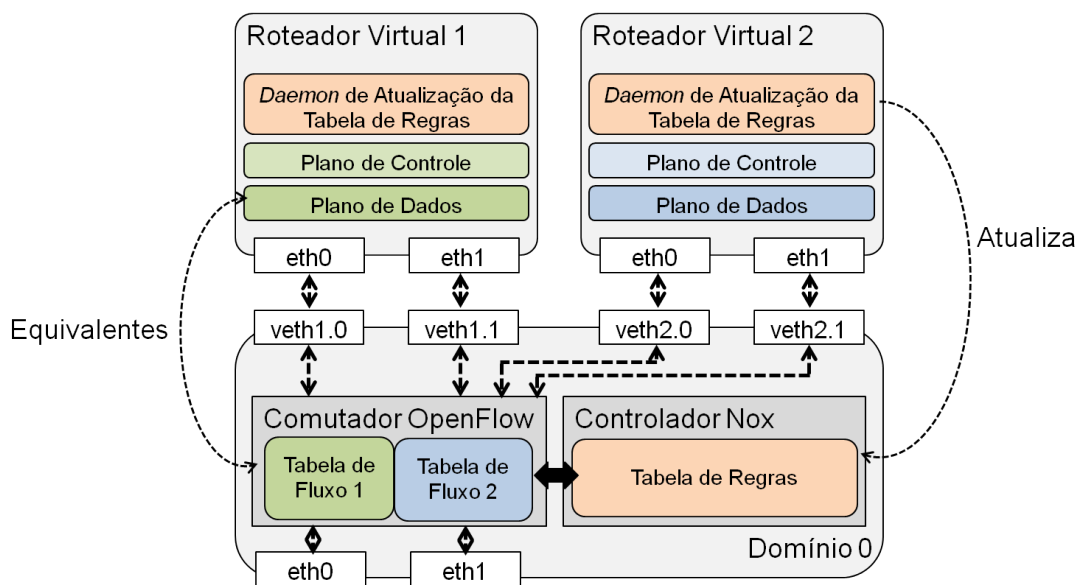
### 3.1. Separação de Planos e Tradução de Rotas em Fluxos

A migração sem perda de pacotes é fundamental na migração de roteadores e este objetivo pode ser alcançado com a técnica de separação de planos. As informações



de rotas estão no plano de controle e as regras de encaminhamento, no plano de dados. Com a separação de planos há necessidade de se ter uma cópia do plano de dados, de cada elemento de rede virtual, no Domínio 0. No XenFlow a cópia do plano de dados no Domínio 0 é composta pelo comutador OpenFlow e pelo Controlador Nox, como ilustrado na Figura 5. O plano de dados de cada elemento de rede virtual deve ser equivalente ao conjunto formado pelas tabelas de fluxos do OpenFlow e pela tabela de regras do Nox. Logo, o plano de controle recebe as informações de rotas e o módulo de *daemon* de atualização de tabela de regras as envia para o controlador Nox. As rotas são então traduzidas em fluxos pelo módulo tabela de regras, executado no Nox, que, por sua vez, atualiza a tabela de fluxos do comutador OpenFlow.

Os pacotes de dados são encaminhados pelo XenFlow da seguinte forma. Um pacote, ao chegar ao comutador OpenFlow do roteador físico, tem dois tratamentos possíveis, ser encaminhado diretamente, caso coincida com um fluxo existente na tabela de fluxos, ou, caso contrário, ser encaminhado para o controlador, para que este defina o seu caminho. O controlador Nox, então, extrai do pacote os doze campos necessários para a definição do fluxo no OpenFlow, consulta a tabela de regras para definir qual é o destino do pacote e insere um fluxo na tabela de fluxos do comutador. É importante ressaltar que o pacote chega com o endereço MAC de destino do roteador virtual e este endereço deve ser trocado pelo endereço MAC do próximo salto, que é obtido da tabela de regras. Este procedimento é que vai viabilizar o mapeamento de um enlace virtual em um ou mais enlaces físicos.



**Figura 5. Roteamento no sistema XenFlow com separação de planos. Os pacotes são encaminhados diretamente pelo Domínio 0, pois esse tem uma cópia traduzida do plano de dados dos roteadores virtuais.**

A tabela de regras é atualizada pelo módulo de *daemon*, o qual repassa as mudanças de rota ocorridas no plano de controle para o módulo de tabela de regras, no plano de dados. Assim, o controlador Nox avalia quais fluxos são afetados pela atualização e os adequa às novas regras de roteamento.



