

Uma abordagem para redução da tabela de encaminhamento sob a ótica da interface de saída dos pacotes.

Luiz Fernando T. de Farias¹, Morganna C. Diniz¹, Sidney C. de Lucena¹.

¹Universidade Federal do Estado do Rio de Janeiro (UNIRIO),
Rio de Janeiro – Brasil

{luiz.farias, morganna, sidney}@uniriotec.br

Abstract. *The progressive increase of the global routing table is an obstacle for the internet growth. Besides, the use of virtualization in datacenter networks and the arise of software-defined networking (SDN) intensified the demand for processing and storage resources, increasing even more the cost of router platforms. This work proposes an algorithm that reduces the forwarding table, or forwarding information base (FIB), from calculations based on the routing table, or routing information base (RIB), from the perspective of the outgoing interface for a given route. This approach allows the reduction of the FIB according to the used network topology. Experiments with the proposed algorithm were done using the BIRD routing software over Linux LXC virtual machines. Measures related to the forwarding table size and network latency were obtained and analysed for a scenario representing two Internet providers announcing network prefixes with the BGP protocol.*

Resumo. *O aumento da tabela de roteamento global é um obstáculo para o crescimento da Internet. Além disso, o uso da virtualização em datacenter e as Redes Definidas por Software (SDN) intensificaram a demanda por recursos de processamento e armazenamento, aumentando ainda mais os custos das plataformas de roteamento. Este trabalho propõe um algoritmo que reduz o tamanho da tabela de encaminhamento (FIB) a partir de cálculos realizados com base na tabela de roteamento (RIB) sob a ótica da interface de saída para uma dada rota. Esta abordagem possibilita a redução da FIB de acordo com a topologia da rede empregada. Experimentos com o algoritmo proposto foram realizados com o software de roteamento BIRD em máquinas virtuais LXC no kernel do Linux. Medidas relativas ao tamanho da tabela de encaminhamento e à latência na rede foram obtidas e analisadas para um cenário que representa dois provedores de serviço de Internet anunciando prefixos com o protocolo BGP.*

1. Introdução

Protocolos de roteamento como o BGP (*Border Gateway Protocol*) ou OSPF (*Open Shortest Path First*) são usados para descobrir o caminho a ser percorrido pelos pacotes da origem até o destino. O resultado é armazenado na base de informação de roteamento (RIB, *Routing Information Base*). Nesta tabela são registrados todos os possíveis caminhos até os prefixos de rede aprendidos por todos os protocolos de roteamento que estão em operação. Na base de informação de encaminhamento (FIB, *Forwarding Information Base*) estão registrados os prefixos da rede de destino e o endereço IP do próximo roteador (*next hop*), já sanados quaisquer conflitos. A diferença entre o conteúdo da RIB e

da FIB está na simplicidade necessária para se obter uma melhor performance no envio das informações. A RIB possui o registro de todos os detalhes das rotas enquanto a FIB possui o necessário para encaminhar o pacote até o próximo salto.

Para criar um quadro que levará o pacote até a próxima interface (*framing*) é necessário que o endereço físico (HW-ADDR, *hardware address*) do próximo roteador seja conhecido. Esta é a função do protocolo de endereçamento ARP (*Address Resolution Protocol*) que associa os endereços IP com os endereços dos dispositivos físicos. Os quadros que chegam na interface de entrada (RX) são processados e enviados para a saída (TX) de acordo com o *cache* das tabelas FIB e ARP na memória (MEM) do dispositivo. O protocolo de roteamento e a RIB são elementos do plano de controle, enquanto a FIB e a tabela ARP são elementos do plano de dados [Mohamed et al. 2012]. A ligação entre esses elementos é observada na Figura 1.

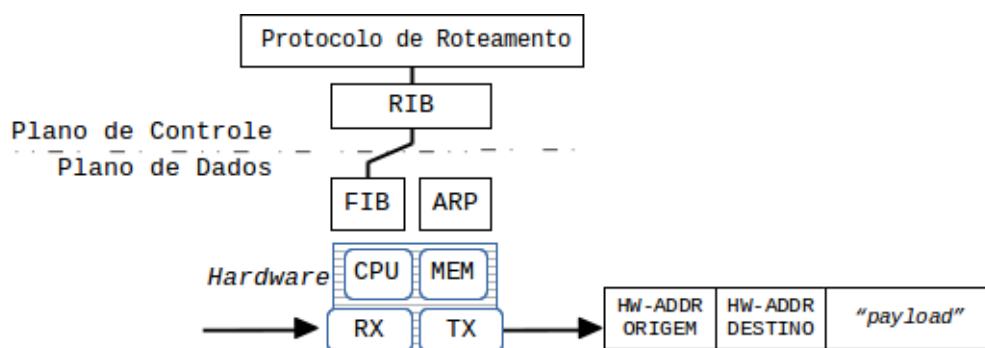


Figura 1. Separação de elementos usados no encaminhamento dos pacotes.

Uma topologia de rede com vários roteadores pode ser obtida em um único computador através de virtualização. A virtualização possibilita o compartilhamento dos recursos computacionais, incluindo o plano de dados, entre os diversos elementos virtuais. No ambiente virtual em que suítes de roteamento (SR) são executadas em diferentes máquinas virtuais (MVs), as configurações de rede, as RIBs e a FIBs são isoladas e independentes entre si. Por exemplo, uma topologia formada com cem roteadores foi criada sobre um único *kernel* Linux nos experimentos executados em [de Farias et al. 2013]. Neste cenário, o enlace entre as SRs foi obtido com uma ponte Linux e as FIBs compartilham a mesma memória física. No caso de ser estabelecida uma conectividade com uma rede física, a memória do dispositivo teria que ser suficiente para acomodar as tabelas de encaminhamento resultantes das rotas aprendidas desta rede física.

Uma situação parecida ocorre em *data centers* quando o recurso da virtualização de redes é usado. Um estudo sobre demandas de hardware que suportem o aumento da FIB com as exigências da virtualização é apresentada em [Rottenstreich et al. 2013]. Problema similar seria observado nas redes definidas por *software* com o uso da *sobreposição de redes*. Nesta técnica, um dispositivo pode ser programado de maneira a participar de diferentes topologias simultaneamente, aumentando o tamanho da FIB [Corrêa et al. 2012].

