Universidade do Minho
Escola de Engenharia

Hélder José da Costa Ribeiro

Encaminhamento inter-domínios por classes de serviço – Uma Proposta

Encaminhamento inter-domínios por classes de serviço – Uma Proposta
Hélder José da Costa Ribeiro
UMinho | 2012

Dezembro de 2012

Universidade do Minho
Escola de Engenharia

Hélder José da Costa Ribeiro

Encaminhamento inter-domínios por classes de serviço – Uma Proposta

Tese de Mestrado
Ciclo de Estudos Integrados Conducentes ao
Grau de Mestre em Engenharia Comunicações

Trabalho efetuado sob a orientação de
Professor Doutor António Costa
Professora Doutora Maria João Nicolau

Dezembro de 2012

DECLARAÇÃO

Nome: Hélder José da Costa Ribeiro

Correio electrónico: helder.dacosta.ribeiro@gmail.com

Tel./Tlm.: 917279755

Número do Bilhete de Identidade: 13201820

Título da dissertação: Encaminhamento inter-domínios por classes de serviço – Uma Proposta.

Ano de conclusão: 2012

Orientadores: Professor Doutor António Costa

Professora Doutora Maria João Nicolau

Designação do Mestrado:

Ciclo de Estudos Integrados Conducentes ao Grau de Mestre em Engenharia Comunicações

Escola: Engenharia

Departamento: Sistemas de Informação

Guimarães, ___/___/_____

Assinatura: _____

"Os amantes da concisão, do modo lacónico, da economia de linguagem, decerto se estarão perguntando porquê, sendo a ideia assim tão simples, foi preciso todo este arrazoado para chegarmos enfim ao ponto crítico. A resposta também é simples, e vamos dá-la utilizando um termo actual, moderníssimo, com o qual gostariamos de ver compensados os arcaismos com que, na provável opinião de alguns, hemos salpicado de mofo este relato, Por mor do background."

José Saramago - *As intermitências da Morte*

Agradecimentos

Enquanto o uso da primeira pessoa do singular me é permitido, eu gostava de mencionar que este documento reflecte o meu trabalho no âmbito da dissertação de mestrado, mas também a influência de algumas pessoas no trabalho e no meu percurso.

Desta forma, para além de um agradecimento, reconheço que:

A minha familia — O Bernardino, a Adelina, a Sofia, o Simão e a Luísa — são importantes demais para terem uma importância que se destaque em particular.

Os meus orientadores, o Professor António Costa e a Professora Maria João Nicolau, quer nos meses em que este trabalho se desenrolou, quer nas disciplinas que me leccionaram, foram uma inspiração e motivo pelo interesse na área de redes de computadores e na conclusão deste documento.

A Maria José e o José Oliveira, desde a minha mais remota memória, foram uma influência no sentido de aprender sempre mais e ser curioso.

O César, o Hugo, o João, o Manuel, o Freddy, o Diogo, o Roupar, a Daniela, o Nuno, o Tiago, o Ricardo, entre uma lista de pares, deram um contributo para este documento.

Existe um conjunto de pessoas, P, que teve influência no meu percurso. Cada $P_i \in P$, devido ao teorema da sobreposição, tem uma influência directa no meu percurso.

Resumo

O presente documento enquadra-se no tema do encaminhamento inter-domínios com informações de qualidade de serviço (QoS). O protocolo utilizado para o encaminhamento inter-domínio é o protocolo BGP e não possui, na sua especificação, qualquer tipo de mecanismos de QoS associados. Estender o protocolo BGP, por forma a permitir encaminhamento com QoS, é, portanto, vital para a desejada qualidade de serviço de fim-a-fim.

O presente documento tem duas partes. Numa primeira parte apresenta-se o conceito de encaminhamento inter-domínio, bem como a especificação e implementação de um modelo, do protocolo BGP no simulador NS-3.

Na segunda parte apresenta-se um estudo sobre o encaminhamento de tráfego com qualidade de serviço. Este estudo é a base da concepção de uma nova proposta para encaminhamento com classes de serviço e representação estatística de métricas de QoS, também apresentada no presente documento.

Palavras Chave: Encaminhamento inter-domínio, encaminhamento com QoS, BGP, NS-3.

Abstract

The theme of the present document is inter-domain routing and forwarding with Quality of Service (QoS) information. The *de facto standard* for inter-domain routing – the BGP protocol – has no QoS mechanisms in its conception. This way, the extension of the BGP protocol, in order to provide QoS routing, is vital to the desired end-to-end QoS.

This document has two parts. In the first part is presented the inter-domain routing, as well as the implementation of a model of the BGP protocol in the NS-3 simulator.