### 3.2. Migração de Topologias Virtuais no XenFlow

Em uma rede com o sistema XenFlow, um enlace virtual pode ser mapeado em um ou mais enlaces físicos. O encaminhamento é feito por uma tabela de fluxos programada dinamicamente pelo Controlador Nox. Assim, a topologia lógica fica desassociada da topologia física. Sendo assim, a migração de nós virtuais em uma rede XenFlow, mostrada na Figura 6, se dá em três etapas: migração do plano de controle, migração do plano de dados e migração de enlaces. O plano de controle é migrado entre dois nós físicos da rede, através do mecanismo de migração ao vivo de máquinas virtuais convencional do Xen [Clark et al. 2005]. Em seguida, a migração do plano de dados é realizada da seguinte forma: os fluxos referentes ao roteador virtual migrado são selecionados e enviados para o roteador físico de destino; no destino, a definição dos fluxos é mapeada para a atual configuração do roteador físico e do roteador virtual. Assim, mantém-se a correspondência das portas de entrada e de saída do fluxo em relação ao comutador virtual do Domínio 0, da origem, no destino. Em seguida, os fluxos traduzidos são adicionados à tabela de fluxos do comutador OpenFlow do Domínio 0 de destino. Depois da migração do plano de dados e do plano de controle, ocorre a migração dos enlaces. A migração de enlaces ocorre nos comutadores OpenFlow dos Domínios 0 e nos outros comutadores da rede. A migração de enlaces ocorre de forma a criar um caminho comutado entre os vizinhos a um salto lógico do roteador virtual até o roteador físico de destino da migração. Para tanto, são criados fluxos nos roteadores físicos que estão no caminho entre o roteador físico de destino e os roteadores físicos que hospedam os roteadores virtuais, que estão a um salto lógico do roteador virtual migrado. No entanto, adicionar somente os fluxos já existentes às tabelas de fluxo dos nós físicos nesse caminho não é suficiente. É necessário que haja um mecanismo de criação automática de novos fluxos sob demanda. Esse mecanismo se dá através da introdução de novas regras nas Tabelas de Regras dos controladores dos nós no caminho.

## 4. Resultados Experimentais

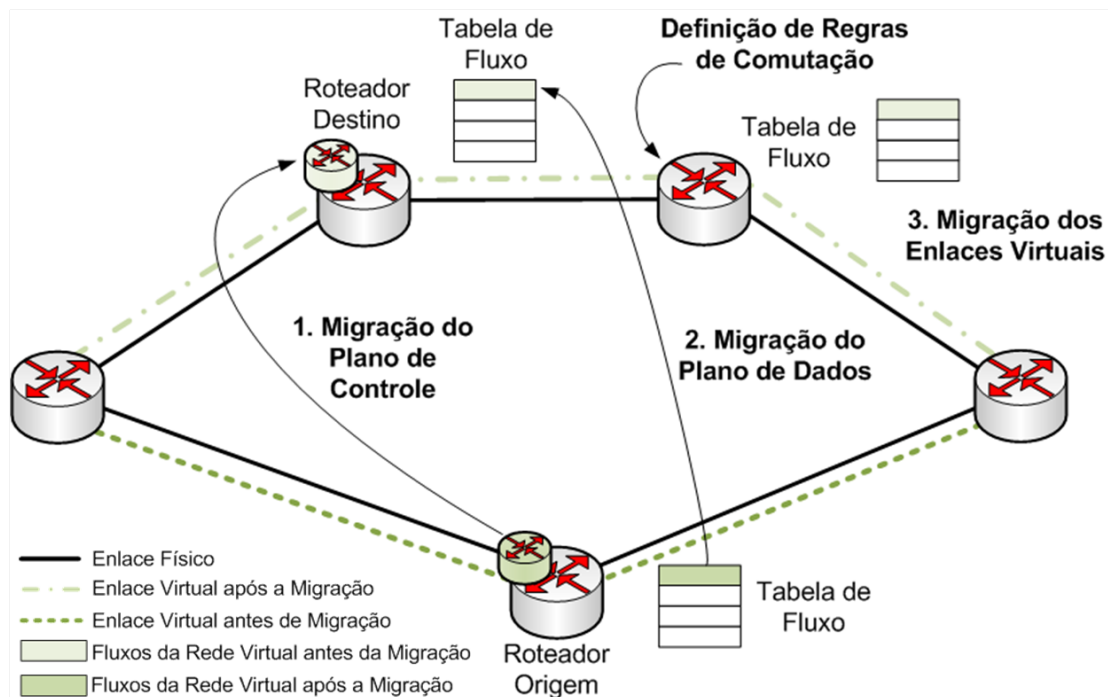
Foi desenvolvido um protótipo para realizar a prova de conceito da migração de roteadores virtuais sem perda de pacotes. O protótipo foi implementado em Python e utiliza a técnica de separação de planos, oferecendo uma interface de migração de roteadores virtuais e uma interface de migração de enlaces. O plano de dados foi implementado como uma tabela de fluxos e uma aplicação do controlador Nox, do comutador OpenFlow. A aplicação define os fluxos diretamente entre interface de entrada e de saída, na tabela de fluxo do comutador OpenFlow local, alterando o endereço MAC de destino dos pacotes encaminhados para o endereço MAC do próximo salto. Para avaliar o desempenho, foi adotada a ferramenta *Iperf*<sup>1</sup> como gerador de pacotes e a ferramenta *tcpdump*<sup>2</sup> para medir quantidade de pacotes gerados, recebidos e perdidos. A perda de pacotes foi medida a partir da comparação das informações coletadas pelo *tcpdump* nas interfaces de rede responsáveis geração e de recepção dos pacotes.