A exigência de maior eficiência torna este aumento da FIB um obstáculo para o crescimento da Internet [Meyer et al. 2007] e a necessidade de soluções torna-se prioritária [Sánchez 2013]. A consulta do próximo salto retarda os roteadores quando realizam o envio dos dados, enquanto o tempo para atualização dos prefixos influencia

a convergência após falhas, ou seja, a atualização da FIB após modificações na RIB [Mohamed et al. 2012]. Em [Bonaventure et al. 2007] foi calculado que 40 a 55% do tempo total de convergência em provedores de serviços (ISP, *Internet Service Provider*) é consumido na atualização das FIBs, e seu tamanho influencia diretamente neste tempo de atualização. O tamanho das FIBs na eficiência das redes privadas virtuais (VPN - *Virtual Private Network*) de larga escala é observado em [Mai and Du 2013].

O aumento da tabela de roteamento global é causado pela forma como os prefixos de rede são administrados [Kumar and Kumar 2013] e na demanda por novos endereços IP. Esta situação é mais evidente na região sem rota padrão (DFZ, *Default Free Zone*) empregada no núcleo da Internet [Berkowitz et al. 2005]. Roteadores usados nos ISPs nível 1 (*TIER-1*) não configuram a rota padrão (0.0.0.0/0 no IPv4 e ::/0 no IPv6), ou seja, as RIBs nestes roteadores possuem rotas definidas para todos os prefixos de rede da Internet. Estas tabelas são denominadas DFRT (*Default Free Routing Table*) e, atualmente, possuem cerca de 500 mil prefixos [Huston 2013].

Este trabalho apresenta um algoritmo que demonstra que a FIB pode ser reduzida se os prefixos informados pela RIB forem sumarizados de acordo com a interface de rede associada. A ideia tem como base a observação de que o próximo salto representa um grupo de prefixos de rede associados ao próximo salto. Dessa forma, a utilização de uma única entrada que englobe grupos de endereços na FIB não interfere no encaminhamento do pacote na direção correta desde que os endereços sejam exclusivos para cada interface de rede. No algoritmo proposto, os prefixos informados pela RIB são separados de acordo com a interface de encaminhamento e depois gravados na FIB.

Portanto, esta proposta contribui com uma solução de redução da FIB sob a ótica da interface da rede. Dessa forma, além de se adequar com diferentes tipos de endereçamento lógico (por exemplo o IPv6), possibilita-se a utilização simultânea com outros mecanismos que também são usados para reduzir o tamanho da FIB, como o S-VA (*Simple Virtual Aggregation*) [Ballani et al. 2009] e o CIDR (*Classless Inter-Domain Routing*) [Fuller and Li 2006]. Nos experimentos realizadas métricas associadas com o tamanho da FIB e latência foram coletadas em um cenário formado por dois provedores de serviço anunciando prefixos com o protocolo BGP.

O restante deste artigo é organizado da seguinte forma: na Seção 2 são apresentadas outras propostas e técnicas empregadas para redução da FIB; na Seção 3 são apresentadas a fundamentação técnica, a descrição do algoritmo implementado e o cenário empregado para coletar as métricas usadas na prova do conceito; a análise dos resultados está na Seção 4 e as conclusões e propostas para trabalhos futuros na Seção 5.

2. Trabalhos anteriores

Para redução da FIB, pode-se usar filtragem ou agregação. Na filtragem, um grupo de prefixos é selecionado conforme caminhos semelhantes pela topologia da rede, e busca-se uma forma reduzida para representar este grupo. Na agregação, uma estrutura de dados particular é usada para compactar a informação dos prefixos que partilham caminhos semelhantes.

No FIFA (*Fast Incremental FIB Aggregation*) três algoritmos são usados para obter agregação que proporcione uma FIB com maior taxa de redução (FIFA-S), com melhor

tempo de consulta (FIFA-T) ou com ambos simultaneamente (FIFA-H) [Liu et al. 2013]. Este trabalho utiliza o ORTC (*Optimal Routing Table Constructor*) [Draves et al. 1999] que adota o módulo Patricia (*Practical Algorithm to Retrieve Information Coded in Alpha-numeric*) [Morrison 1968] para fazer pesquisas dos endereços IPs estruturados em árvore. O FIFA obtém melhor desempenho comparado ao SMALTA (*Saving Memory And Lookup Time via Aggregation*) que utiliza buscas na árvore binária, e por isso demanda maior carga de processamento pois realiza buscas em todo os nós [Uzmi et al. 2011].

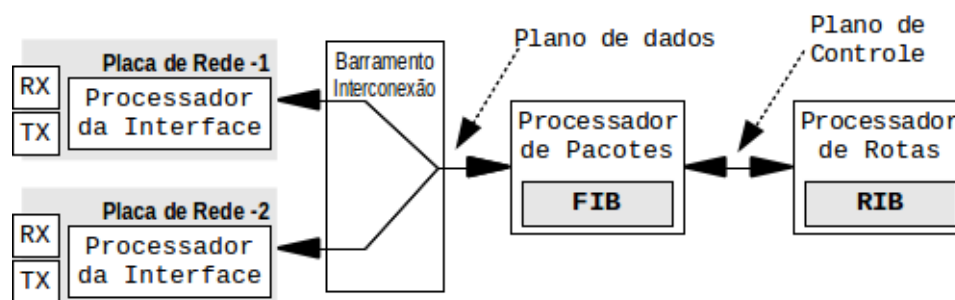


Figura 2. Arquitetura de roteadores com FIB centralizada.

[Mohamed et al. 2012] propõe que a redução no tempo de convergência em roteadores com arquitetura distribuída pode ser obtida com a filtragem na FIB. Neste tipo de equipamento a RIB é copiada para todas as FIBs que são armazenadas nas interfaces, por isso o algoritmo faz a redução evitando que prefixos desnecessários sejam registrados. No trabalho são descritos dois tipos de arquiteturas usadas na DFZ: roteadores com encaminhamento centralizado e roteadores com encaminhamento distribuído.