In the second part is presented a study, concerning inter-domain routing and forwarding with Quality of Service (QoS) information. This study is the basis of a new proposal, with classes of service and statistical representation of QoS metrics, that's also presented in this document.

Keywords: Inter-domain routing; QoS-Routing; BGP; NS-3.

Índice

Lista de Figuras

Lista de Tabelas

1 Introdução

1.1 Enquadramento do Problema

A Internet é, hoje em dia, uma infra-estrutura que suporta todo o mundo às costas. Já não tem só fins académicos ou de investigação, mas sociais, recreativos e principalmente, comerciais. Acreditando que exista uma lei de Moore para a Internet [2], faz todo o sentido o estudo de formas de optimizar o encaminhamento de tráfego e assim o desempenho da Internet, como por exemplo a introdução de Qualidade de Serviço (QoS).

A Internet pode, portanto, crescer ou evoluir. Se por crescer entende-se aumentar a capacidade dos encaminhadores ligações entres estes, por exemplo, por evoluir entende-se implementar novos mecanismos, ou melhorias de mecanismos já existentes, que permitam, com o mesmo tamanho, fornecer mais e melhor serviços. O presente trabalho enquadra-se no fazer evoluir a Internet.

A Internet não é mais que um grande conjunto de redes, estas redes agregam-se em sistemas autónomos, que são, de uma forma geral, conjuntos de redes administrados pela mesma entidade. Existem vários mecanismos

de introdução de QoS para o encaminhamento de tráfego dentro de um Sistema Autónomo, no entanto para a comunicação entre estes domínios não há um desenvolvimento significativo, no sentido de incorporar QoS. [3]

O facto de o protocolo de encaminhamento de tráfego inter-domínios, – o protocolo BGP (*Border Gateway Protocol*) – não ter como objectivo o desempenho da rede, mas sim as relações comerciais entre os sistemas autónomos, é um dos vários entraves à incorporação de extensões de QoS.

1.2 Objectivos

Não existe, actualmente, uma bancada de teste do protocolo BGP, onde se possam testar, de forma expedita, eventuais extensões ao protocolo. Existe a possibilidade de desenvolver aplicações sobre a implementações do BGP para encaminhadores, no entanto, entende-se que o desenvolvimento em ambiente de simulação permite obter resultados mais rapidamente. Nesse sentido, pretende-se desenvolver um bancada de teste do protocolo BGP para o simulador NS-3. Existem implementações do BGP no antecessor do NS-3 (o NS-2), no entanto a diferença na arquitectura dos dois simuladores leva à decisão de implementar o BGP neste simulador.

O trabalho a que se refere o presente documento, tem também como grande objectivo, o estudo de vários trabalhos sobre o encaminhamento inter-domínio com QoS e apresentação de uma proposta para o problema. É objectivo que esta proposta tenha em conta o trabalho estudado como ponto de partida.

1.3 Estrutura do Documento

O presente documenta encontra-se, então organizado da seguinte forma. No capítulo 2 apresenta-se uma introdução ao tema do encaminhamento de tráfego entre domínios. Posteriormente descreve-se o *standard de facto* desse cenário – o protocolo BGP.

O capítulo 3 refere-se ao primeiro objectivo do trabalho, a implementação de uma bancada de testes para protocolo BGP. Neste capítulo descreve-se o simulador onde se vai implementar o protocolo BGP e descreve-se, também a implementação.

O capítulo 4 apresenta o estudo realizado sobre o encaminhamento inter-domínios com qualidade de serviço. Neste capítulo são apresentadas duas propostas, de extensões do BGP, para permitir o encaminhamento com classes de serviço e uma proposta defendendo a inclusão de métricas estatísticas, no sentido de manter a estabilidade e escalabilidade do protocolo BGP.

No capítulo 5 apresenta-se uma proposta para o encaminhamento de tráfego entre domínios com classes de serviço. Esta proposta tem como base os trabalhos descritos no capítulo 4.

Por fim, no capítulo **??**, conclui-se o presente documento, apresentado o trabalho realizado, as contribuições que surgiram da realização do mesmo e o trabalho futuro.

2 Encaminhamento de Tráfego Inter-Domínio

"The technology, called internetworking, accommodates multiple, diverse underlying hardware technologies by providing a way to interconnect heterogeneous networks and a set of communication conventions that makes them interoperate." [4]

2.1 Introdução

O presente capítulo trata o encaminhamento de tráfego na internet, com foco sobre o encaminhamento de tráfego entre domínios. É feita uma breve abordagem ao encaminhamento de tráfego, onde são expostos conceitos básicos, mas fundamentais para o problema em questão. Partindo do encaminhamento de tráfego, apresenta-se o caso concreto do encaminhamento inter-domínio. Posteriormente, descreve-se o protocolo de encaminhamento inter-domínio que é utilizado na Internet. É descrito o comportamento do protocolo, bem como o formato das mensagens e atributos do mesmo.

