

RDNA: Arquitetura Definida por Resíduos para Redes de *Data Centers*

Alextian Bartholomeu Liberato¹, Magnos Martinello²,
Moisés Renato Nunes Ribeiro²

¹Instituto Federal de Educação, Ciência e Tecnologia do Espírito Santo (IFES) – Colatina/ES

²Universidade Federal do Espírito Santo (UFES) – Vitória/ES

alextian@ifes.edu.br, moises@ele.ufes.br, magnos@inf.ufes.br

Abstract. *Datacenter (DC) design has been moved towards the edge computing paradigm motivated by the need of bringing cloud resources closer to end users. However, the Software Defined Networking (SDN) architecture offers no clue to the design of Micro Datacenters (MDC) for meeting complex and stringent requirements from next generation 5G networks. This is because canonical SDN lacks a clear distinction between functional network parts, such as core and edge elements. Besides, there is no decoupling between the routing and the network policy. In the thesis, we introduce Residue Defined Networking Architecture (RDNA) as a new approach for enabling key features like ultra-reliable and low-latency communication in MDC networks. RDNA explores the programmability of Residues Number System (RNS) as a fundamental concept to define a minimalist forwarding model for core nodes. Instead of forwarding packets based on classical table lookup operations, core nodes are tableless switches that forward packets using merely remainder of the division (modulo) operations. By solving a residue congruence system representing a network topology, we found out the algorithms and their mathematical properties to design RDNA's routing system that (i) supports unicast and multicast communication, (ii) provides resilient routes with protection for the entire route, and (iii) is scalable for 2-tier Clos topologies. Experimental implementations on Mininet and NetFPGA SUME show that RDNA achieves 600 ns switching latency per hop with virtually no jitter at core nodes and sub-millisecond failure recovery time.*

Resumo. *O design do datacenter caminha rumo ao paradigma da computação de borda, motivado pela necessidade de aproximar os recursos da nuvem aos usuários finais. No entanto, a arquitetura SDN não oferece nenhuma orientação para o design de micro Datacenters suficiente para atender aos requisitos complexos e rigorosos das redes 5G. Isso ocorre porque a SDN canônica não possui uma distinção clara entre as partes funcionais desta rede, como elementos de núcleo e de borda. Além disso, não há desacoplamento entre o roteamento e a política de rede. Na tese, apresentamos a Arquitetura Definida por Resíduos (RDNA) como uma nova abordagem para habilitar recursos-chave como comunicação ultra-confiável e de baixa latência em redes DC. A RDNA explora a programabilidade do Sistema Numérico de Resíduos (RNS) como um conceito fundamental para definir um modelo de encaminhamento minimalista para os elementos de núcleo. Em vez de encaminhar pacotes com base*

em operações clássicas de pesquisa na tabela, os switches núcleo não possuem tabelas e encaminham os pacotes usando apenas o resto da operação de divisão (módulo). Resolvendo um sistema de congruência de resíduos representando uma topologia de rede, descobrimos os algoritmos e suas propriedades matemáticas para projetar o sistema de roteamento RDNA que: (i) suporta comunicação unicast e multicast, (ii) fornece rotas resilientes com proteção para toda a rota e (iii) é escalável para topologias Clos de 2 camadas. Implementações experimentais em Mininet e NetFPGA SUME mostram que o encaminhamento na RDNA alcança 600 ns de latência por salto com nenhum jitter nos elementos de núcleo e tempo de recuperação de falha em sub-milissegundos.

1. Caracterização do problema e motivação

Recentemente, *Data Centers* menores e mais próximos aos usuários, se proliferaram como uma alternativa para o armazenamento e processamento dos dados de maneira eficiente, além de conectar outros aplicativos e usuários que estão distribuídos globalmente. Esse novo modelo, chamado de *Edge Computing*, representa uma tendência com grandes benefícios em relação aos *Data Centers* tradicionais, tais como, menor latência e maior desempenho, redução de custo (CAPEX e OPEX) com uso de equipamentos baseado no padrão Ethernet, rápida implementação, facilidade de gerenciamento e orquestração, dentre outros [Dominicini et al. 2017]. A virtualização, a containerização e a Internet das Coisas, têm levado a um crescimento explosivo no número de pontos finais (*endpoints*) com uma demanda significativa por conectividade da nuvem [Hari et al. 2015].