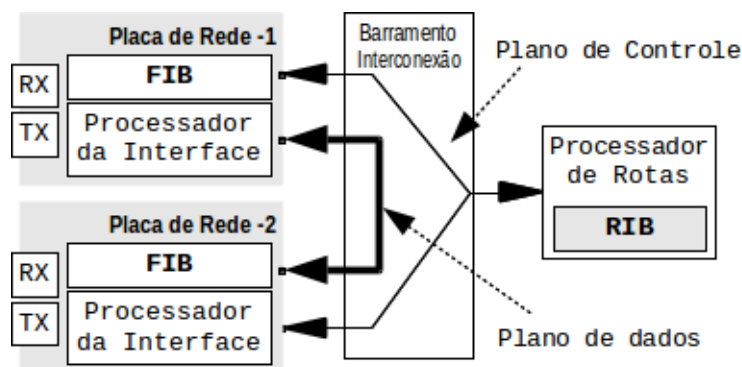


Figura 3. Arquitetura de roteadores com FIB distribuída.

Na arquitetura centralizada (Figura 2) o barramento de interconexão é usado para a transferência dos quadros até o processador de pacotes que é compartilhado com todas as outras interfaces. Na arquitetura distribuída (Figura 3) o barramento de interconexão é usado para transferência dos quadros diretamente entre as interfaces (plano de dados) e para obter atualizações da RIB (plano de controle). A decisão de encaminhamento é realizada de forma independente, pois cada interface possui uma FIB própria.

A RFC-6769 [Raszuk et al. 2012] descreve a metodologia de agregação virtual S-VA (*Simple Virtual Aggregation*) como método para ser empregado na DFZ. De acordo

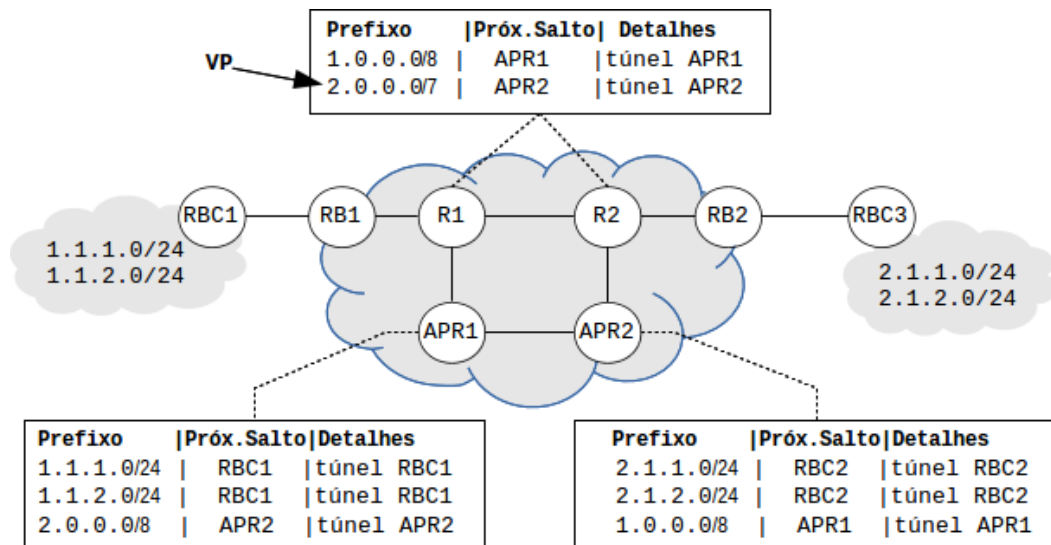


Figura 4. S-VA: Simple Virtual Aggregation.

com [Ballani et al. 2009], ISPs podem configurar roteadores para manter a RIB completa enquanto os prefixos armazenados na FIB são distribuídos em uma topologia virtual. O roteador ponto de agregação (APR, *Aggregation Point Router*) possui na FIB todos os prefixos de um agregado e os endereços dos próximos saltos que deverão ser usado pelo pacote (Figura 4). A técnica consiste na utilização do MPLS (*Multi Protocol Label Switching*) para conectar os roteadores participantes através de túneis. Quando um pacote entra na borda do AS ele é encapsulado e comutado até o APR.

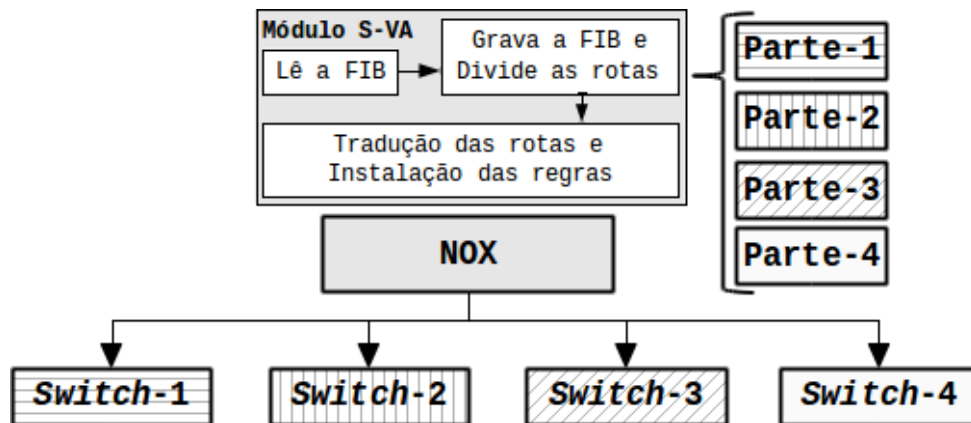


Figura 5. Simple Virtual Aggregation em SDN.

No contexto de SDN, o trabalho de [Sánchez 2013] demonstra a viabilidade da implementação do S-VA em redes OpenFlow. Conforme descrito pelo autor, o uso de OpenFlow reduz a complexidade da implementação, pois a coleta da RIB e a distribuição das regras são executadas a partir de um módulo central. O aplicativo utiliza um controlador OpenFlow (por exemplo o NOX) para executar as tarefas de ler e gravar na FIB, otimizar a divisão das regras e seu envio para os *switches* OpenFlow (Figura 5).