Figura 2.1: *Arquitectura de cinco camadas da Internet*

2.2 Encaminhamento em Redes IP

> *"The main function of the network layer is routing packets from the source machine to the destination machine." [5]*

2.2.1 O Protocolo de Rede

A bibliografia referente a redes de computadores refere com recorrência o modelo OSI (*Open Sistems Interconnect*) da ISO (*International Organization for Standardization*). O modelo OSI aplica-se na comunicação entre sistemas heterogéneos e define sete camadas de abstracção, onde cada camada se refere a uma área funcional e fornece serviços à camada acima. As arquitecturas da Internet, no entanto, implementam um modelo de cinco camadas para a comunicação entre sistemas (RFC 1122 [6]), como ilustrado na figura 2.1. As cinco camadas implementam as seguintes funções:

Aplicação A camada de Aplicação é a camada de topo da pilha e especifica protocolos que servem directamente as aplicações, como por exemplo o FTP (*File Transfer Protocol*), o TELNET ou o HTTP (*Hypertext Transfer Protocol*).

Transporte A camada de transporte fornece um serviço de comunicação fim-a-fim entre aplicações. Os dois principais protocolos desta camada são o TCP (*Transmission Control Protocol*)(RFC 793 [7]) e o UDP (*User Datagram Protocol*) (RFC 768 [8]). O Protocolo UDP fornece um serviço não orientado à conexão e não-fiável para envio de mensagens individuais. Se uma aplicação pretender fiabilidade com UDP deverá implementar essa fiabilidade. O TCP é um serviço de transporte orientado à conexão que fornece fiabilidade fim-a-fim.

Rede Os protocolos de transporte utilizam como protocolo de rede o protocolo IP (Internet Protocol) (RFC 791 [9]). O IP fornece um serviço de envio de dados entre terminais sem garantias. Os dados podem chegar ao destino fora de ordem ou corrompidos. Qualquer garantia que se pretenda deve ser implementada pelos protocolos das camadas superiores. Esta característica do IP é a base de toda a arquitectura da Internet.

Ligação de Dados A camada de ligação de dados permite a comunicação com terminais de redes directamente conectadas. Existem vários protocolos de ligação de dados, como por exemplo o `Ethernet` IEEE 802.3.

Física A camada Física refere-se aos elementos electromecânicos das LAN (Local Area Network) e redes ponto-a-ponto bem como os seu conectores.

O foco do presente documento e do trabalho realizado é a camada de rede, nomeadamente ao nível do protocolo IP. O protocolo IP (RFC - 791 [9]) é um protocolo não orientado à conexão, logo, os pacotes IP possuem, em si próprios, toda a informação necessária para serem encaminhados até ao destino. Esta propriedade simplifica a complexidade dos encaminhadores na rede, pois essa complexidade é uma função dos encaminhadores ligados na rede e não do número de fluxos de tráfego que podem percorrer um encaminhador.

O protocolo IP tem duas versões, a versão 4 (IPv4) e a mais recente versão 6 (IPv6 – RFC 2460 [10]). A versão 6 do IP tem o objectivo de substituir o IPv4 devido a, não só mas também, terem-se esgotado os endereços IPv4. Para além da diferença no espaço de endereçamento, o IPv6 traz novas funcionalidades, um cabeçalho diferente, mas o protocolo IP mantém-se sem garantias e não orientado à conexão.

2.2.2 Encaminhamento de Tráfego em Redes IP

O protocolo IP, como referido acima, funciona de uma forma não orientada à conexão, encaminhando cada pacote de forma independente. De uma forma geral, cada encaminhador possui uma tabela de encaminhamento, onde estão listadas todas as redes para as quais o encaminhador tem conectividade. Cada entrada da tabela de encaminhamento associa um destino a um conjunto de informações, entre as quais se encontram a interface que o encaminhador utiliza para encaminhar um pacote, bem como o endereço do próximo encaminhador. Quando recebe um pacote IP, um encaminhador procura na tabela pelo destino que mais se aproxima do destino do pacote e encaminha o pacote, pela interface indicado na tabela, para o próximo encaminhador. Na figura

2.2 verifica-se um exemplo de uma tabela de encaminhamento de um sistema terminal, mais concretamente, um computador pessoal.