Com o objetivo de reduzir a complexidade e flexibilizar o encaminhamento no núcleo das DCNs, diversas propostas defendem o uso do roteamento na origem. A parte comum entre estas propostas é a precomputação do caminho para cada destinatário com base em um rótulo (*tag*). Portanto, a indicação do próximo salto não é realizada pelo endereço do destinatário contido no cabeçalho, como nos algoritmos de roteamento tradicionais, mas sim com base em um índice que indica a porta de saída para cada pacote. O uso dos rótulos produz alguns benefícios, como por exemplo, maior facilidade de gerenciamento através da agregação de endereços em classe de serviços e menor tempo de busca nas tabelas. Porém, vai exigir o armazenamento dos estados (índices) nas tabelas de *switches* (*stateful*). Outra desvantagem é que a agregação utilizada no mapeamento dos rótulos se apresenta inadequada para políticas de gerenciamento do fluxo por estado, exigidas nos DCs para atender a crescente demanda por conectividade [Hari et al. 2015].

No que se refere às SDNs (*Software Defined Networks*) aplicadas nos DCs, o modelo de encaminhamento adotado pelos comutadores OpenFlow é baseado em *match-action* [Xia et al. 2015]. Resumidamente, nesse modelo um subconjunto de *Bytes* (cabeçalho) dos pacotes são buscados em uma tabela, as entradas correspondentes especificam ações que são aplicadas ao pacote determinando o seu encaminhamento. Dessa maneira, equipamentos tradicionais habilitados com OpenFlow fornecem sua tabela de busca (*lookup table*) como operação para comutação de pacotes. Portanto, a manutenção dos estados, o modelo de encaminhamento (*match-action*), e o tempo de busca na tabela, representam consideráveis pontos de variabilidade e incremento da latência de encaminhamento dos pacotes, principalmente pela limitação significativa do espaço da *Ternary Content Addressable Memory* (TCAM), o que exige o armazenamento dos estados

em outras memórias mais lentas [Martinello et al. 2017]. Ademais, os mecanismos de recuperação de falha da SDN tradicional requerem o armazenamento de estados adicionais nas tabelas [da Silva et al. 2015], o que gera perda de agilidade na recuperação de falhas, dificulta o processo de gerenciamento das DCNs.

A falta de mecanismos de reação rápida à falha dentro das DCNs é outra significativa limitação das propostas existentes. Aplicações em áreas críticas dependem de um alto nível de confiabilidade do dado transmitido, pois os mesmos podem ser produzidos apenas em um determinado instante e sua perda ou atraso na entrega compromete significativamente o serviço. Protocolos tradicionais de reação à falha em Camada 2, introduzem nas redes ineficiências substanciais como o bloqueio de alguns caminhos a fim de assegurar que não aconteçam *loops*, além de elevado tempo de convergência.

Além da baixa latência e resiliência dentro das DCNs, algumas aplicações requerem comunicação multicast, um exemplo são as aplicações para ambiente de monitoramento, economia de energia, automação, telecomunicações e segurança. Transmissões multicast beneficiam aplicações de grupo, logo devem reduzir o tráfego de rede, economizar custos de comunicação, melhorar a taxa de transferência, fornecendo assim uma infraestrutura eficiente para atender a crescente demanda por conectividade. Essas melhorias são significativas para pequenas redes *Cloud Center* [Jia 2014]. No entanto, as soluções tradicionais [Moy 1994, Pansiot 2010, Li and Freedman 2013], aplicadas nas DCNs não são eficazes, pois os *switches* podem não reconfigurar os grupos multicast IP nas taxas necessárias [Shahbaz et al. 2018].

Diante do cenário apresentado, observou-se que as tecnologias existentes aplicadas a *Data Centers* ainda careciam de melhorias significativas para atender as demandas por conectividade, principalmente para aplicações críticas que dependem de baixa latência e confiabilidade. Percebemos através de pesquisas na literatura e das atividades desenvolvidas pelo grupo de pesquisa (LabNerds), que o Teorema Chinês do Resto [Garner 1959], possui características relevantes na área de roteamento e encaminhamento, mantendo os *switches* de núcleo/intermediário sem tabela. Esta característica simplifica o gerenciamento das rotas nas DCNs, além de possivelmente reduzir o CAPEX com uso de equipamentos mais baratos por dispensarem o uso de memórias caras como as TCAMs. Também observamos que o TCR tem sido pouco explorado nas DCNs, tornando-se o principal motivador da tese.

