Recuperação Ciente de QoS e Sensível ao Contexto: Uma Heurística de Tolerância a Falhas para Redes de *Backbone*

Igor D. B. Caldeira[1], Italo V. S. Brito[1], Maycon L. M. Peixoto[1], Leobino N. Sampaio[1]

[1]Programa de Pós-Graduação em Ciência da Computação (PGCOMP)
Departamento de Ciência da Computação – Universidade Federal da Bahia (UFBA)
Salvador – BA – Brasil

{igor.david, italovalcy, maycon.leone, leobino}@ufba.br

Abstract. *Current fault tolerance strategies in backbone networks usually react based on the condition set by the operator instead of network historical performance data. This can generate congestion or impact the application's QoS when links fail due to the to new topology and traffic prioritization requirements. In Software Defined Networking (SDN), the logically centralized global network view provides new perspectives to deal with network failures. This paper proposes a hybrid heuristic for locally link failover, providing fast protection, followed by optimizing the flow path with QoS-aware recovery. Experiments emulating the characteristics of the Rede Ipê/RNP showed significant improvements in QoS metrics compared to conventional strategies.*

Resumo. *As estratégias atuais de tolerância a falhas em redes de backbone geralmente reagem com base no estado configurado pelo operador e não com base no histórico de desempenho da rede. Face às características da nova topologia e das classes de priorização, isso pode gerar congestionamento ou impacto na QoS das aplicações quando da ocorrência de falhas. A visão global e logicamente centralizada de controladores em redes definidas por software abre novas perspectivas para tratamento de falhas na rede. Este trabalho propõe uma heurística híbrida para tratar localmente falhas dos enlaces, proporcionando rápida proteção, seguido pela otimização no caminho dos fluxos com recuperação ciente de QoS. Experimentos conduzidos emulando características da Rede Ipê/RNP apontaram ganhos significativos em métricas de QoS frente à estratégias convencionais.*

1. Introdução

Falhas e congestionamento de enlaces são dois problemas que impõem desafios para operadores de rede [Lin et al. 2016]. Para superar esses desafios, são aplicadas técnicas de proteção de enlaces e engenharia de tráfego. Não obstante, quando da ocorrência de falhas, essas estratégias de tolerância a falhas geralmente reagem com base no estado configurado pelo operador e não com base no histórico de desempenho da rede face às características da nova topologia e das classes de priorização das aplicações.

Ao centralizar o plano de controle, o paradigma das Redes Definidas por *Software* (do inglês *Software Defined Networking – SDN)*, trouxe novas perspectivas acerca do desenvolvimento de soluções de Tolerância a Falhas (TF) em redes [Kreutz et al. 2014, Adrichem et al. 2014, Stephens et al. 2016]. Aliando visão global, a programabilidade e

a separação entre plano de dados e plano de controle, os pesquisadores têm voltado sua atenção ao provimento de QoS às diversas aplicações de rede [Karakus and Durresi 2017].

Para tratar falhas em enlaces de redes SDN, duas estratégias são tipicamente adotadas: proteção e recuperação [Lin et al. 2016]. A proteção segue uma abordagem pró-ativa, onde regras de fluxos são implantadas antecipadamente nos comutadores, de forma que, na ocorrência de uma falha, rotas alternativas possam ser utilizadas sem a necessidade de nova comunicação com o plano de controle [Lin et al. 2016]. A recuperação, por sua vez, segue uma abordagem reativa na qual os comutadores se comunicam com o controlador de rede ao detectar um evento de falha [Dusia and Sethi 2016], ficando sujeitos à disponibilidade e ao atraso da comunicação com o plano de controle [Cascone et al. 2017]. O controlador então calcula uma nova topologia lógica que define um caminho alternativo e informa as respectivas regras de encaminhamento aos comutadores. Apesar dos benefícios, ambos os mecanismos apresentam desvantagens [Chen et al. 2015]. Na proteção, a implantação antecipada não garante a escolha de melhores caminhos alternativos para eventuais falhas, dado que a rede pode apresentar condições de desempenho diferentes ao ocorrer uma falha. Já a recuperação incorre em aumento no tempo de convergência da rede devido à necessidade de comunicação com o controlador. Além disso, nos trabalhos relacionados, a definição de novas rotas pouco considera os parâmetros de QoS das classes de tráfego de aplicações mais relevantes.

Desta forma, este trabalho propõe o uso de uma heurística de TF híbrida ciente de QoS e sensível ao contexto para prover recuperação de falhas com garantias de desempenho para as aplicações. Com esse objetivo, são considerados pesos de métricas da rede no cálculo de caminhos, visando obter um melhor aproveitamento dos caminhos alternativos e recuperação de falhas mais eficaz. Os pesos das métricas são definidos a partir da observação do histórico de desempenho da rede. A heurística então propõe um re-roteamento imediato do modo proativo, de forma a prover uma rápida recuperação (FF, do inglês *Fast Failover*) que não dependa do controlador de rede. Em seguida, no modo reativo, o controlador faz o cálculo de novos caminhos e promove um novo re-roteamento para os casos em que rotas mais eficientes são encontradas. Desta forma, a heurística proposta permite ao plano de controle fazer a escolha dinâmica de melhores caminhos na topologia, onde os *switches* são configurados com regras de encaminhamento que priorizam requisitos específicos de classes de aplicação. No caso específico deste trabalho, foram priorizados os caminhos com melhor vazão da rede.

A avaliação da solução consistiu na implementação da heurística em ambiente emulado reproduzindo características da rede acadêmica brasileira (Rede Ipê), mantida pela Rede Nacional de Ensino e Pesquisa (RNP)[1]. Resultados experimentais demonstraram a eficácia da heurística proposta, que apresentou ganhos tanto na recuperação proativa quanto na reativa. As principais contribuições deste trabalho são: i) a definição de uma estratégia para cálculo de distância que se baseia no uso de uma função ponderada das métricas de desempenho; ii) uma heurística híbrida que combina técnicas pró-ativas e reativas no tratamento de falhas.

Este artigo está organizado da seguinte forma: a Seção 2 apresenta os trabalhos relacionados; a Seção 3 descreve a heurística proposta; a Seção 4 descreve uma estratégia

[1]https://www.rnp.br/servicos/conectividade/rede-ipe

para determinação de parâmetros da heurística; a Seção 5 apresenta uma avaliação de desempenho da proposta; e a Seção 6 apresenta as conclusões do trabalho.