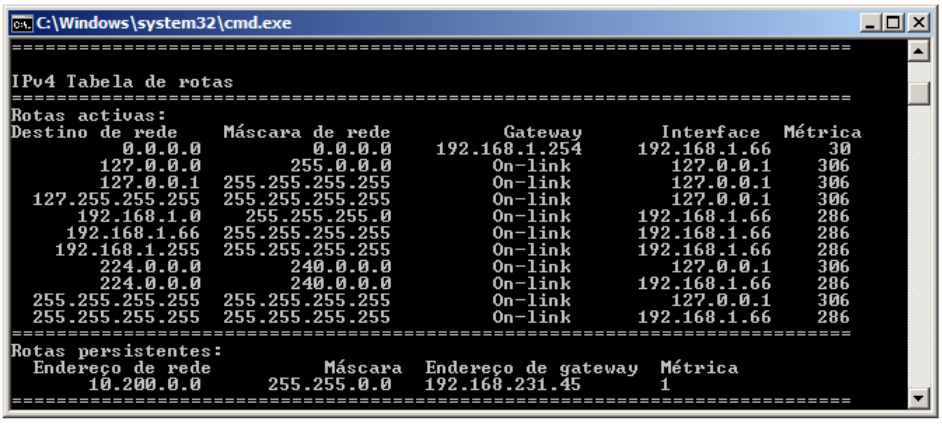

Figura 2.2: *Exemplo de tabela de encaminhamento num terminal*

Por forma a permitir aos encaminhadores preencher a tabela de encaminhamento com mais destinos, é necessário que exista algum processo a descobrir destinos e a associá-los a uma interface de saída. Esse tipo de processos são referidos como protocolos de encaminhamento. Um protocolo de encaminhamento tem como objectivo construir e alterar a tabela de encaminhamento conforme o estado da rede. Tipicamente, os protocolos de encaminhamento de um encaminhador comunicam com os encaminhadores vizinhos através de mensagens especificadas no protocolo. Desta forma, os encaminhadores anunciam os destinos para os quais garantem conectividade permitindo a um encaminhador vizinho a conectividade para um destino ao qual este não se encontra directamente conectado. A figura 2.3 ilustra um exemplo de uma topologia onde é necessário existir um protocolo de encaminhamento. Só após o agente de encaminhamento no encaminhador `R1` anunciar a `R2` a rede `Z` é que `R2` consegue encaminhar tráfego para esta.

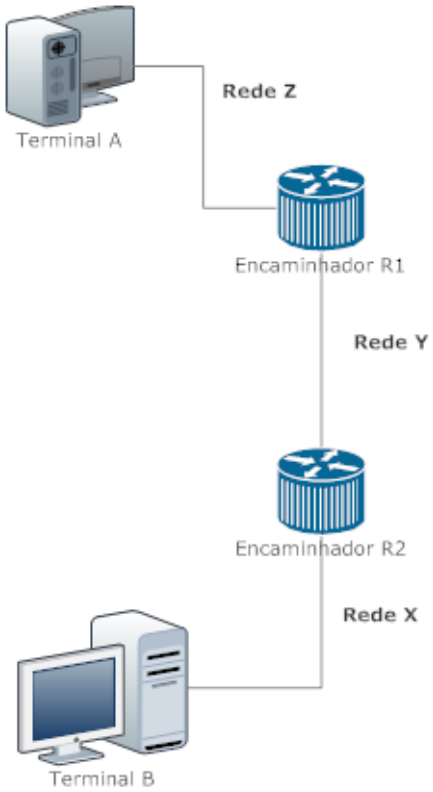

Figura 2.3: *Exemplo de uma topologia de rede simples*

Existem vários exemplos de protocolos de encaminhamento entre eles o RIP (Routing Information Protocol, RFC 2453 [11]), o OSPF (Open Shortest Path First, RFC 2328 [12]), ou o EIGRP (Enhanced Interior Gateway Routing Protocol), sendo o último propriedade da *Cisco Systems*.

2.3 Encaminhamento de Tráfego Inter-Domínio

> *"A different protocol is needed between ASes because the goals of an interior gateway protocol and an exterior gateway protocol are not the same." [5]*

A Internet é actualmente composta por inúmeras redes com as mais variadas especificações técnicas. Um conjunto de redes sob uma entidade administrativa denomina-se Sistema Autónomo (AS, RFC 1930 [13]) ou domínio. Estes domínios apresentam, muitas vezes, relações comerciais entre si, existindo camadas de domínios onde uns são clientes de outros. Dentro de um domínio operam os protocolos de encaminhamento intra-domínio (IGP's - Interior Gateway Protocol). Estes protocolos, tipicamente, têm como principal objectivo garantir conectividade, pelo melhor caminho, entre os diversos encaminhadores e terminais. O termo "melhor caminho" refere-se, no contexto de encaminhamento intra-domínio, ao conjunto de ligações que optimiza as métricas utilizadas dentro do domínio. Estas métricas podem ser atributos das ligações, como por exemplo, a capacidade e o custo em cada encaminhador. Existem diversos protocolos de encaminhamento intra-domínio, sendo o RIP e o OSPF exemplos de padrões de protocolos deste tipo. A figura 2.4 ilustra um exemplo de uma topologia onde existem métricas – largura de banda – associadas às ligações. O melhor caminho entre os terminais `A` e `B` é o caminho que percorre os encaminhadores `R1`, `R2` e `R3`.