O cenário de testes foi composto por quatro máquinas. Duas máquinas executam a função de encaminhamento de pacotes e nelas foi instalado o protótipo. Essas máquinas são equipadas com processador Intel Core 2 Quad e três interfaces de rede Ethernet de

---

<sup>1</sup><http://iperf.sourceforge.net/>

<sup>2</sup><http://www.tcpdump.org/>

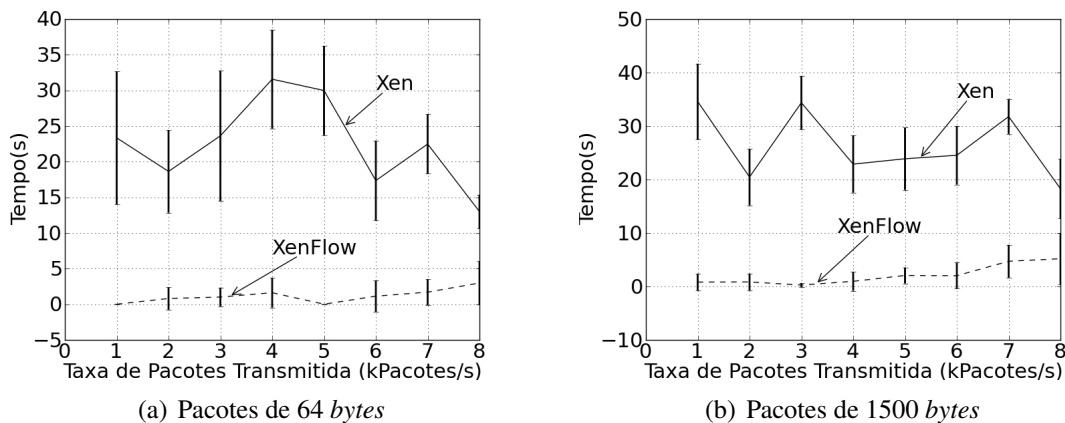


**Figura 6. As três etapas da migração de topologia virtual em uma rede XenFlow.**

1Gb/s, executando o hipervisor Xen 4.0-amd64. Em uma dessas máquinas, foi instanciada uma máquina virtual, com um CPU virtual, 128 MB de memória, duas interfaces de rede e executando o sistema operacional Debian 2.6-32-5. A máquina virtual realiza a função de roteador. Os testes usam ainda duas outras máquinas, equipadas com processador Intel Core 2 Duo, que geram ou recebem pacotes, cada uma com uma placa de rede Ethernet de 1Gb/s, ligadas a uma rede de controle, e duas placas de redes Ethernet de 100Mb/s, para se comunicarem simultaneamente com os dois roteadores físicos. Os testes foram realizados com o roteador virtual encaminhando pacotes UDP de 64 e 1500 bytes, que são, respectivamente, o tamanho mínimo do conteúdo de um quadro Ethernet e o tamanho mais comum de MTU (*Maximum Transmission Unit*).