2. Trabalhos Relacionados

Na ocorrência de falhas, soluções de TF pró-ativas em SDN apresentam um melhor tempo de resposta na recuperação das falhas caracterizadas por interrupções de interfaces ou enlaces [Chen et al. 2015], porém não exploram a visão global do controlador de rede para fazer o cálculo de novos caminhos sob demanda e de acordo com o tráfego corrente. Por tais motivos, visando dirimir as desvantagens das soluções pró-ativas e reativas, alguns trabalhos relacionados têm sido propostos na literatura [Fonseca and Mota 2017].

Em [Sahri and Okamura 2014], os autores apresentam uma política de TF capaz de implementar o FF para minimizar o tempo de convergência, tratando localmente a disponibilidade, ao mesmo tempo que informa a falha ao controlador; paralelamente, buscam reduzir o congestionamento da rede ao priorizar a seleção por caminhos com menores filas de encaminhamento nos comutadores. Já em [Machado et al. 2014], os autores apresentam uma abordagem que permite ao controlador verificar requisitos específicos para certos tipos de tráfego e, assim, identificar previamente (pró-ativamente e com base de métricas de desempenho da rede) o melhor caminho na topologia para que os *switches* envolvidos sejam configurados com as respectivas regras de encaminhamento. No trabalho [Karakus and Durresi 2017] é apresentado um levantamento de aplicações SDN cientes de QoS para diferentes contextos, bem como os recursos de QoS disponíveis nas diferentes versões do protocolo OpenFlow. Em [Lin et al. 2016], o FF e o congestionamento são tratados separadamente, sendo o controle de tráfego aplicado apenas quando um enlace está próximo de seu limite, e não demonstra preocupação com outras características da rede capazes de influenciar no QoS do tráfego.

Apesar de tais iniciativas apresentarem contribuições, nenhuma considera características de desempenho do caminho antes de sua seleção, nem referentes aos valores nominais e nem referentes ao histórico de aferições. Sendo assim, a heurística descrita na próxima seção apresenta duas contribuições principais em relação a tais trabalhos relacionados: (i) o aprimoramento da combinação de técnicas pró-ativas e reativas proposta por [Sahri and Okamura 2014], considerando características da aplicação e diferenciação de políticas de encaminhamento baseada em parâmetros de Qualidade do Serviço [Machado et al. 2014]; e (ii) uma nova estratégia de seleção de caminhos que se baseia no histórico das métricas de desempenho da rede.

3. Heurística Proposta

A heurística de TF proposta baseia-se na garantia de QoS das aplicações diante da ocorrência de falhas. O objetivo da estratégia é melhorar o desempenho e o tempo de convergência na recuperação de falhas, a partir do histórico de desempenho da rede e do contexto (aplicação, status da rede, usuário). Para isso, a heurística utiliza as medições mais recentes das métricas de desempenho de rede a fim de definir novos caminhos ou realocar caminhos existentes, visando contemplar classes de tráfego mais importantes do domínio em questão. Sendo assim, através da heurística proposta busca-se obter: i) rápido tempo de recuperação pós-falha; e ii) seleção de caminhos ciente do desempenho da rede. Nesta seção a heurística será apresentada em maiores detalhes.

3.1. Visão Geral da Heurística

A Figura 1 ilustra um cenário em que a heurística é aplicada. Neste exemplo, em caso de falha no caminho principal (*R1-R3*), o caminho alternativo para transferência de dados é selecionado com base nos enlaces com maior banda disponível (*R1-R2-R3*), ao passo que o caminho alternativo para tráfego de Voz sobre IP prioriza o menor atraso (*R1-R4-R3*).

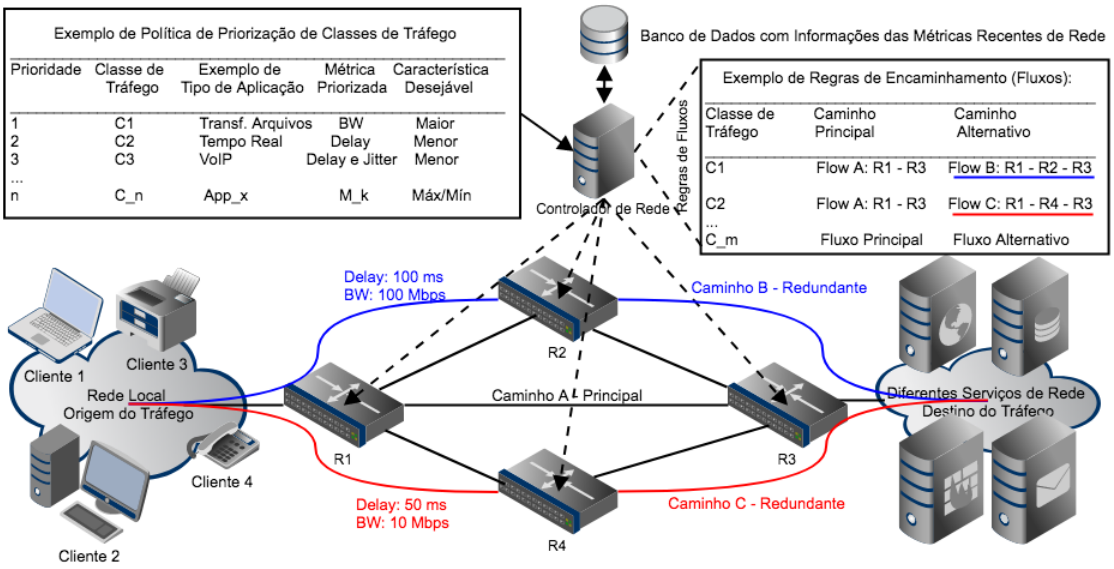


Figura 1. Cenário simplificado de aplicação da heurística, com base em uma política de priorização de classes de tráfego de aplicações.