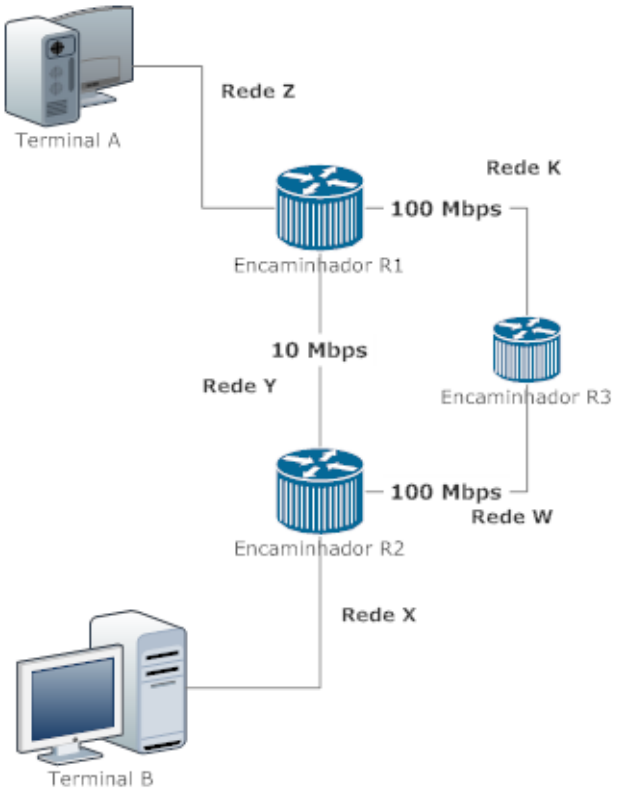

Figura 2.4: *Exemplo de uma topologia de rede com largura de banda de cada ligação*

No caso do encaminhamento inter-domínio, o funcionamento dos protocolos é diferente. Para garantir conectividade entre redes de domínios diferentes é necessário protocolos de encaminhamento inter-domínio (EGP's - Exterior Gateway Protocol). Um AS executa um EGP com os seus vizinhos para troca de informações de conectividade por forma a conseguir conectividade a toda a internet. É esta troca de informações que permite a AS's trocarem tráfego na sua fronteira. Por serem protocolos que operam entre domínios diferentes, os EGP têm de permitir uma variedade de informações que podem ou não ser compreendidas nos domínios vizinhos. Assim sendo, é muito difícil definir mé-

tricas para este tipo de protocolos. Na figura 2.5 apresenta-se um exemplo de uma topologia de rede que apresenta três domínios: `AS 1`, `AS 2` e `AS 3`. Os encaminhadores de cor preta representam os encaminhadores intra-domínio, ou seja, encaminhadores nos quais apenas corre um protocolo intra-domínio. Os encaminhadores de cor azul são denominados encaminhadores-fronteira. Nestes encaminhadores-fronteira, geralmente, existem dois tipos de protocolos em execução: IGP (RIP ou OSPF, por exemplo) e EGP. Os encaminhadores-fronteira transmitem para o domínio as rotas que aprendem pelo protocolo EGP e permitem assim uma conectividade inter-domínios.