Este trabalho se diferencia dos anteriores nos seguintes aspectos:

1. A ideia primária usa uma técnica de filtragem;
2. Após o esquema de filtragem proposto, a ordenação dos prefixos “substitui” a necessidade de busca em estruturas de dados usadas para agregação, o que não impede delas serem futuramente usadas;
3. Selecionamos prefixos a partir da RIB, permitindo estratégias diversas, tais como (i) usar rotas que não sejam *best* para balancear a carga, (ii) ou para reduzir ainda mais a FIB, (iii) ou para ações relacionadas a roteamento virtualizado em redes SDN/OpenFlow.

3. Metodologia e implementação

A confecção da FIB para diferentes cenários de rede pode ser obtida mantendo-se o foco na interface conectada ao enlace que leva ao próximo salto (*next-hop*). Desta forma, procura-se criar agrupamentos de redes de destino cujos endereços de próximo salto estejam conectados através de um mesmo enlace (ou interface de rede). A ideia consiste em se obter o menor número possível de agrupamentos e de representar unicamente cada agrupamento por um mesmo prefixo de rede.

Esta abordagem considera dois aspectos importantes de ordem prática: (i) que a distribuição geográfica das faixas de IP pelo mundo gera situações nas quais diversos blocos de rede, anunciados por diferentes AS de uma mesma região, possuam faixas de IP próximas umas das outras [Alderson et al. 2005]; (ii) que a interligação entre roteadores se dá através de enlaces ponto-a-ponto. Este último aspecto será considerado no algoritmo proposto de maneira a aumentar no nível de redução da FIB, o que se torna possível quando não se faz necessário considerar o endereço de próximo salto das rotas, apenas a interface de saída que conduz a este próximo salto (*next-hop*). Entretanto, o algoritmo também pode ser generalizado para contemplar o *next-hop*, conforme será visto a seguir.

3.1. Algoritmo proposto

O algoritmo proposto configura a FIB usando os registros da RIB, mantidos pela suíte de roteamento, sempre que a mesma é atualizada (Figura 6). Para isto, duas informações são necessárias: o prefixo da rede de destino e a interface usada para enviar os pacotes para o próximo salto da rota. A redução da FIB é obtida calculando-se um novo prefixo para representar exclusivamente um agrupamento de endereços. Este prefixo funciona como um filtro para apontar a interface de saída que deve ser usada para atingir os prefixos de destino contidos neste agrupamento.

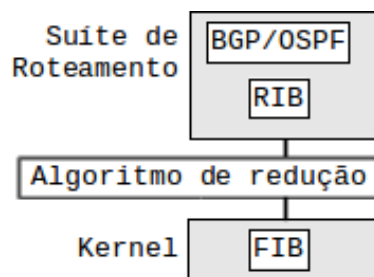


Figura 6. Ligação entre elementos de rede e o algoritmo

A primeira etapa é a obtenção dos campos DESTINO, INTERFACE e DETALHES da RIB, conforme Figura 7a. A partir destes campos é gerada uma lista ordenada

de endereços de rede, com suas respectivas interfaces de saída 7b. Estes endereços são obtidos com a remoção da máscara usada no prefixo de destino e as interfaces de saída são obtidas a partir da melhor rota assinalada no campo DETALHES da RIB(Figura 7a).

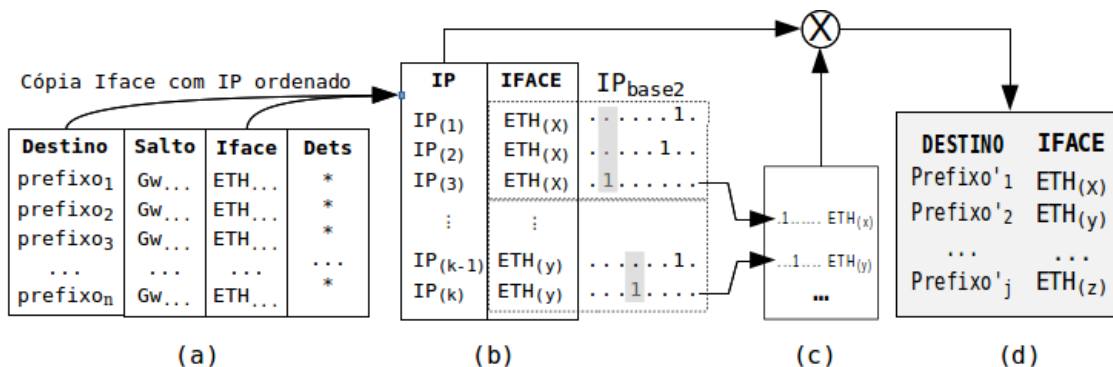


Figura 7. Etapas: (a) RIB (B) Ordenação e agrupamento (C) Filtragem e (D) FIB.

Na segunda etapa do algoritmo, os endereços de rede são inspecionados para se verificar a posição do primeiro bit 1 de cada endereço. Os endereços de rede que possuam este primeiro bit 1 na mesma posição inicialmente farão parte de um mesmo grupo 7c. Estes grupos, após formados, são cada qual subdivididos de acordo com as interfaces de saída existentes para as redes neles contidas.

Na terceira etapa, obtém-se prefixos de rede que representem unicamente cada subdivisão obtida na etapa anterior. Tais prefixos são obtidos observando-se a ocorrência do bit 1 mais significativo de cada subdivisão e sua respectiva posição. Será então definido um prefixo no formato IP/máscara que represente um número IP contendo apenas este bit 1.

Caso não haja nenhuma ambiguidade, a redução da FIB se encerra na terceira etapa. Entretanto, uma quarta etapa ocorrerá caso existam prefixos iguais associados a interfaces diferentes. Neste caso, haverá um refinamento do processo de geração de prefixos. Os endereços de rede menores associados ao prefixo em questão serão novamente inspecionados de maneira a se obter uma máscara de rede maior que represente bits em comum dentro da mesma subdivisão. Este novo prefixo, derivado deste processo de refinamento, terá maior prioridade de encaminhamento com relação ao prefixo original. Esta situação é análoga à regra de *longest prefix-match*. Esta etapa se repete até que todas as situações de ambiguidade sejam resolvidas. Uma vez finalizada, torna-se possível gerar a tabela da Figura 7d, que representa a FIB reduzida.