A heurística é híbrida porque combina um mecanismo proativo para proteção do enlace com um mecanismo reativo para recuperação e readequação de fluxos em conformidade com as classes de tráfego definidas. Assim, de forma pró-ativa a estratégia obtém um menor tempo de recuperação através da exploração de caminhos principal e alternativo previamente estabelecidos e configurados nos comutadores. Na ocorrência de falhas no caminho principal, a indisponibilidade é tratada localmente e de forma automática. Já a abordagem reativa é utilizada para explorar a visão global do controlador SDN para fazer o recálculo dos caminhos a partir das métricas de desempenho de rede. Quando um enlace volta a ficar ativo, o controlador recalcula os melhores caminhos e, se for o caso, atualiza as tabelas de encaminhamento.

O controlador emprega uma lógica específica para as regras de encaminhamento, de acordo com uma política de priorização de classes de tráfego definida no plano de gerenciamento. A partir desta política de priorização, o controlador seleciona a melhor rota para aplicação com base no histórico de desempenho da rede e calcula caminhos alternativos para deixá-los pré-configurados a fim de tratar imediatamente falhas nos enlaces. Ao receber informações de uma falha, o controlador refaz os cálculos de rotas das aplicações afetadas e, se for o caso, reconfigura a aplicação por um caminho mais adequado.

É importante salientar que, embora o tratamento de falhas no controlador não seja alvo deste trabalho, falhas no plano de controle não prejudicam a proteção dos enlaces da rede, tendo em vista que caminhos alternativos são previamente configurados.

Todo o processo que envolve a heurística proposta consiste em: (i) definir políticas de priorização a partir do plano de gerenciamento da rede; (ii) identificar métricas de QoS mais importantes a partir da política; (iii) contabilizar os níveis de importância das métricas de desempenho em cada classe de tráfego; e (iv) selecionar proativa e reativamente os caminhos. Para este último item, faz-se a) monitoramento e detecção de falhas, b) seleção de caminhos sob demanda, c) instalação proativa de regras de encaminhamento e proteção, e d) seleção e realocação reativa de fluxos mediante a ocorrência de falhas. A Figura 2 apresenta um fluxograma da heurística que resume seu funcionamento geral.

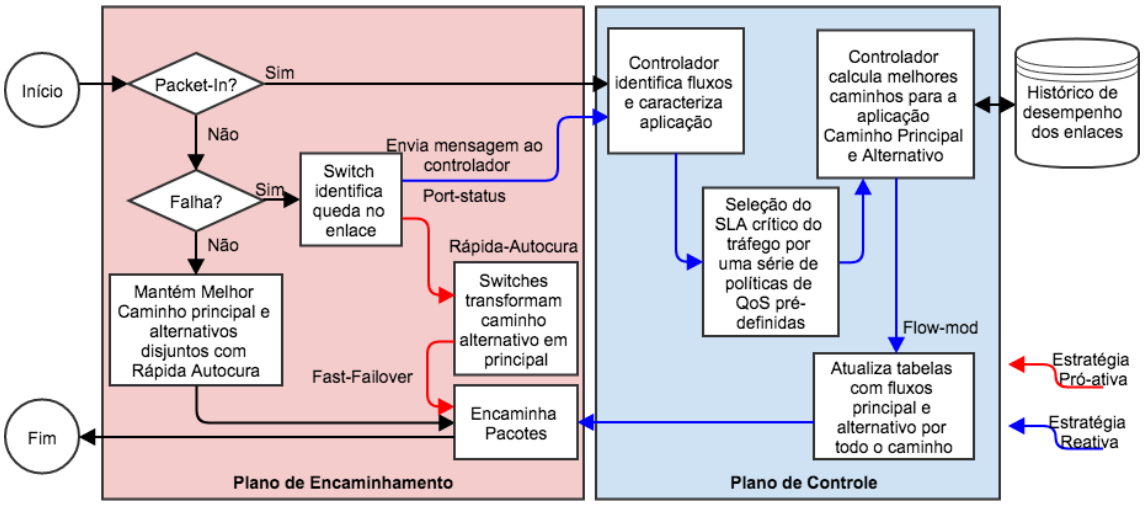


Figura 2. Fluxograma da Heurística de Tolerância a Falhas Híbrida Proposta.

Primeiramente, as demandas de tráfego para novos fluxos são recebidas e os caminhos principal e alternativo são instalados na rede, viabilizando que as quedas de enlaces sejam tratadas pelos próprios comutadores. Tal característica é alcançada através do emprego das ações de tabela de grupo do OpenFlow 1.3, em particular do tipo *Fast Failover*, proporcionando rápida recuperação de forma pró-ativa.

Por outro lado, diante da ocorrência de uma falha, os comutadores notificam o controlador acerca da indisponibilidade e ele, por sua vez, inicia a estratégia de recuperação. Considerando apenas os enlaces disponíveis na topologia restante, o caminho de cada aplicação afetada é recalculado e as tabelas de fluxo dos comutadores são atualizadas. O recálculo de caminhos funciona de acordo com a política de priorização de classes de tráfego da aplicação em questão.

Portanto, o cálculo dos caminhos na topologia é iniciado no controlador tanto a partir de pacotes *packet-in* quanto de mensagens *port-status*, após a caracterização do tráfego e identificação da política de encaminhamento associada. Já as atualizações das tabelas de fluxo são realizadas através de mensagens *flow-mod*, conforme algoritmo de definição de caminhos discutido a seguir.

3.2. Algoritmo Utilizado para Cálculo de Caminhos

Esta proposta apresenta uma alternativa de composição de métricas que, historicamente, influenciam com diferentes graus de importância no QoS da rede. Tais métricas são combinadas para definição de pesos dos enlaces e utilizadas no algoritmo de cálculo de caminhos. Portanto, a heurística é capaz de caracterizar variados tipos de tráfego e aplicar

algoritmos de encaminhamento específicos para cada um, segundo políticas predefinidas aplicadas no controlador da rede.

Para a implementação de uma seleção de caminhos com tais características considerou-se diferentes pesos para os enlaces no algoritmo de Dijkstra, amplamente utilizado no cálculo de menor caminho em grafos. A primeira adaptação consistiu no ajuste de uma constante, de valor "C", que o algoritmo original considera como 1. Assim, é possível regular os pesos dos saltos no algoritmo, elevando a significância das métricas de rede na determinação do caminho e permitindo até que sejam selecionados caminhos com maior número de saltos, para se obter um melhor QoS. Neste trabalho, a segunda adaptação é no acréscimo de dois fatores no cálculo do peso de um enlace: (i) banda residual; e (ii) atraso de pacotes. Vale salientar que qualquer métrica disponível na rede pode ser utilizada, de acordo com o contexto a ela relacionado.