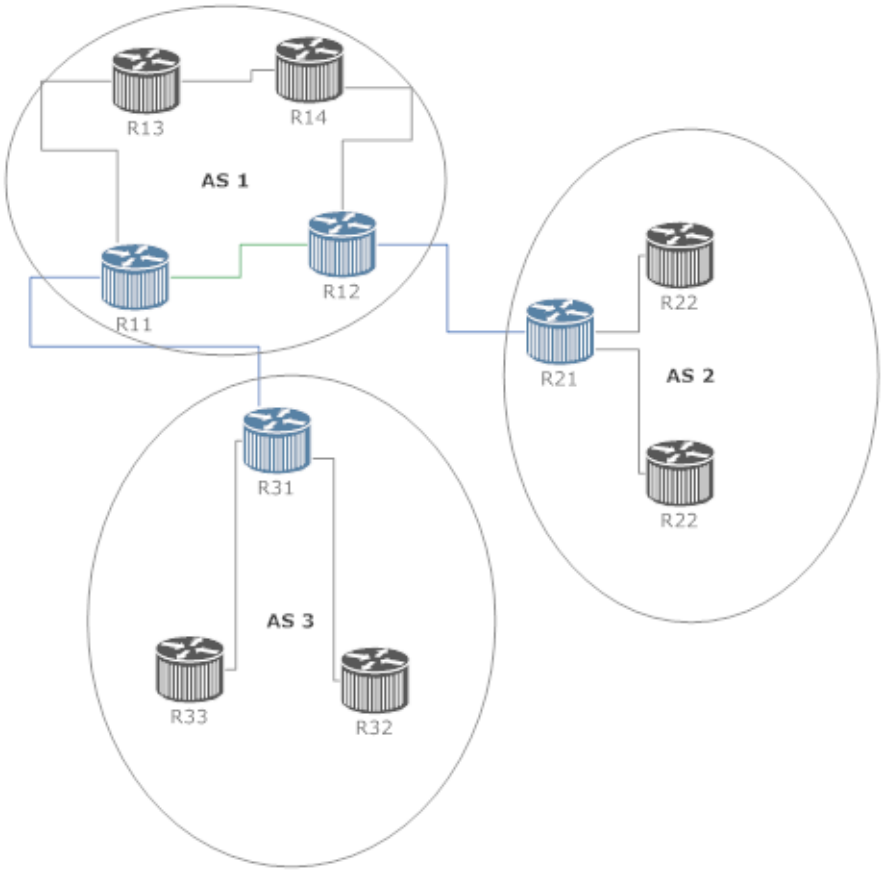

Figura 2.5: *Exemplo de uma topologia inter-domínio com ligações de IGP e EGP*

Contrariamente ao caso intra-domínio, onde são usados vários protocolos diferentes, no caso inter-domínio o *standard de facto* na internet é o protocolo BGP (Border Gateway Protocol, RFC 4271 [14]). O protocolo BGP é portanto um protocolo cujo domínio é toda a Internet e permite, de facto, conectividade global entre terminais e encaminhadores de sistemas autónomos diferentes. Tendo em conta este facto, o foco do presente documento do trabalho realizado, ao nível do protocolo de encaminhamento inter-domínio, cai sobre o protocolo BGP.

2.4 Protocolo BGP

> *"BGP is unusual in several ways. Most important, BGP is neither a pure distance vector protocol nor a pure link state protocol."* [4]

O protocolo BGP é o protocolo utilizado para encaminhamento de tráfego entre domínios. O principal objectivo deste protocolo é a troca de informações de conectividade com outros AS's. Um nó BGP anuncia os prefixos para os quais tem conectividade e a lista dos AS's que compõem o caminho até o prefixo anunciado. Esta informação permite construir um grafo de AS's para evitar ciclos e impor políticas de encaminhamento. Quando um nó anuncia um prefixo com um determinado caminho, deve encaminhar, forçosamente, todo o tráfego com o destino anunciado pelo caminho definido no anúncio. Ou seja, cada AS só anuncia os prefixos que efectivamente utiliza.

O protocolo BGP permite que os prefixos anunciados tenham associados a si um conjunto de atributos. Dois exemplos destes atributos são o caminho para alcançar o prefixo anunciado e a forma como o AS originário descobriu a rota a anunciar. Estes atributos estão descritos na secção 2.4.3.

Uma sessão BGP pretende-se que tenha garantias de fiabilidade e detecção de erros. Como o protocolo IP não apresenta nenhuma dessas garantias, as sessões BGP estabelecem-se sobre a camada de transporte TCP, sendo o primeiro passo para a troca de informações entre os nós, denominados pares. Por definição, e pela utilização do protocolo TCP, uma sessão BGP estabelece-se entre exactamente dois pares. Após estabelecida a ligação os pares trocam a informação de conectividade entre si.

Ao contrário de protocolos IGP, onde são utilizadas métricas que representam a capacidade da rede, protocolo BGP baseia-se em políticas para decidir se uma rota é ou não válida. Estas políticas permitem filtrar rotas por um de-

terminado AS, ou forçar o tráfego a ser encaminhado por um determinado AS, entre outros. Este comportamento traduz uma vertente comercial ou administrativa do protocolo BGP, pois são privilegiadas ligações com alguns pares em detrimento de outros, sem que isso traduza o estado da rede ou qualquer tipo de métricas. O melhor caminho não pode violar relações comerciais entre os sistemas autónomos.

Se as políticas de encaminhamento não produzirem a versão final da tabela de encaminhamento, o BGP utiliza os atributos das rotas para preencher a mesma. Um dos atributos usados refere-se ao número de AS's que compõem o caminho que garante conectividade a um destino – `AS Path`. Este comportamento aproxima o BGP a um protocolo de vectores-distância, sendo mesmo referido na literatura como um protocolo de vectores-caminho (no inglês *path-vector*).