Um exemplo do algoritmo é apresentado nas Figuras 8 e 9. A Figura 8 apresenta a etapa 2 do algoritmo e a Figura 9 mostra as etapas 3 e 4. Por motivo de simplicidade, os IP que representam os endereços de rede contidos na RIB possuem apenas o primeiro octeto diferente de zero. Ao final da etapa 2, verificamos duas situações de ambiguidade relativas aos prefixos 64.0.0.0/2 e 128.0.0.0/1. Estas entradas estão sinalizadas pela sigla NOK na primeira tabela da esquerda para direita na Figura 9. Tais situações geram processos de refinamento dos respectivos prefixos apresentados na tabela imediatamente à direita.

Para o prefixo 64/2, seleciona-se a subdivisão com menor número de prefixos e aumenta-se a máscara para os respectivos prefixos rede (no caso, apenas um) de maneira

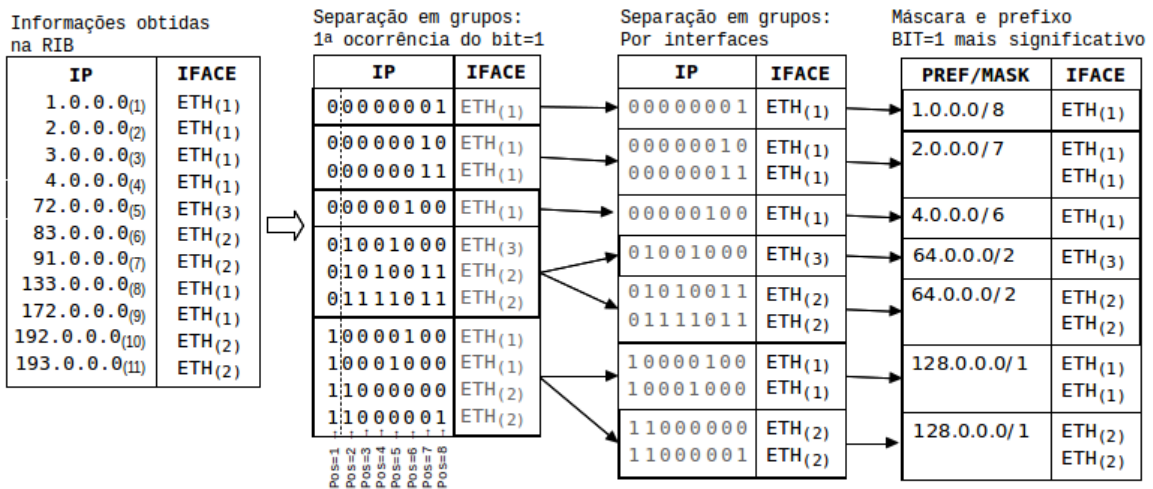


Figura 8. Etapa 2 do Algoritmo.

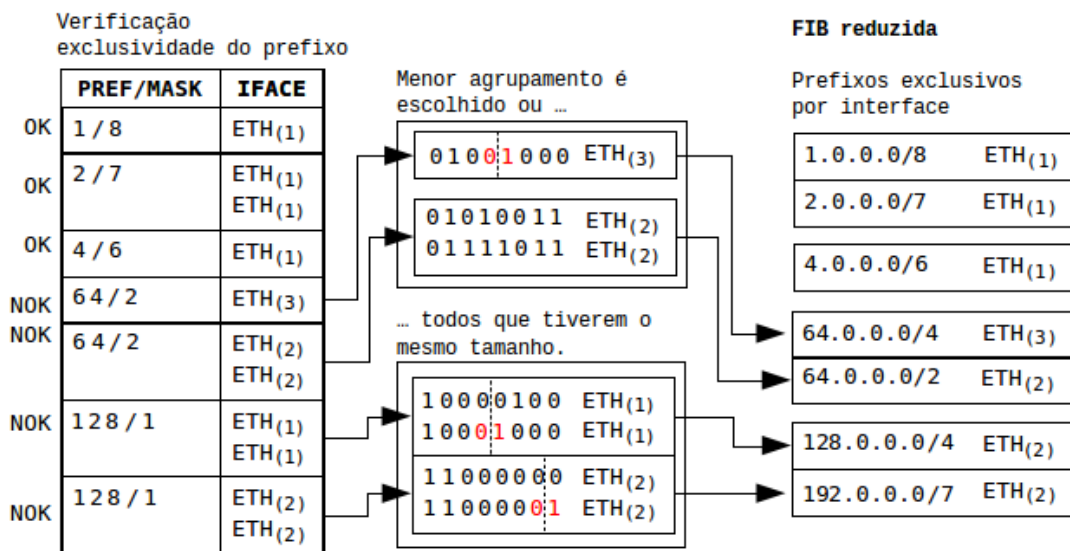


Figura 9. Etapa 3 e 4 do Algoritmo.

a diferenciar esta subdivisão da posterior. Desta forma, a máscara irá até a posição 4 resultando no prefixo 64/4. O outro agrupamento ficará inalterado, sendo que esta entrada somente será usada caso a rota anterior (rota com prefixo mais longo) não atenda. Para o prefixo 128/1, ambas subdivisões possuem o mesmo número de entradas. Neste caso, para cada subdivisão busca-se a maior máscara de rede que as diferencie. Na primeira subdivisão, referente aos menores prefixos de rede, esta máscara vai até o bit de posição 4, resultando no prefixo 128/4. Para a segunda subdivisão, esta máscara vai até o bit de posição 7, resultando no prefixo 192/7.

O caso com maior eficiência é observado com agrupamentos que possibilitem uma entrada por interface de saída na FIB reduzida. Já o pior caso seria não ter nenhum agrupamento, produzindo uma FIB sem redução. É interessante observar que esta situação também ocorre com as técnicas de agregação apresentadas na Seção 2.

3.2. Cenário de testes e métricas

O cenário testado objetiva uma prova de conceito para reproduzir, numa escala muito menor, um possível arranjo na DFZ.

Para os testes deste trabalho foi utilizada uma máquina com processador *Quad Core* de 64 bits, 8 GBytes de memória e sistema GNU-Linux Debian 6.0.6 como sistema operacional hospedeiro. O ambiente de testes foi criado com máquinas virtuais LXC 0.7.2, do tipo contêiner, nas quais foram instaladas a suíte de roteamento BIRD 1.3.4 [de Farias et al. 2013].