A Equação 1 apresenta o método proposto para ser utilizado no cálculo dos pesos de cada enlace, tendo como premissa que os fatores de desempenho são responsáveis pelos impactos no desempenho da rede. A fim de evitar que fossem usados valores absolutos com diferentes ordens de grandeza, as métricas da equação são previamente normalizadas. Na equação, as métricas com índice "m" são obtidas da média do tráfego de rede no período observado, como é o caso da largura de banda e do atraso. Já aquelas com índice "max", a saber $BWEnlace_{max}$ e $Atraso_{max}$, representam a largura de banda nominal do enlace e o atraso máximo medido no período, respectivamente.

$$peso[enlace] = C + \alpha \times \frac{BW_m}{BWEnlace_{max}} + \beta \times \frac{Atraso_m}{Atraso_{max}} + ... + \zeta \times \frac{Metrica_m}{Metrica_{max}}. \quad (1)$$

De acordo com a Equação 1, o peso de um enlace reflete suas características históricas de desempenho e ajuda na escolha de caminhos mais apropriados para o perfil de tráfego da aplicação. Em particular, pode-se considerar que: (i) o peso é inversamente proporcional à capacidade ociosa do enlace, ou seja, quanto maior largura de banda disponível, menor o peso e, portanto, maior a chance de sua seleção na escolha do caminho; e (ii) o peso é diretamente proporcional aos atrasos de pacotes – quanto maiores forem seus valores, mais estes caminhos tendem a ser evitados. A equação ora proposta pode ainda ser estendida para outros contextos ou aplicações, bastando complementá-la com as métricas relacionadas ao QoS desejado. Os parâmetros α, β e ζ na Equação 1 são utilizados para dinamicamente ajustar a influência de cada métrica no perfil de tráfego.

O cálculo da influência é baseado na calibração da proposta com base no histórico de medições de uso da rede e seu impacto nas variáveis de resposta observados. Para isso, foi utilizada a metodologia de avaliação de desempenho *K Fatorial* [Jain 1990].

A metodologia proposta foi validada considerando um recorte do backbone da Rede Ipê/RNP, cujas métricas de desempenho estão disponíveis em portais web públicos do MonIpê[2] e do Panorama de Tráfego[3]. Neste caso, α, β, ... e ζ definem o grau de importância de cada métrica de desempenho no cálculo da distância. A função dos coeficientes citados é fornecer um fator capaz de corrigir e adaptar a representatividade das características dos diferentes enlaces segundo o respectivo impacto, adicionando ao

[2]http://monipe-portal.rnp.br/

[3]https://www.rnp.br/servicos/conectividade/trafego

controlador a possibilidade de diferenciar cada enlace na seleção de melhores caminhos para a aplicação. Estas métricas de desempenho são coletadas regularmente e usadas na atualização do cálculo dos coeficientes. Dessa forma, a estratégia consegue acomodar dinamicamente mudanças no perfil de tráfego da rede. A determinação destes coeficientes será objeto de estudo da seção seguinte.

4. Determinação dos fatores de influência das métricas no peso do enlace

A determinação dos fatores de influência das métricas da heurística proposta neste artigo consistiu em uma série de experimentos conduzidos através de emulação com o ambiente do Mininet, buscando identificar o relacionamento das métricas de rede ao QoS alvo.

4.1. Métricas de Avaliação dos Experimentos

Apesar da Heurística poder ser aplicada para qualquer métrica, o presente estudo focou na busca pela máxima banda residual, proporcionando o menor tempo de transferência de arquivos. Para avaliar os experimentos, as métricas consideradas na validação foram:

- (i) **Banda Residual**: é a capacidade de transferência disponível entre dois *hosts*.
- (ii) **Latência**: é o atraso bidirecional de um pacote ao percorrer a rede da origem ao destino e retornar;
- (iii) **Perda**: é a porcentagem de pacotes descartados no caminho, ou seja, que não chegam ao seu destino;
- (iv) **Jitter**: é a variação do atraso entre dois *hosts*;
- (v) **Tempo de Transferência**: é o período de transferência de um arquivo; e
- (vi) **Tempo de Recuperação**: Consiste no período de tempo entre a interrupção do encaminhamento de pacotes pela rede e o reestabelecimento do tráfego.

As métricas (i), (ii), (iii) e (iv) foram consideradas nos testes entre os clientes e servidores emulados na topologia da Rede Ipê, no caminho entre o PoP da Bahia e o PoP São Paulo. O objetivo foi de quantificar o impacto das características dos enlaces na banda residual dos caminhos emulados, para calibração da heurística e da Equação 1, verificando a parcela reativa através da transferência de um arquivo, viabilizando uma análise sobre (v). Posteriormente, na avaliação da solução pró-ativa, foram ressaltados os impactos na ocorrência de falhas comparando a proposta híbrida com um referencial legado OSPF, e observando as respostas das métricas (ii) e (iii). Foi adicionado ainda um referencial SDN, o ONOS[4], para comparação no tempo de recuperação (vi).

4.2. Metodologia Adotada nos Experimentos

Os experimentos foram conduzidos por meio de emulação. Os valores reais referentes à largura de banda, atraso, perda de pacotes e jitter em cada enlace foram obtidos a partir de dados da Rede Ipê durante o mês de Novembro de 2017. Para simplificar o ambiente, o passo seguinte foi definir um recorte do cenário da Rede Ipê, conforme a Figura 3, trecho da rede entre os PoPs da Bahia e São Paulo.