2. Objetivos e contribuições

2.1. Objetivo geral:

Na tese foi proposto uma arquitetura para infraestrutura de rede destinada à *Data Centers*, capaz de suportar a comunicação unicast e multicast de forma eficiente, atendendo assim as demandas por conectividade para os serviços que requerem uma combinação de baixa latência de encaminhamento e resiliência dentro dos DCs. Além disso, esta nova arquitetura fornece latência determinística de encaminhamento mesmo sob condições de falhas transitórias, é escalável e compatível com o padrão Ethernet.

2.2. Objetivos específicos:

1. Demonstrar um sistema de roteamento unicast que auxiliem a redução da latência de encaminhamento e sua variabilidade nas DCNs baseado nos resíduos numéricos;

2. Formular um mecanismo de encaminhamento unicast que permita a rápida recuperação de falhas, seja em dispositivos ou em enlaces físicos, de forma proativa;
3. Desenvolver um sistema de roteamento para comunicação multicast que permita representar de forma eficiente a árvore de multicast sem a necessidade de armazenamento de estados nos *switches* de núcleo e sem modificar o pacote a cada salto.

O uso do Teorema Chinês do Resto (TCR) para suportar um sistema de roteamento centrado em resíduos não é novo, e já foi apresentada pelas propostas: Keyflow [Martinello et al. 2014], COXcast [Jia 2014] e KAR [Gomes et al. 2016]. Entretanto, no decorrer de toda revisão da literatura e ao longo de todo desenvolvimento da tese, percebemos que o TCR foi pouco explorado pelos muitos trabalhos relacionados à área de *Data Center*, especialmente na comunicação unicast considerando recuperação à falha e na comunicação multicast.

Neste contexto, destacamos duas contribuições principais da tese: i) projeto, implementação e validação de um mecanismo de proteção que permite reagir rapidamente à falhas ao longo de toda rota principal. Este mecanismo garante total proteção contra falhas em qualquer ponto da rota principal utilizando apenas um único identificador de rota emergencial, mantendo os *switches* de núcleo da topologia sem tabela; ii) projeto e análise de escalabilidade de um mecanismo de roteamento e encaminhamento para comunicação multicast. Nosso mecanismo constrói a árvore de multicast com base na codificação de um mapa de bits que permite representar o conjunto de portas no qual os pacotes devem ser transmitidos, sem reescrita de cabeçalho no pacote a cada salto e sem tabela de encaminhamento.

O mecanismo de recuperação apresentado na tese possui características únicas não encontradas na literatura, mesmo sem manter nenhum estado armazenado em tabela e sem depender de nenhum protocolo de convergência, graças a inovadora abordagem projetada, implementada e validada, cada pacote na comunicação unicast que entra na rede de núcleo transporta um identificador emergencial que permite a rápida reação à falha ao longo de toda rota principal. Portanto, a **Arquitetura Definida por Resíduos para Redes de Data Centers (RDNA)**, amplia o roteamento na origem para suportar uma abordagem resiliente determinística baseado em nosso mecanismo de proteção. O identificador de rota emergencial é calculado de forma a representar o conjunto de *switches* de núcleo e suas respectivas portas de saída. Para isso, nosso algoritmo de proteção considera rotas disjuntas e a menor distância em número de saltos a partir de cada *switch* de núcleo que compõe a rota principal. Os resultados demonstram que nosso mecanismo de reação à falha mantém a latência de encaminhamento determinística sob condição de não congestionamento das filas.

No que diz respeito ao roteamento e encaminhamento multicast, projetamos uma abordagem especial. Nosso esquema utiliza em conjunto duas subáreas da Teoria dos Números, o TCR e os polinômios algébricos. A RDNA monta a árvore de multicast a partir de um mapa de bits (*bitmap*) que representa o conjunto de portas em cada *switch* que faz parte da rota multicast. Um simples algoritmo calcula o grau do polinômio com base nos identificadores localizados nos *switches* de núcleo e na quantidade total de portas. A partir do *bitmap*, um segundo algoritmo calcula os coeficientes utilizados na expressão polinomial específica. Esta transformadora abordagem mantém-se integrada com os benefícios