O funcionamento do protocolo BGP pode ser descrito, muito sucintamente, da seguinte forma:

1. Estabelecer ligação BGP com os pares

2. Trocar as informações de conectividade

3. Trocar actualizações às rotas

Numa primeira fase estabelece-se uma ligação BGP com os pares. Esta ligação é feita usando o protocolo TCP na porta 179. Assim que exista uma ligação BGP com um par, são enviadas as rotas que se pretende anunciar. Sempre que existe a necessidade de actualizar uma rota, ou anunciar a não-conectividade a uma rota outrora anunciada, são enviadas actualizações de rotas. Estas mensagens estão bem definidas na norma do BGP.

2.4.1 Mensagens BGP

O protocolo BGP define um conjunto de mensagens enviadas entre os pares para permitir o seu funcionamento. Os tipos de mensagens definidos pelo protocolo BGP são os seguintes:

OPEN - Mensagens enviadas para iniciar uma sessão BGP entre dois pares

UPDATE - Mensagens enviadas para anunciar conectividade

KEEPALIVE - Mensagens enviadas periodicamente para confirmar o estado activo da ligação

NOTIFICATION - Mensagens enviadas em caso de ocorrência de erros no BGP

Todas as mensagens BGP possuem um cabeçalho comum. A figura 2.6 apresenta o cabeçalho comum do BGP O cabeçalho comum é composto pelos

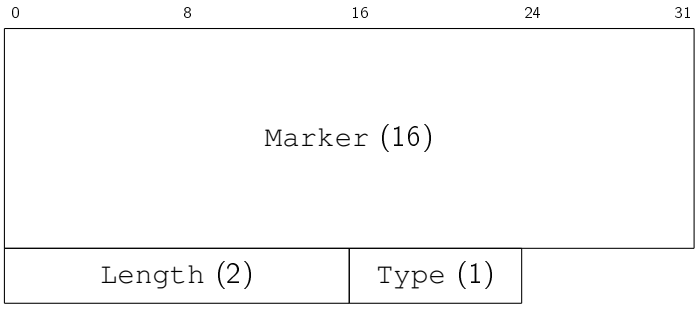

Figura 2.6: *Cabeçalho comum das mensagens BGP*

seguintes campos:

Marker O campo `Marker` ocupa 16 octetos e tem dois propósitos: sincronização e segurança. Caso se trate da primeira mensagem enviada, ou de não se utilizar nenhum mecanismo de segurança, este campo deve

ser composto por 1s. Caso contrário, o valor do campo `Marker` é especificado pelo mecanismo de segurança que se utilizar.

Length O campo `Length` especifica o tamanho total, em octetos, da mensagem BGP, incluindo o cabeçalho comum.

Type O campo `Type` especifica o tipo da mensagem BGP a enviar. Este valor é representado por um número inteiro, relativo ao tipo da mensagem segundo a seguinte regra:

1. `OPEN`
2. `UPDATE`
3. `NOTIFICATION`
4. `KEEPALIVE`

Após o cabeçalho comum, seguem-se, existindo, os dados respectivos de cada mensagem. Estes dados são interpretados conforme o tipo especificado no campo `Type`. O formato das mensagens dos vários tipos tem uma especificação diferente de tipo para tipo, sendo apresentada de seguinte.

2.4.1.1 Mensagem `OPEN`

A mensagem `OPEN` deve ser enviada logo após o estabelecimento da conexão TCP. Esta mensagem permite que um par se identifique perante outro e acordem parâmetros para a sessão BGP. A figura 2.7 ilustra o cabeçalho da mensagem de `OPEN`. Os campos da mensagem são os seguintes:

Version O campo `Version`, que ocupa um octeto, especifica a versão que o par utiliza, sendo um campo não negociável. Se não houver acorde

entre a versão a utilizar, é necessário estabelecer uma nova conexão TCP para negociar uma versão anterior.

My Autonomous System O campo `My Autonomous System` indica o ASN (Autonomous System Number) (RFC 1930 [13]) do par que envia a mensagem de `OPEN`. Este número ASN tem dois octetos e identifica o AS. O ASN permite validar o par, pois na configuração inicial de cada encaminhador, o ASN de cada par é definido. É também através do ASN que um par descobre se o seu par pertence ao mesmo AS, tendo dessa foram um comportamento ligeiramente diferente nessa sessão.

Hold Time O campo `Hold Time` ocupa dois octetos do cabeçalho da mensagem `OPEN` e tem como objectivo especificar o número de segundos que o par propõe para o temporizador da sessão BGP. Este temporizador está associado a cada ligação BGP, sendo o tempo máximo em segundos que um encaminhador pode esperar por receber dados so seu par. Caso um encaminhador não receba do seu par uma mensagem `UPDATE` ou `KEEPALIVE` num período de tempo do temporizador, a sessão BGP considera-se expirada e as rotas recebidas por esse par são eliminadas. Caso o valor do campo `Hold Time` seja zero, significa que não existem temporizadores. O valor mínimo, não-nulo, do temporizador é de três segundos. O valor do temporizador de uma conexão é o mínimo do valor proposto pelos dois pares.