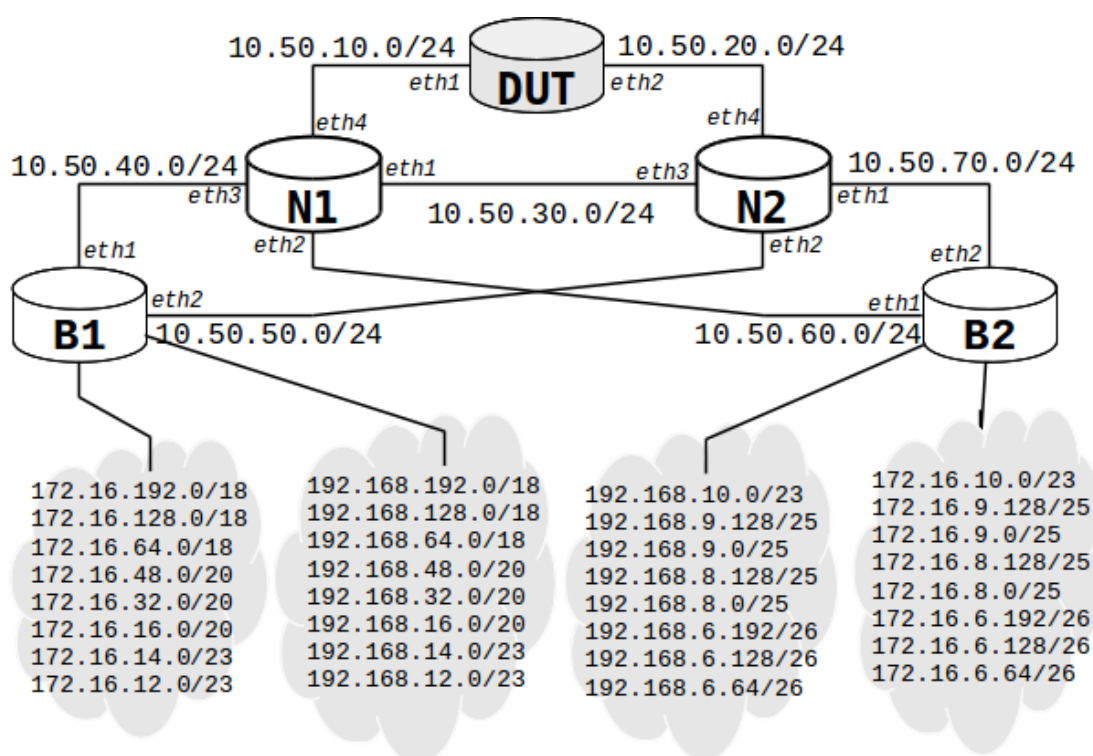


Figura 10. Cenário usado para os testes.

A conectividade que permite a troca de mensagens entre os contêineres é obtida através da associação da interface de rede local *ethernet* (ETH) e as interfaces virtuais (VETH - *Virtual Ethernet device*). Isto é feito através da configuração de uma ponte (*linux bridge*) no *kernel* do Linux da máquina real. Cada VETH tem associado um número *ethernet* (MAC address) e um número IP. Desta forma, cada contêiner corresponde a um roteador e consiste em uma cópia isolada da suíte de roteamento. Portanto, neste experimento com cinco roteadores existem cinco contêineres virtuais.

Três ASN são usados, onde o roteador DUT (*device under test*) é o único no AS10, os roteadores N1 e B1 estão no AS100 e os roteadores N2 e B2 no AS200. A função do DUT no cenário é coletar as métricas para análise [Manral et al. 2005], por isso ele recebe os anúncios BGP, envia pacotes ICMP e mantém registros de uso da memória e CPU. O núcleo é formado pelos roteadores DUT, N1 e N2 e as bordas pelos roteadores B1 e B2. Trinta e duas rotas são anunciadas por ISPs subordinados e estão caracterizadas como nuvens conectadas às bordas. A existência de caminhos redundantes nos clientes

foi obtida com a topologia em malha totalmente ligada entre N1, N2, B1 e B2. A Figura 10 representa essa topologia.

Duas métricas são analisadas neste trabalho: o tamanho da tabela de roteamento e a latência da rede. Na primeira métrica é verificada a taxa de agregação do algoritmo proposto. Para isso, a quantidade de entradas na FIB gerada pela solução proposta será comparada com o tamanho da FIB gerada pela solução adotada pelo BIRD de forma dinâmica. Na segunda métrica, foi coletada a latência da rede conforme a RFC 3222 [Trotter 2001] para medição da performance do roteador considerando o tamanho da FIB.

A coleta dos dados será realizada por um *script* residente em memória *daemon* a intervalos de 1 segundo em 30 rodadas de testes. Os testes são executados em ciclos caracterizados por quatro fases. Na primeira fase é feita a criação de um novo ambiente virtual com o anúncio dos blocos de endereço até ocorrer a convergência da rede. Na segunda fase é implementado o algoritmo proposto que coleta os dados da RIB e calcula a solução. Na terceira fase o BIRD é desativado e a solução proposta é implementada no *kernel* como rotas estáticas, de maneira a ser possível verificar o funcionamento da FIB obtida através do algoritmo de redução proposto. Na fase final o ambiente virtual é eliminado, garantindo que a próxima rodada seja isolada das anteriores.