A validação dos mecanismos de proteção e recuperação da proposta híbrida é demonstrada através da reprodução experimental da falha e observação das etapas da recuperação expostas na Figura 4, com um estudo comparativo com o protocolo legado

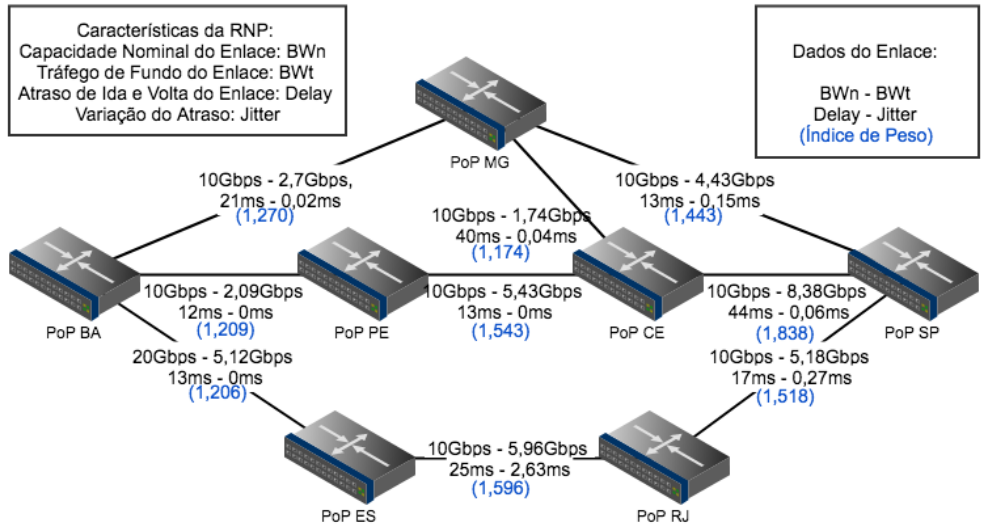
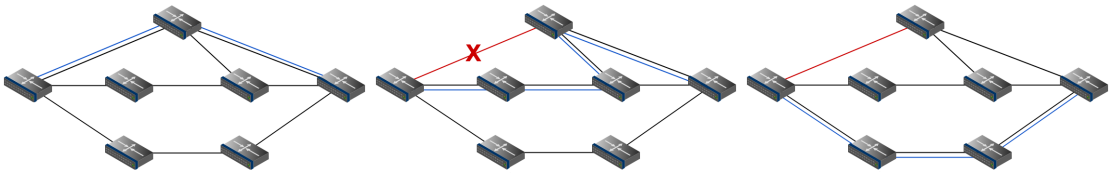


Figura 3. Topologia do Cenário da Rede Ipê.



(a) Caminho Principal (b) Caminho Redundante (c) Caminho Recalculado

Figura 4. Etapas da Recuperação Proposta no Cenário da Rede Ipê.

OSPF. Essas etapas demonstram a operação completa da heurística diante da ocorrência de falhas, inicialmente em sua parcela pró-ativa e posteriormente em sua parte reativa.

Os fluxos instalados são destacados com uma linha azul na Figura 4a, onde inicialmente o caminho principal é BA-MG-SP. É então reproduzida uma falha ao interromper logicamente uma das interfaces em uso do enlace BA-MG, em vermelho, conforme ilustrado na Figura 4b, e imediatamente os elementos de encaminhamento se utilizam da solução pró-ativa alternativa via Fast-Failover para contornar a falha pelo caminho redundante BA-PE-CE-MG-SP. Futuramente, em um intervalo de tempo configurável, a solução otimizada para proporcionar o melhor QoS atualiza as tabelas de fluxo, conforme a Figura4c, adotando o caminho recalculado BA-ES-RJ-SP.

4.3. Definição dos Pesos das Métricas de Desempenho

Para cada métrica de QoS, é necessário identificar e ponderar seus respectivos fatores de influência. O cenário de experimentação foi reproduzido conforme a Figura 3 com respectivas bandas nominais, tráfegos de fundo, atraso e jitter. Com foco na máxima vazão, foram definidos 4 casos de estudo, com variação de 2 fatores: C1 (sem atraso e sem *background*), C2 (com atraso da RNP e sem *background*), C3 (sem atraso e com *background* da RNP) e C4 (com atraso da RNP e com *background* da RNP). Neles foi realizada a

[4]https://onosproject.org

calibração do ambiente, utilizando o *iperf3* para aferição da banda residual, buscando a identificação da relação entre as métricas de rede e a maior taxa de transferência. Cada experimento foi repetido 10 vezes, mensurando a média das larguras de banda TCP recebidas, produzindo os resultados da Figura 5.

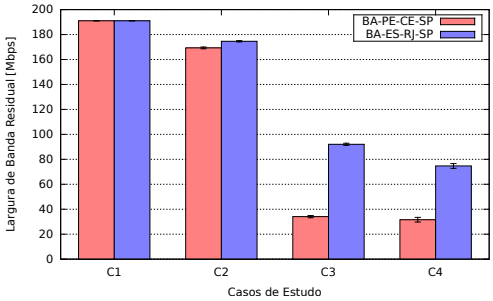


Figura 5. Resultados da Calibração na Rede Ipê.

Os resultados dos experimentos dos casos 3 e 4 da Figura 5, que concorreram com tráfego de fundo proporcional aos valores da RNP, causaram o maior impacto do tráfego de fundo na banda residual. Do caso 2 para o 1 também é perceptível uma pequena queda devido ao atraso, que prejudica a confirmação de recebimento dos pacotes, mas em uma proporção muito menos significativa. Isso convalida a ideia de que os caminhos mais sobrecarregados devam ser evitados, para se obter uma maior vazão.

Os experimentos comparam a vazão por dois caminhos distintos, com o objetivo de identificar alguma previsibilidade e aprimorar os encaminhamentos sensíveis à aplicação, quando este for o foco principal do QoS. Existiram poucos *outliers* e o desvio padrão foi pequeno, sendo possível garantir o intervalo de confiança em 95%. A partir da influência dos fatores neste ambiente, foram definidos os valores para os pesos α e β da Equação 1, de forma a priorizar caminhos de característica mais relevantes para o QoS.

Através da correlação dos experimentos e análise do desempenho no trecho, aplicando a Metodologia K Fatorial, foi calculada a influência do fator banda isoladamente como superior a 90%, implicando em um valor de $\alpha = 0.90$. Outros 5% foram referentes ao caminho, e a parcela do atraso foi inferior a 2%, sugerindo um $\beta = 0.02$. As influências das perdas e do jitter não foram consideradas, e uma lógica similar permite descartar também o fator atraso no cálculo dos pesos. Para simplificar os cálculos, uma vez que a significância da banda ociosa foi muito maior, pode-se simplificar a Equação 1 com os valores de $\alpha = 1$ e $\beta = 0$ para a Equação 2.

$$peso[enlace] = 1 + \frac{BW_m}{BWEnlace_{max}}. \tag{2}$$

Em uma rede convencional, de acordo com a análise experimental apresentada, a sensibilidade ao tráfego recomenda fazer a identificação de grandes tráfegos e encaminhá-los para um caminho com a máxima vazão, obtendo melhor desempenho e viabilizando aproveitamento máximo da infraestrutura de rede com a ciência de contexto. Isto é possível através do protocolo *OpenFlow*, por viabilizar o encaminhamento diferenciado para portas ou tipos de serviço, aplicando tratamento especial a fluxos que necessitem de maior vazão, como tráfego multimídia ou transferência de grandes volumes de dados.