Durante a migração o plano de controle sofre um período de suspensão e este período não pode ser muito grande para que mensagens de controle de roteamento sejam perdidas. Assim, o primeiro teste objetiva medir o tempo de suspensão do plano de controle durante a migração. O teste foi realizado enviando-se pacotes de controle que passam pelo roteador virtual durante a migração. Durante o período de suspensão do plano de controle, para a cópia das últimas páginas de memória, verifica-se uma interrupção no encaminhamento desses pacotes de controle. O tempo de perda da conexão com o plano de controle é dado pela diferença do tempo do pacote de controle recebido imediatamente antes da migração com o tempo do pacote de controle recebido imediatamente após a migração. A Figura 7 mostra o tempo de suspensão do plano de controle para o sistema XenFlow e para a migração nativa do Xen, em função da taxa de pacotes enviada. Os resultados mostram que o tempo de suspensão do roteador virtual é próximo de zero no sistema XenFlow, independente do tamanho dos pacotes. Já na migração nativa do Xen, o tempo de suspensão do roteador virtual variou entre 12 e 35 segundos. Essa diferença se dá por dois motivos principais. O primeiro é que, na migração usando o XenFlow, não

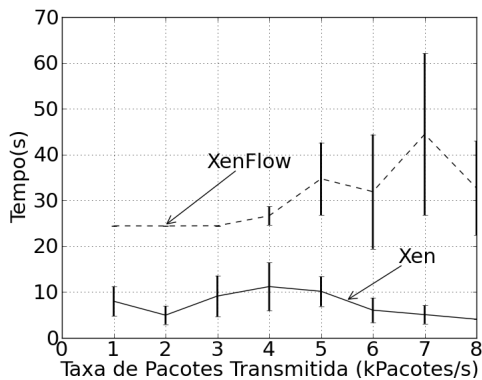
há escrita de memória na máquina virtual, pois os pacotes são encaminhados diretamente pelo Domínio 0, ao passo que na migração do Xen, todos pacotes são encaminhados pela máquina virtual, gerando escritas e leituras de memória enquanto a máquina virtual é migrada. O maior uso da memória acarreta em mais páginas sujas e, portanto, no momento da cópia das últimas páginas, maior tempo de suspensão da máquina virtual. O segundo motivo é que, na migração do XenFlow, há uma etapa de migração de enlaces, realocando os fluxos nas máquinas geradora e receptora para as interfaces corretas. Na migração nativa do Xen, tal tarefa é realizada através do envio de pacotes de *ARP Reply*, para indicar em qual nova interface uma máquina virtual migrada está disponível. No entanto, o funcionamento do mecanismo de *ARP Reply* está condicionado ao vencimento da entrada ARP nas tabelas do sistema. Isso pode adicionar um atraso na atualização da interface que a máquina deve utilizar para se comunicar com a máquina migrada.



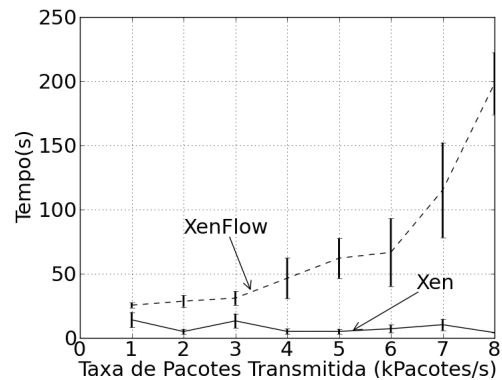
**Figura 7. Tempo de suspensão do plano de controle durante a migração.**