4. Resultados

4.1. Redução da FIB

A RIB obtida no DUT apresenta 68 rotas para alcançar os prefixos anunciados, incluindo os caminhos alternativos. Destas, 44 prefixos de destino e os respectivos endereços de próximo salto são passados para a FIB conforme a solução tradicional implementada no BIRD que é observada na Figura 11. Isso permite observar a simplificação usual na FIB, pois 35,2% das rotas não são registradas. A FIB do BIRD foi coletada através do comando *ip route show*.

```
~# ip route show
10.10.10.0/24 dev eth0 proto kernel scope link src 10.10.10.1
10.10.30.0/24 via 10.50.10.2 dev eth1 proto bird
10.10.60.0/24 via 10.50.20.2 dev eth2 proto bird
10.10.80.0/24 via 10.50.10.2 dev eth1 proto bird
10.10.100.0/24 via 10.50.20.2 dev eth2 proto bird
10.50.10.0/24 dev eth1 proto kernel scope link src 10.50.10.1
10.50.20.0/24 dev eth2 proto kernel scope link src 10.50.20.1
10.50.30.0/24 via 10.50.10.2 dev eth1 proto bird
10.50.40.0/24 via 10.50.10.2 dev eth1 proto bird
10.50.50.0/24 via 10.50.10.2 dev eth1 proto bird
10.50.60.0/24 via 10.50.10.2 dev eth1 proto bird
10.50.70.0/24 via 10.50.20.2 dev eth2 proto bird
172.16.6.64/26 via 10.50.20.2 dev eth2 proto bird
172.16.6.128/26 via 10.50.20.2 dev eth2 proto bird
172.16.6.192/26 via 10.50.20.2 dev eth2 proto bird
172.16.8.0/25 via 10.50.20.2 dev eth2 proto bird
```

```

172.16.8.128/25 via 10.50.20.2 dev eth2 proto bird
172.16.9.0/25 via 10.50.20.2 dev eth2 proto bird
172.16.9.128/25 via 10.50.20.2 dev eth2 proto bird
172.16.10.0/23 via 10.50.20.2 dev eth2 proto bird
172.16.12.0/23 via 10.50.10.2 dev eth1 proto bird
172.16.14.0/23 via 10.50.10.2 dev eth1 proto bird
172.16.16.0/20 via 10.50.10.2 dev eth1 proto bird
172.16.32.0/20 via 10.50.10.2 dev eth1 proto bird
172.16.48.0/20 via 10.50.10.2 dev eth1 proto bird
172.16.64.0/18 via 10.50.10.2 dev eth1 proto bird
172.16.128.0/18 via 10.50.10.2 dev eth1 proto bird
172.16.192.0/18 via 10.50.10.2 dev eth1 proto bird
192.168.6.64/26 via 10.50.20.2 dev eth2 proto bird
192.168.6.128/26 via 10.50.20.2 dev eth2 proto bird
192.168.6.192/26 via 10.50.20.2 dev eth2 proto bird
192.168.8.0/25 via 10.50.20.2 dev eth2 proto bird
192.168.8.128/25 via 10.50.20.2 dev eth2 proto bird
192.168.9.0/25 via 10.50.20.2 dev eth2 proto bird
192.168.9.128/25 via 10.50.20.2 dev eth2 proto bird
192.168.10.0/23 via 10.50.20.2 dev eth2 proto bird
192.168.12.0/23 via 10.50.10.2 dev eth1 proto bird
192.168.14.0/23 via 10.50.10.2 dev eth1 proto bird
192.168.16.0/20 via 10.50.10.2 dev eth1 proto bird
192.168.32.0/20 via 10.50.10.2 dev eth1 proto bird
192.168.48.0/20 via 10.50.10.2 dev eth1 proto bird
192.168.64.0/18 via 10.50.10.2 dev eth1 proto bird
192.168.128.0/18 via 10.50.10.2 dev eth1 proto bird
192.168.192.0/18 via 10.50.10.2 dev eth1 proto bird

```

Figura 11. FIB obtida com o BIRD.

Com a utilização do algoritmo proposto, a FIB obtida apresenta um total de 12 prefixos de rede o que equivale a uma redução de 72,7% com relação a FIB obtida com tradicional. Este resultado é listado na Figura 12.

```

~# ip route show
10.10.10.0/24 dev eth0 proto kernel scope link src 10.10.10.1
10.10.30.0/24 via 10.50.10.2 dev eth1 proto bird
10.10.60.0/24 via 10.50.20.2 dev eth2 proto bird
10.10.80.0/24 via 10.50.10.2 dev eth1 proto bird
10.10.100.0/24 via 10.50.20.2 dev eth2 proto bird
10.50.10.0/24 dev eth1 proto kernel scope link src 10.50.10.1
10.50.20.0/24 dev eth2 proto kernel scope link src 10.50.20.1
10.50.32.0/20 via 10.50.10.2 dev eth1 proto bird
172.16.8.0/20 via 10.50.20.2 dev eth2 proto bird
192.168.8.0/20 via 10.50.20.2 dev eth2 proto bird

```

```
172.16.128.0/17 via 10.50.10.2 dev eth1 proto bird
192.168.128.0/17 via 10.50.10.2 dev eth1 proto bird
```

Figura 12. FIB obtida com o algoritmo de filtragem.

O algoritmo proposto pode ainda ser modificado de maneira que uma rota padrão seja empregada para aumentar ainda mais a taxa de agregação. No caso do cenário adotado, nota-se que a interface eth1 é utilizada por 31 das 68 rotas da RIB. Por conta disto, se a solução proposta fosse implementada para que a eth1 represente uma rota padrão (0.0.0.0/0), o tamanho da FIB seria reduzido de 12 para 7 rotas.

4.2. Latência

A verificação da influência da redução da FIB no tempo de ida e volta de um pacote foi obtida com o comando *ping*. Para isso 30 pacotes com tamanho de 900 bytes foram enviados para as SRs B1 e B2 a partir do DUT. A Tabela 1 relaciona as médias dos tempos mínimo, médio e máximo de ida e volta do pacote (RTT - *Round Trip Time*). Em média, houve uma redução de 31% no RTT.

	min.	méd.	máx.
Sem redução	0,077	0,199	0,759
Com redução	0,062	0,137	0,303

Tabela 1. Tempo de ida e volta em milissegundos.

5. Conclusão

O aumento da FIB é um fator tecnológico que causa impacto no desempenho dos roteadores e que tem se intensificado com o emprego de virtualização e SDN. Neste contexto, este trabalho propõe um algoritmo para redução da FIB sob a ótica do enlace físico da rede. Este algoritmo usa uma abordagem na qual são criados agrupamentos de prefixos de destino cujos endereços de próximo salto estejam conectados através de uma mesma interface de rede. Procura-se então obter o menor número possível de agrupamentos que sejam associados a prefixos que exclusivamente representem faixas de rede distintas que possuam uma mesma interface de saída.