A Figura 3 apresenta também os pesos usados no cálculo da distância atribuídos em cada enlace, segundo a base de dados de uso obtida da Rede Ipê, destacados entre parênteses. Nela é possível identificar a inteligência adicionada à seleção do caminho com máxima vazão. Executando o algoritmo proposto são traçados os caminhos listados na Tabela 1, em ordem de prioridade, por apresentarem menor distância.

Tabela 1. Definição dos Caminhos Prioritários para a Maior Vazão.

Prioridade	Caminho	Pesos	Distância
1	BA-MG-SP	1,270+1,443	2,713
2	BA-MG-CE-SP	1,270+1,174+1,838	4,282
3	BA-ES-RJ-SP	1,206+1,596+1,518	4,320
4	BA-PE-CE-SP	1,209+1,543+1,838	4,590
5	BA-PE-CE-MG-SP	1,209+1,543+1,174+1,443	5,369

A próxima seção apresenta de forma detalhada a experimentação e a análise dos resultados dos mecanismos de prevenção e recuperação.

5. Experimentação e Análise de Resultados

A avaliação da proposta foi realizada emulando características da Rede Ipê/RNP e comparando o desempenho com uma abordagem tradicional baseada em OSPF e uma abordagem SDN baseada no ONOS. O OSPF foi configurado de forma especial, para fornecer rápida convergência[5], diferente do seu modo convencional de operação.

5.1. Planejamento de Experimentos

Os experimentos foram conduzidos utilizando o modelo de planejamento de experimentos fatorial completo [Jain 1990], considerando os seguintes fatores que influenciam no desempenho do sistema: [Fator **A** - Estratégia]= {Híbrida, OSPF}; [Fator **B** - Tráfego de Background]={com, sem}; [Fator **C** - Topologia]= {Simples, RNP}; [Fator **D** - Tipo de Falha]= {Simples, Roteador}. Os fatores e níveis do sistema foram definidos através de valores característicos da Rede Ipê/RNP[6] e valores mais prováveis utilizados na literatura, permitindo avaliar a completude do efeito que essas variáveis provocam ao ambiente. Todos os experimentos foram analisados a partir do intervalo de confiança[7], da média e do desvio padrão aferidos. Esses parâmetros são utilizados como base para o cálculo da soma dos quadrados, resultando na influência de cada fator nas variáveis de resposta.

O ambiente de experimentação possui as seguintes características: (i) servidor Intel Xeon E52609 2.4Ggz, 16GB de RAM; (ii) Linux 3.16.0-4-amd64; (iii) Mininet 2.2.2; (iv) Open vSwitch 2.5.4-1; (v) Ryu 4.15; (vi) gerador de tráfego *iperf3*.

A Figura 6 apresenta a observação da normalidade na execução dos experimentos. O esperado é que os pontos do gráfico, relacionados aos experimentos, residam sobre ou próximos à linha normal, como é observado.

[5]O OSPF pode ser configurado para rápida convergência com o parâmetro *ip ospf dead-interval minimal*

[6]Os tipos de falha (simples e roteador) foram baseadas naquelas mais comuns no *backbone* RNP, segundo relatórios de disponibilidade públicos.

[7]Utilização da distribuição *t-student* com replicações de 10 execuções por experimento e $\alpha = 0,05\%$.

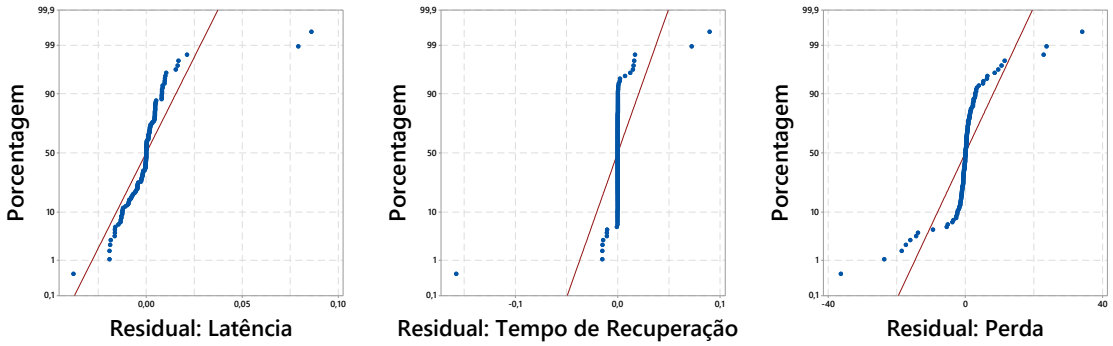


Figura 6. Observação da Normalidade na Execução dos Experimentos.

A Figura 7 mostra o gráfico pareto e o gráfico normalizado dos efeitos para o projeto fatorial 2^k. Esse gráfico revela o grau de influência que os fatores exercem sobre as variáveis de resposta: latência (RTT), tempo recuperação e perda. O fator topologia influenciou de maneira mais significativa a variável de resposta latência. A influência da topologia na latência se dá principalmente pelo uso de características reais da Rede Ipê comparada com uma topologia simples. Por outro lado, o fator estratégia foi o responsável por exercer maior influencia sobre as variáveis de resposta tempo de recuperação e perda. Essa informação confirma a hipótese de que a escolha da estratégia de recuperação produz alteração significativa nas variáveis de resposta, conforme será analisado posteriormente.