Não se pode iniciar uma nova migração enquanto uma migração anterior não estiver completada. Logo, o tempo total de uma migração estabelece o período mínimo de tempo entre duas migrações consecutivas de um mesmo elemento virtual. O segundo teste realizado avalia o tempo total da migração. O tempo total da migração considera o tempo de execução de todas as operações referentes ao processo de migração. A Figura 8 apresenta os resultados do tempo total de migração em função da taxa de pacotes enviada. Os resultados demonstram que a migração no sistema XenFlow apresenta um acréscimo no tempo total que pode chegar até quatro vezes mais do que o tempo de migração de uma máquina virtual no Xen nativo porque envolve mais etapas, sendo uma das etapas é a própria migração nativa do Xen. Parte do adicional do tempo da migração deve-se à migração dos fluxos, para a reconstrução do plano de dados no roteador físico de destino, e também à migração de enlaces, que é responsável por configurar a nova topologia da rede virtual sobre a rede física. A Figura 8(b) mostra que para pacotes de 1500 bytes há um aumento no tempo total da migração no XenFlow, a medida em que a taxa de pacotes aumenta. O aumento é devido ao fato de que, para pacotes de 1500 bytes, há a saturação dos enlaces de 100 Mb/s a taxas de aproximadamente 8.000 pacotes.

Em migração de roteadores o fato de não haver perdas de pacotes é fundamental.



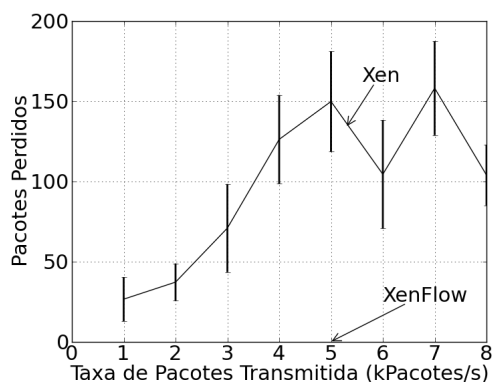
(a) Pacotes de 64 bytes



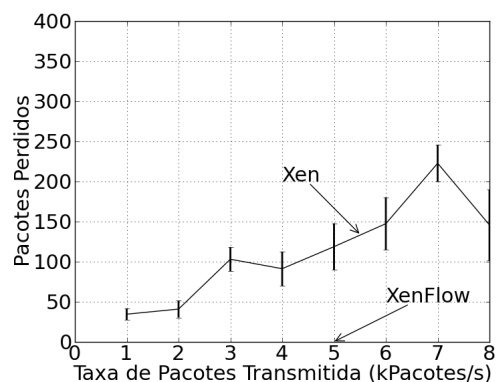
(b) Pacotes de 1500 bytes

**Figura 8. Tempo de total de duração do processo de migração.**

O terceiro experimento apresenta o número de pacotes perdidos durante a migração convencional e a comprovação de perda zero no XenFlow. A Figura 9 revela que durante a migração no XenFlow não há perdas de pacotes. Durante a migração, há um momento em que o caminho antigo e o novo caminho, respectivamente, o que passa pelo roteador físico de origem e o que passa de destino, estão ativos. Assim, pode ocorrer de a máquina receptora receber pacotes desordenados e até mesmo duplicados. Embora gere um maior custo de processamento, esse procedimento é necessário para garantir que não haverá pacotes perdidos entre a interrupção do plano de dados antigos e a ativação do novo. A Figura 9 mostra ainda que a perda zero de pacotes do sistema XenFlow é independente da taxa de pacotes encaminhados. Já a migração nativa do Xen apresenta perdas maiores para taxas maiores de pacotes enviados. Isso é reflexo do tempo de interrupção do serviço de encaminhamento, como visto na Figura 7. Como o tempo de interrupção do encaminhamento no Xen nativo é aproximadamente constante, a quantidade de pacotes perdidos nesse intervalo de tempo tende a aumentar com o aumento da taxa de pacotes encaminhados.



(a) Pacotes de 64 bytes



(b) Pacotes de 1500 bytes

**Figura 9. Número de pacotes perdidos em função da taxa de pacotes transmitida.**

## 5. Conclusão

Este artigo propõe o XenFlow, um sistema de processamento de fluxos que provê uma forma robusta e eficiente de migração de topologias virtuais. O principal objetivo do XenFlow é realizar migrações de roteadores virtuais, com perda zero de pacotes e eliminar a necessidade de túneis ou mecanismos externos para a migração de enlaces. A proposta usa técnica de separação de planos através da aplicação no controlador Nox que controla o plano de dados baseada em regras de encaminhamento atualizadas por um *daemon* que executa em cada roteador virtual. Os resultados mostram que o tempo de interrupção do plano de controle no sistema proposto chega a ser até 30 vezes menor do que o tempo de interrupção na abordagem nativa do Xen. Os resultados também demonstram que o tempo total de migração aumenta quando comparamos o sistema XenFlow com a migração nativa do Xen. Esse último resultado deve-se ao fato de o XenFlow introduzir novas etapas em relação à migração do Xen, durante o processo de migração da topologia virtual. Contudo, o aumento do tempo total de migração não é um fator significativo para a migração de roteadores virtuais e apenas estabelece o tempo mínimo entre duas migrações consecutivas. Os resultados obtidos mostram que a migração de um roteador virtual sobre o sistema XenFlow ocorre sem perda de pacotes, o que torna o sistema adequado ao cenário de redes virtuais ao contrário da migração nativa da plataforma de virtualização Xen.