da comunicação unicast apresentadas nesta tese. Ademais, a abordagem RDNA para comunicação multicast é consideravelmente mais escalável que as concorrentes, reduzindo em até 50% o tamanho do cabeçalho multicast para algumas das topologias avaliadas. Em síntese, a arquitetura RDNA oferece consideráveis avanços em termos de redução de latência no mecanismo de encaminhamento unicast. Do mesmo modo, a RDNA fornece um eficiente mecanismo de roteamento e encaminhamento multicast. Além disso, também consideramos a implementação e validação da RDNA como uma valiosa contribuição. No melhor conhecimento deste autor, essa foi a primeira implementação de uma proposta SDN que de fato usa uma rede de núcleo *fabric* em netFPGA SUME 10G para redes de *Data Center*, utilizando o TCR como sistema de roteamento e encaminhamento, incluindo reação rápida à falhas na rede de núcleo. Todas as propostas anteriores de rede de núcleo eram puramente conceituais [Casado et al. 2012, Hari et al. 2015].

3. Trabalhos relacionados

Com base na característica da pesquisa, optou-se em dividir o referencial teórico em três categorias, descritas resumidamente a seguir. São elas:

Encaminhamento de pacotes baseado em rótulos

Na literatura encontramos uma numerosa quantidade de propostas e arquiteturas que fazem uso do roteamento por segmento. Também é comum encontrar nesses trabalhos o uso de mecanismos de encaminhamento baseado em rótulos ou etiquetas, incluindo: MPLS [Rosen et al. 2001], VxLANs [Mahalingam et al. 2013], Segment Routing [Filsfils et al. 2014], dentre outros discutidos na tese.

Nossa arquitetura, ao contrário das propostas existentes, elimina completamente qualquer necessidade de armazenamento de estados nos *switches* de núcleo, além disso, nenhuma reescrita de pacote é executada sob condições normais da rede. Portanto, nosso mecanismo de encaminhamento no núcleo simplifica o gerenciamento e a escalabilidade da rede.

Roteamento Resiliente

A adoção do padrão Ethernet pela indústria causou a criação de grandes domínios em Camada 2 que exigem convergência rápida. Todavia, em nossa pesquisa, observamos que independente do protocolo de Camada 2 utilizado, um desafio comum em realizar resiliência eficiente, ou seja, no menor tempo possível, está na troca de mensagens de controle dos protocolos para prevenção de *loops* e inconsistências.

Nosso mecanismo de reação a falhas rápidas usa um identificador de rota de emergencial (IRE). O IRE é calculado de forma a representar um conjunto de *switches* necessários para contornar uma falha na rota principal. Este recurso é importante e desejável em redes de *Data Center*, principalmente pelo alto grau de conectividade comumente encontrado nas topologias.

Comunicação Multicast

Cabe aos algoritmos de roteamento e técnicas de encaminhamento de pacotes utilizadas na comunicação multicast apoiar os padrões um-para-muitos, muitos-para-um e muitos-para-muitos. Os operadores de DCNs podem usar multicast IP nativo [Li and Freedman 2013]. No entanto, essa solução não é eficaz porque os *switches* podem não reconfigurar grupos multicast IP nas taxas necessárias [Shahbaz et al. 2018].

Nossa arquitetura de roteamento multicast foi projetada considerando as seguintes premissas: i) não ser necessário armazenar estado nos *switches* de núcleo, logo, o projeto do *hardware* pode ser simplificado, além de manter a compatibilidade com nosso mecanismo de recuperação ultra-rápida no núcleo da rede; ii) visando a redução da latência de comutação do pacote, não existe reescrita do pacote a cada salto. Portanto, com essas duas premissas buscamos uma solução mais eficiente de transmissão multicast em redes de *Data Center*, além de ser escalável, conforme apresentamos em nossa análise de escalabilidade.

4. Resultados obtidos

Como resultado obtidos na tese, destacamos, a eficiência da RDNA, quando comparada com algumas propostas concorrentes, requerendo um comprimento de cabeçalho menor. Outra vantagem da RDNA foi manter o *frame* Ethernet inalterado, considerando diferentes arranjos topológicos. No padrão de comunicação multicast, analisamos a variação de tamanho da rede (quantidade de *switches* na árvore multicast), e o impacto no cabeçalho variando a quantidade de membros por grupo multicast. Também neste tipo de comunicação a RDNA se mostrou eficaz, reduzindo em até 50% o tamanho do cabeçalho multicast para alguns dos casos avaliados.