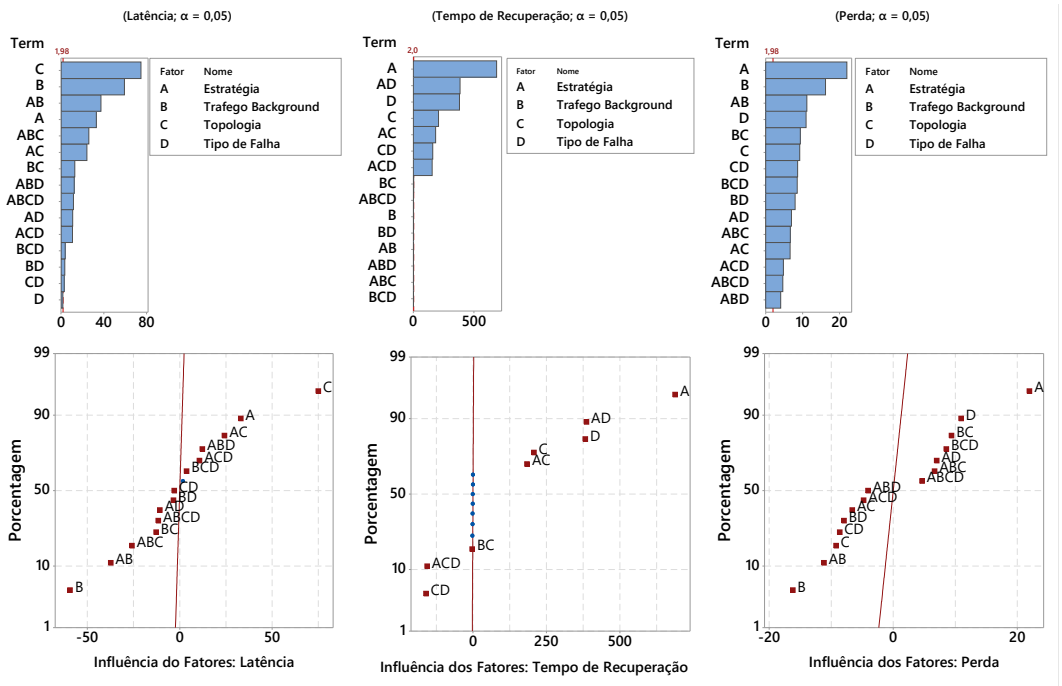


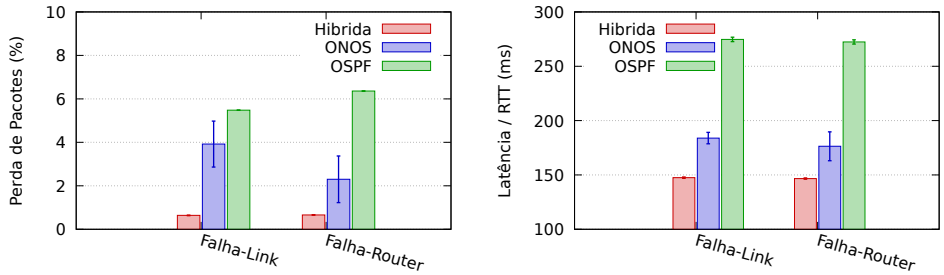
Figura 7. Influência dos Fatores.

5.2. Análise dos Resultados

Considerando o planejamento de experimentos e os fatores que mais influenciaram nas variáveis de resposta, incluiu-se também a abordagem de orquestração da rede através do ONOS. O controlador SDN ONOS possui aplicações para configuração dos enlaces através de *intents*; em particular utilizou-se o tipo de enlace ponto-a-ponto, cuja estratégia

de recuperação de falha se dá através do recálculo de rotas no controlador ao receber uma mensagem de mudança do estado do enlace. A seguir as métricas serão analisadas para cada estratégia considerando o cenário com tráfego de fundo e topologia RNP.

A Figura 8a representa a perda de pacotes acumulada na rede considerando o tráfego entre os roteadores BA-SP (150Mbps) e o tráfego de fundo na rede (70Mbps). Após a ocorrência de falhas na rede (i.e. enlace BA-MG ou roteador MG), as estratégias convencionais (i.e. OSPF e ONOS) aplicam o algoritmo de recálculo para recuperação e definem um novo caminho sem considerar as características de desempenho atuais do enlace, o que pode gerar concorrência com o tráfego de fundo e congestionamento. É possível observar esse comportamento através da Figura 8a pois a perda de pacotes no OSPF e ONOS ficaram elevadas, $5,48 \pm 0,001\%$ e $3,92 \pm 1,05\%$ respectivamente para falha de enlace e $6,36 \pm 0,007\%$ e $2,30 \pm 1,07\%$ respectivamente para falha de roteador, ao passo que a perda de pacotes manteve-se próxima a zero na abordagem híbrida, com $0,64 \pm 0,01\%$ para falha de enlace e $0,66 \pm 0,01\%$ para falha de roteador.



(a) Perda de Pacotes (b) Latência (RTT)

Figura 8. Métricas por Tipo de Falha com tráfego de fundo e topologia RNP.

Outra métrica analisada foi a latência (RTT) do tráfego na ocorrência de falhas. A Figura 8b representa a média de latência no tráfego com a recuperação de falhas em cada estratégia. Pode-se observar que a estratégia híbrida estabelece caminhos de recuperação da falha de forma a melhorar a latência, mantendo valores de latência de $147,50 \pm 0,64$ ms para falha de enlace e $146,67 \pm 0,61$ ms para falha de roteador, enquanto que as abordagens OSPF e ONOS causam impactos maiores na latência, com valores de $274,71 \pm 2,04$ ms e $183,92 \pm 5,23$ ms respectivamente para falha de enlace e $272,37 \pm 1,95$ ms e $176,40 \pm 13,29$ ms respectivamente para falha de roteador. Vale salientar que ambas as estratégias OSPF e ONOS não aplicam mecanismos de escolha do caminho ciente do QoS, no entanto o ONOS possui desempenho melhor devido ao fato de que o caminho BA-PE-CE-SP e BA-ES-RJ-SP possuem o mesmo peso e em alguns momentos, de forma não determinística, um deles é selecionado na recuperação.

Por fim, foi analisado o tempo de recuperação para verificar a melhoria proporcionada pelo uso de mecanismo de proteção pró-ativa baseada em *OpenFlow FF*. Essa métrica foi calculada de forma equivalente a trabalhos anteriores [Lin et al. 2016], onde são enviados pacotes de BA para SP a cada 10ms e, em SP, mede-se o intervalo de tempo entre o último pacote recebido antes da falha e o primeiro após a falha.