Como trabalhos futuros, pretende-se desenvolver novas aplicações além de roteamento. Nesse novo contexto, as máquinas virtuais passam a prestarem serviços antes destinados à *middle boxes*, tais como balanceador de carga e *firewall*. Assim, o sistema XenFlow torna-se uma plataforma para a implementação de nós especializados, que podem ser migrados para entre diferentes pontos da rede.

## 6. Referências

- [Bolla et al. 2009] Bolla, R., Bruschi, R., Davoli, F., and Ranieri, A. (2009). Energy-aware performance optimization for next-generation green network equipment. In *Proceedings of the 2nd ACM SIGCOMM workshop on Programmable routers for extensible services of tomorrow*, pages 49–54. ACM.
- [Clark et al. 2005] Clark, C., Fraser, K., Hand, S., Hansen, J., Jul, E., Limpach, C., Pratt, I., and Warfield, A. (2005). Live migration of virtual machines. In *Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation-Volume 2*, pages 273–286. USENIX Association.
- [Egi et al. 2008] Egi, N., Greenhalgh, A., Handley, M., Hoerd, M., Huici, F., and Mathy, L. (2008). Towards high performance virtual routers on commodity hardware. In *Proceedings of the 2008 ACM CoNEXT Conference*, pages 1–12. ACM.
- [Feamster et al. 2007] Feamster, N., Gao, L., and Rexford, J. (2007). How to lease the Internet in your spare time. *ACM SIGCOMM Computer Communication Review*, 37(1):61–64.
- [Fernandes et al. 2010] Fernandes, N., Moreira, M., Moraes, I., Ferraz, L., Couto, R., Carvalho, H., Campista, M., Costa, L., and Duarte, O. (2010). Virtual networks: Isolation, performance, and trends. *Annals of Telecommunications*, pages 1–17.

- [Gude et al. 2008] Gude, N., Koponen, T., Pettit, J., Pfaff, B., Casado, M., McKeown, N., and Shenker, S. (2008). NOX: towards an operating system for networks. *ACM SIGCOMM Computer Communication Review*, 38(3):105–110.
- [McKeown et al. 2008] McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S., and Turner, J. (2008). OpenFlow: enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review*, 38(2):69–74.
- [Pisa et al. 2010] Pisa, P., Fernandes, N., Carvalho, H., Moreira, M., Campista, M., Costa, L., and Duarte, O. (2010). Openflow and xen-based virtual network migration. In Pont, A., Pujolle, G., and Raghavan, S., editors, *Communications: Wireless in Developing Countries and Networks of the Future*, volume 327 of *IFIP Advances in Information and Communication Technology*, pages 170–181. Springer Boston.
- [Ratnasamy et al. 2005] Ratnasamy, S., Shenker, S., and McCanne, S. (2005). Towards an evolvable Internet architecture. *ACM SIGCOMM Computer Communication Review*, 35(4):313–324.
- [Sherwood et al. 2009] Sherwood, R., Gibb, G., Yap, K., Appenzeller, G., Casado, M., McKeown, N., and Parulkar, G. (2009). Flowvisor: A network virtualization layer. Technical report, Tech. Rep. OPENFLOW-TR-2009-01, OpenFlow Consortium.
- [Wang et al. 2008] Wang, Y., Keller, E., Biskeborn, B., van der Merwe, J., and Rexford, J. (2008). Virtual routers on the move: live router migration as a network-management primitive. *ACM SIGCOMM Computer Communication Review*, 38(4):231–242.
- [Wang et al. 2007] Wang, Y., van der Merwe, J., and Rexford, J. (2007). VROOM: Virtual routers on the move. In *Proc. ACM SIGCOMM Workshop on Hot Topics in Networking*. Citeseer.