Descrevemos em detalhes a implementação da arquitetura RDNA. Para demonstrar a viabilidade apresentamos o projeto e a implementação de dois diferentes protótipos: i) ambiente emulado; e ii) testbed 10 Gbps em NetFPGA SUME. A RDNA atinge baixa latência de comutação por salto ($\approx 0.6 \mu s$) e nenhuma variação de latência no encaminhamento dentro do núcleo da rede. Além disso, nossa arquitetura oferece proteção ao longo de toda rota principal com recuperação de falha ultra-rápida (sub-milissegundos). Este resultado é extremamente desejado para serviços e aplicações de missão crítica de forma geral.

Comparamos dois projetos para arquitetura de *switches*, com diferentes métodos de comutação. O método de comutação de pacotes *cut-through switching* não requer o armazenamento completo do *frame*, portanto, após receber e processar as informações do cabeçalho necessárias para comutação, o pacote já pode ser enviado para a fila da porta de saída. Já o método *store-and-forward switching* vai requerer no mínimo o armazenamento de todo *frame* em um *buffer* temporário. Ao contrário a arquitetura SDN OpenFlow tradicional, a RDNA não requer nenhum estado armazenado em tabelas, o que permitirá desenvolver no futuro um dispositivo mais barato para DCNs, além de consumir menos energia. Ademais, a RDNA suporta a implementação do método de comutação *cut-through*, que reduz a latência de encaminhamento, conforme demonstramos no protótipo implementado nas NetFPGAs. Para finalizar, destacamos que ao tornar o tempo de *lookup* determinístico, abrimos uma nova linha de investigação de políticas para engenharia de tráfego que podem levar a serviços determinísticos, entretanto, esta linha de investigação não foi analisada por estar fora do escopo desta tese e serão investigados nos trabalhos futuros.

5. Subprodutos do trabalho

As contribuições dos últimos 4 (quatro) anos, diretamente relacionadas com a arquitetura RDNA são listadas em ordem cronológica inversa através das seguintes publicações:

1. **LIBERATO, A. B.**; MARTINELLO, M.; GOMES, R. L.; BELDACHI, A. F.; HUGUES-SALAS, E.; VILLACA, R.; RIBEIRO, M. R. N.; KANELLOS, G.; NEJABATI, R.; GORODNIK, A.; SIMEONIDOU, D. RDNA: Residue-Defined Networking Architecture Enabling Ultra-Reliable Low-Latency Datacenters. *IEEE Transactions on Network and Service Management* (Volume: 15, Issue: 4, Dec. 2018. DOI: 10.1109/TNSM.2018.2876845).
2. MARTINELLO, M.; **LIBERATO, A. B.**; BELDACHI, A. F.; KONDEPU, K.; GOMES, Roberta L.; VILLACA, R.; RIBEIRO, M. R. N.; YAN, Y.; HUGUES-SALAS, E.; SIMEONIDOU, D. Programmable Residues Defined Networks for Edge Data Centres. 13th International Conference on Network and Service Management (CNSM, 2017), Tokyo, Japan, Nov. 26-30, 2017.
3. GOMES, R. R.; DOMINICINI, C. K.; **LIBERATO, A. B.**; RIBEIRO, M. R. N.; MARTINELLO, M. Analytical Modeling Approach of Routing Deflection for Intra-domain Networks In: XXXVI Congresso da Sociedade Brasileira de Computação, 2016, Porto Alegre/RS. XV Workshop em Desempenho de Sistemas Computacionais e de Comunicação, 2016.
4. **LIBERATO, A. B.**; MARTINELLO, M.; RIBEIRO, M. R. N.; MARQUEZ-BARJA, J. M.; KAMINSKI, N.; DASILVA, L. A. Dynamic Backhauling within Converged Networks In: ACM SIGCOMM, 2016, Florianópolis. ACM SIGCOMM Workshop on Fostering Latin-American Research in Data Communication Networks (LANCOMM), 2016.
5. GOMES, R. R.; **LIBERATO, A. B.**; DOMINICINI, C. K.; RIBEIRO, M. R. N.; MARTINELLO, M. KAR: Key-for-Any-Route, a Resilient Routing System In: The 46th IEEE/IFIP International Conference on Dependable Systems and Networks (DSN'2016), 2016, Toulouse-France. The 2nd Workshop on Dependability Issues on SDN and NFV (DISN), 2016.
6. LIMA, D. S. A.; GUIMARAES, R.; **LIBERATO, A. B.**; SPALLA, E. S.; VASSOLER, G. L.; MARTINELLO, M.; VILLACA, R. REUNI: Um algoritmo para REduzir tabelas de encaminhamento e UNiformizar a distribuição dos fluxos em redes com topologia hipercubo In: XXXIV Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC), 2016, Salvador-Bahia. XXXIV Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC), 2016.
7. LIMA, D. S. A.; GUIMARAES, R.; VASSOLER, G. L.; MARTINELLO, M.; VILLACA, R.; **LIBERATO, A. B.** Avaliação do Uso do OpenFlow na Recuperação de Falhas em Data Centers Centrados nos Servidores In: XIV Workshop em Desempenho de Sistemas Computacionais e de Comunicação (WPerformance 2015), 2015, Pernambuco-Recife. XXXIV Congresso da Sociedade Brasileira de Computação – CSBC, 2015.