A Figura 9 apresenta uma comparação entre as estratégias Híbrida, OSPF e ONOS. É possível observar um valor elevado no OSPF, mesmo sendo configurado para

rápida convergência, com valores de $1,43 \pm 0,001$ seg para falha de enlace e $2,35 \pm 0,023$ seg para falha de roteador. Já as abordagens Híbrida e ONOS possuem tempos de recuperação abaixo de 200 ms, independente do tipo de falha. Vale salientar que no caso da abordagem ONOS, o tratamento de falha se dá no controlador e não localmente como na abordagem Híbrida que usa OpenFlow FF. Portanto, o tempo de recuperação da abordagem ONOS é função da latência na comunicação com o controlador e do tempo de processamento no controlador. No experimento em questão tais valores não foram refletidas pois o ambiente utilizava comunicação OpenFlow *out-of-band* e controlador local.

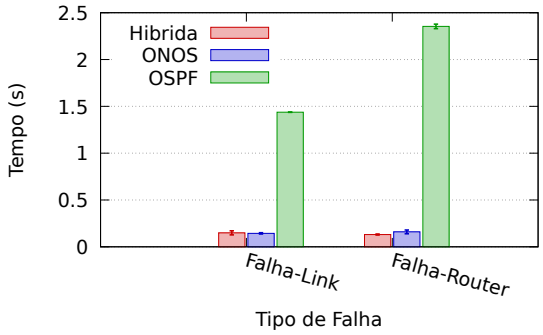


Figura 9. Tempo de Recuperação por Tipo de Falha.

6. Conclusões e Trabalhos Futuros

As estratégias convencionais de tolerância a falhas em redes de *backbone* geralmente reagem com base no estado configurado pelo administrador e não com base no histórico de desempenho da rede, o que pode causar congestionamento de enlaces ou impacto na QoS quando da ocorrência de falhas. Este trabalho apresentou uma heurística híbrida para tratamento de falhas em enlaces, ciente de QoS e sensível ao contexto. A heurística ora proposta trata localmente falhas a partir de um mecanismo proativo de proteção (OpenFlow FF) e, em seguida, de forma reativa, calcula melhores caminhos para recuperação de acordo com a política de QoS de cada aplicação, segundo o histórico de medições na rede e as classes de priorização de tráfego.

Foram identificadas vantagens quando comparado a algoritmos convencionais, especialmente em enlaces com diferentes características. A heurística obteve um tempo de recuperação superior a estratégias convencionais como OSPF, ao passo que sua execução local na fase proativa tende a fornecer melhores resultados que estratégias que dependem da comunicação com o controlador SDN, como o ONOS. Na fase reativa de recuperação, a partir da visão global do controlador, a heurística de cálculo do peso dos enlaces permite o estabelecimento de melhores caminhos, agregando novas vantagens com a otimização dos fluxos de encaminhamento sensíveis ao contexto. Com isso, importantes métricas de QoS são preservadas mediante a ocorrência de falhas e definição de novas rotas.

Em trabalhos futuros espera-se investigar os valores de "C" apropriados para cada topologia, além de avaliar múltiplas falhas de enlaces e outros modelos de falhas na rede. Outra possibilidade é avaliar o uso de políticas de QoS predefinidas para alguns tipos de tráfego, fazendo a análise concomitante de transferência de dados com diferentes características, e viabilizando preferência entre os tráfegos de rede distintos. Por fim, pretende-

se avaliar a utilização de outras abordagens para a determinação dos fatores de influência das métricas no peso do enlace, como abordagem de planejamento de capacidade.

7. Agradecimentos

Os autores agradecem o apoio financeiro da FAPESB, CAPES, CNPq e do MCTI/UFBA - Edital PROPESQ 004/2016.

Referências

Adrichem, N. L. M. v., Asten, B. J. v., and Kuipers, F. A. (2014). Fast recovery in software-defined networks. In *Proceedings of the 2014 Third European Workshop on Software Defined Networks*, EWSDN '14, pages 61–66, Washington, DC, USA. IEEE Computer Society.

Cascone, C., Sanvito, D., Pollini, L., Capone, A., and Sansò, B. (2017). Fast failure detection and recovery in sdn with stateful data plane. *Int. Journal of Network Management*, 27(2).

Chen, J., Chen, J., Xu, F., Yin, M., and Zhang, W. (2015). *When Software Defined Networks Meet Fault Tolerance: A Survey*, pages 351–368. Springer International Publishing, Cham.

Dusia, A. and Sethi, A. S. (2016). Recent advances in fault localization in computer networks. *IEEE Communications Surveys Tutorials*, 18(4):3030–3051.

Fonseca, P. and Mota, E. (2017). A survey on fault management in software-defined networks.

Jain, R. (1990). *The art of computer systems performance analysis: techniques for experimental design, measurement, simulation, and modeling*. John Wiley & Sons.

Karakus, M. and Durresi, A. (2017). Quality of service (qos) in software defined networking (sdn): A survey. *J. Network and Computer Applications*, 80:200–218.

Kreutz, D., Ramos, F. M. V., Veríssimo, P., Rothenberg, C. E., Azodolmolky, S., and Uhlig, S. (2014). Software-Defined Networking: A Comprehensive Survey. *Proceedings of the IEEE*, 103(1):63.

Lin, Y. D., Teng, H. Y., Hsu, C. R., Liao, C. C., and Lai, Y. C. (2016). Fast failover and switchover for link failures and congestion in software defined networks. In *2016 IEEE International Conference on Communications (ICC)*, pages 1–6.

Machado, C. C., Granville, L. Z., Filho, A. E. S., and Wickboldt, J. A. (2014). Towards sla policy refinement for qos management in software-defined networking. In Barolli, L., Li, K. F., Enokido, T., Xhafa, F., and Takizawa, M., editors, *AINA*, pages 397–404. IEEE Computer Society.

Sahri, N. M. and Okamura, K. (2014). Fast failover mechanism for software defined networking: Openflow based. In *Proceedings of The 9th International Conference on Future Internet Technologies*, CFI '14, pages 16:1–16:2, New York, NY, USA. ACM.

Stephens, B., Cox, A. L., and Rixner, S. (2016). Scalable multi-failure fast failover via forwarding table compression. In *Proceedings of the Symposium on SDN Research*, SOSR '16, pages 9:1–9:12, New York, NY, USA. ACM.