Para avaliar o funcionamento da solução proposta e a consequente melhora de desempenho pela redução da FIB, utilizou-se suítes de roteamento BIRD executadas em ambiente virtualizado. Métricas associadas com a redução da FIB e a consequente redução da latência foram analisadas em experimentos realizados num cenário DFZ, ou seja, sem rota padrão, constatando-se a melhoria obtida com o algoritmo proposto.

Como trabalho futuro, pretende-se fazer uma projeção em cenário real usando as tabelas de roteamento BGP disponibilizadas nas bases de dados do projeto Route Views [Routeviews 2014, Caida 2014].

Referências

Alderson, D., Li, L., Willinger, W., and Doyle, J. C. (2005). Understanding internet topology: principles, models, and validation. *Networking, IEEE/ACM Transactions on*, 13(6):1205–1218.

- Ballani, H., Francis, P., Cao, T., and Wang, J. (2009). Making Routers Last Longer with ViAggre. In *Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation, NSDI'09*, pages 453–466, Berkeley, CA, USA. USENIX Association.
- Berkowitz, H., Davies, E. B., Hares, S., Krishnaswamy, P., and Lepp, M. (2005). Terminology for Benchmarking BGP Device Convergence in the Control Plane. RFC 4098 (Informational). In RFC 4098. IETF Network Working Group. June 2005. Category: Informational. Online. Acessado em Agosto/2013. Disponível em: <http://tools.ietf.org/html/rfc4098>.
- Bonaventure, O., Filsfils, C., and Francois, P. (2007). Archiving Sub-50 Milliseconds Recovery Upon BGP Peering Link Failures. *Networking, IEEE/ACM Transactions on*, 15(5):1123–1135.
- Caida (2014). The Cooperative Association for Internet Data Analysis. University of California San Diego Supercomputer Center. Acessado em Março/2014. Disponível em: <http://www.caida.org>.
- Corrêa, C., Lucena, S., Rothenberg, C., and Salvador, M. (2012). Uma plataforma de roteamento como serviço baseada em redes definidas por software. In *XVII Workshop de Gerência e Operação de Redes e Serviços*. SBC - Sociedade Brasileira de Computação.
- de Farias, L. F. T., Diniz, M. C., de Lucena, S. C., and Corrêa, C. N. A. (2013). Um Estudo Comparativo de Softwares de Roteamento para Uso em Redes Definidas por Software. In *Workshop em Desempenho de Sistemas Computacionais e de Comunicação*. SBC - Sociedade Brasileira de Computação.
- Draves, R. P., King, C., Venkatachary, S., and Zill, B. D. (1999). Constructing optimal ip routing tables. In *INFOCOM'99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 1, pages 88–97. IEEE.
- Fuller, V. and Li, T. (2006). Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan. In RFC 4632. IETF Network Working Group. August 2006. Category: Best Current Practice. The Internet Society. Online. Acessado em Novembro/2013. Disponível em: <http://tools.ietf.org/html/rfc4632>.
- Huston, G. (2013). BGP Routing Table Analysis Reports. In Personal site of Geoff Huston that provides updated statistical analysis and reports on Internet BGP data. Online. Acessado em Setembro/2013. Disponível em: <http://bgp.potaroo.net>.
- Kumar, M. and Kumar, S. (2013). Improving routing in large networks inside autonomous system. *International Journal of System Assurance Engineering and Management*, pages 1–8. Springer India.
- Liu, Y., Zhang, B., and Wang, L. (2013). FIFA: Fast incremental FIB aggregation. In *INFOCOM, 2013 Proceedings IEEE*, pages 1–9. IEEE.
- Mai, J. and Du, J. (2013). BGP performance analysis for large scale VPN. In *Information Science and Technology (ICIST), 2013 International Conference on*, pages 722–725.
- Manral, V., White, R., and Shaikh, A. (2005). Benchmarking basic ospf single router control plane convergence. In RFC 4061. IETF Internet Requests for Comments. April

2005. Category: Informational. Online. Acessado em Junho/2012. Disponível em: <http://tools.ietf.org/html/rfc4061>.
- Meyer, D., Zhang, L., and Fall, K. (2007). Report from the IAB Workshop on Routing & Addressing. In RFC 4984. IETF Network Working Group. April 2007. Category: Internet-Draft. Online. Acessado em Junho/2013. Disponível em: <http://tools.ietf.org/html/draft-iab-raws-report-02>.
- Mohamed, H., Bashandy, A., and Shaheen, S. I. (2012). Protocol based selective FIB download for distributed forwarding architecture. In *Globecom Workshops (GC Wkshps), 2012 IEEE*, pages 7–12. IEEE.
- Morrison, D. R. (1968). Patricia—practical algorithm to retrieve information coded in alphanumeric. *Journal of the ACM (JACM)*, 15(4):514–534.
- Raszuk, R., Heitz, J., Lo, A., Zhang, L., and Xu, X. (2012). Report from the IAB Workshop on Routing & Addressing. In RFC 6769. IETF Internet Requests for Comments. October 2012. Category: Informational. Online. Acessado em Junho/2013. Disponível em: <http://tools.ietf.org/html/rfc6769>.
- Rottenstreich, O., Radan, M., Cassuto, Y., Keslassy, I., Arad, C., Mizrahi, T., Revah, Y., and Hassidim, A. (2013). Compressing forwarding tables. In *INFOCOM, 2013 Proceedings IEEE*, pages 1231–1239.
- Routeviews (2014). Route Views Project. University of Oregon. Acessado em Março/2014. Disponível em: <http://www.routeviews.org>.
- Sánchez, B. C. (2013). Virtual Aggregation in OpenFlow Networks. Master's thesis, KTH Royal Institute of Technology, Stockholm, Sweden.
- Trotter, G. (2001). Terminology for forwarding information base (fib) based router performance. In RFC 3222. IETF Internet Requests for Comments. December 2001. Category: Informational. Online. Acessado em Agosto/2013. Disponível em: <http://tools.ietf.org/html/rfc3222>.
- Uzmi, Z. A., Nebel, M., Tariq, A., Jawad, S., Chen, R., Shaikh, A., Wang, J., and Francis, P. (2011). SMALTA: Practical and Near-optimal FIB Aggregation. In *Proceedings of the Seventh Conference on Emerging Networking EXperiments and Technologies, CoNEXT '11*, pages 29:1–29:12, New York, NY, USA. ACM.