Referências

- Casado, M., Koponen, T., Shenker, S., and Tootoonchian, A. (2012). Fabric: A retrospective on evolving sdn. In *Proceedings of the First Workshop on Hot Topics in Software Defined Networks*, HotSDN '12, pages 85–90, New York, NY, USA. ACM.
- da Silva, A. S., Smith, P., Mauthe, A., and Schaeffer-Filho, A. (2015). Resilience support in software-defined networking. *Comput. Netw.*, 92(P1):189–207.

- Dominicini, C. K., Vassoler, G. L., Meneses, L. F., Villaca, R. S., Ribeiro, M. R. N., and Martinello, M. (2017). Virtphy: Fully programmable nfv orchestration architecture for edge data centers. *IEEE Transactions on Network and Service Management*, 14(4):817–830.
- Filsfils, E. C., Previdi, E. S., Systems, I. C., Decraene, B., Litkowski, S., Orange, Shkir, R., and Communications, J. (2014). Segment Routing Architecture. Internet-Draft Segment Routing Architecture draft-ietf-spring-segment-routing-00, Network Working Group. Standards Track.
- Garner, H. L. (1959). The residue number system. *Transactions on Electronic Computers*, pages 140 – 147.
- Gomes, R. R., Liberato, A. B., Dominicini, C. K., Ribeiro, M. R. N., and Martinello, M. (2016). Kar: Key-for-any-route, a resilient routing system. In *2016 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshop (DSN-W)*, pages 120–127.
- Hari, A., Lakshman, T. V., and Wilfong, G. (2015). Path switching: Reduced-state flow handling in sdn using path information. In *Proceedings of the 11th ACM Conference on Emerging Networking Experiments and Technologies, CoNEXT '15*, pages 36:1–36:7, New York, NY, USA. ACM.
- Jia, W. K. (2014). A scalable multicast source routing architecture for data center networks. *IEEE Journal on Selected Areas in Communications*, 32(1):116–123.
- Li, X. and Freedman, M. J. (2013). Scaling ip multicast on datacenter topologies. In *Proceedings of the Ninth ACM Conference on Emerging Networking Experiments and Technologies, CoNEXT '13*, pages 61–72, New York, NY, USA. ACM.
- Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., and Wright, C. (2013). VXLAN: A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks. Internet Draft.
- Martinello, M., Liberato, A. B., Beldachi, A. F., Kondepu, K., Gomes, R. L., Villaca, R., Ribeiro, M. R. N., Yan, Y., Hugues-Salas, E., and Simeonidou, D. (2017). Programmable residues defined networks for edge data centres. In *2017 13th International Conference on Network and Service Management (CNSM)*, pages 1–9.
- Martinello, M., Ribeiro, M. R. N., De Oliveira, R. E. Z., and De Angelis Vitoi, R. (2014). Keyflow: A prototype for evolving SDN toward core network fabrics. *IEEE Network*, 28(2):12–19.
- Moy, J. (1994). Multicast extensions to ospf. RFC 1584, <https://tools.ietf.org/html/rfc1584>.
- Pansiot, J.-J. (2010). *Multicast Routing on the Internet*. ISTE.
- Rosen, E., Viswanathan, A., and Callon, R. (2001). RFC 3031: Multiprotocol Label Switching Architecture. Technical report, IETF.
- Shahbaz, M., Suresh, L., Feamster, N., Rexford, J., Rottenstreich, O., and Hira, M. (2018). Elmo: Source-Routed Multicast for Cloud Services. *ArXiv e-prints*.
- Xia, W., Wen, Y., Foh, C. H., Niyato, D., and Xie, H. (2015). A survey on software-defined networking. *IEEE Communications Surveys Tutorials*, 17(1):27–51.