BGP Identifier O campo `BGP Identifier` ocupa quatro octetos e pretende identificar o par que envia a mensagem `OPEN`. Geralmente, embora não especificado no padrão, os encaminhadores utilizam para este valor um endereço de uma interface virtual que possuam.

Optional Parameters Length O campo `Optional Parameters Length` ocupa um octeto da mensagem e especifica o número de octetos que ocupa o campo `Optional Parameters`.

Optional Parameters O campo `Optional Parameters` tem comprimento variável e contem quaisquer parâmetros opcionais para o BGP. Entre os parâmetros opcionais inclui-se o parâmetro `Capabilities` (RFC [15]), este parâmetro tem como objectivo facilitar a introdução de novas capacidades ao BGP. Os parâmetros adicionais são enviados na seguinte codificação, como ilustra a figura2.8:

Parm. Type O campo `Parm. Type` identifica o parâmetro opcional, ocupando um octeto.

Parm. Length O campo `Parm. Length` especifica o tamanho do parâmetro opcional, ocupando também um octeto.

Parm. Value O campo `Parm. Value` especifica o valor do parâmetro opcional, podendo ocupar um número variável de octetos.

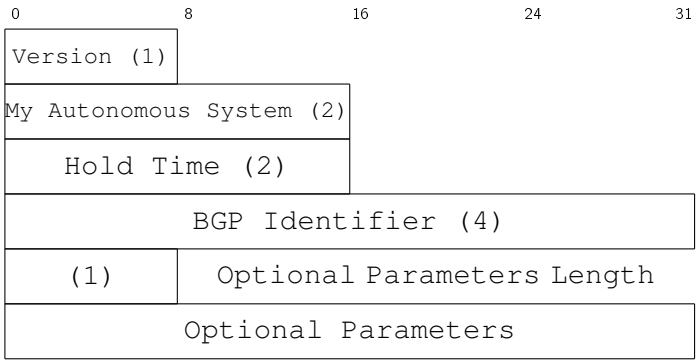

Figura 2.7: *Cabeçalho da mensagem `OPEN` do BGP*

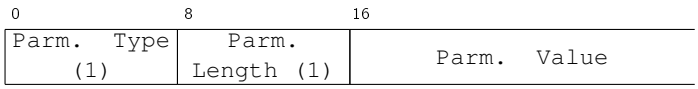

Figura 2.8: *Codificação dos parâmetros opcionais de uma mensagem* `OPEN`

2.4.1.2 Mensagem `UPDATE`

As mensagens do tipo `UPDATE` são utilizadas para anunciar as informações de conectividade. Um par, numa mensagem `UPDATE` pode:

- Anunciar prefixos com uma lista de atributos comuns, aos quais o par que faz o anuncio garante conectividade.
- Anunciar prefixos aos quais, o par que faz o anúncio, não garante mais conectividade.

O formato da mensagem `UPDATE`, que se encontra ilustrado na figura 2.9, é composto pelos seguintes campos:

`Unfeasible Routes Length` O campo `Unfeasable Routes Length`, que ocupa dois octetos, especifica o tamanho, em octetos, do campo `Unfeasible Routes`. Este campo permite também calcular o tamanho do campo `Network Layer Reachability Information`.

`Unfeasible Routes` O campo `Unfeasible Routes` inclui a lista dos prefixos, para os quais o anunciante não garante mais conectividade. Os prefixos são codificados na mensagem, conforme ilustra a figura 2.10, da seguinte forma:

`Length` O campo `Length`, que ocupa um octeto, indica o número de *bits* que compõem o prefixo.

Prefix O campo `Prefix`, de tamanho variável, indica o prefixo IP (IPv4 ou IPv6) anunciado. Caso o valor deste campo, em *bits*, não seja múltiplo de oito, são adicionados *bits* adicionais até ao campo poder ser dividido em octetos. Estes *bits* adicionais não têm influência nos dados, devido ao valor do campo `Length` truncar os tamanho dos dados.

Path Attributes Length O campo `Path Attributes Length`, que ocupa dois octetos, indica o tamanho, em octetos, do campo `Path Attributes`.

Path Attributes O campo `Path Attributes`, de tamanho variável, especifica os atributos referentes a todas as rotas anunciadas na mensagem `UPDATE`. A forma como os atributos se integram na mensagem `UPDATE` é exposta mais adiante.

Network Layer Reachability Information O campo `Network Layer Reachability Information`, de tamanho variável, contém a lista dos prefixos aos quais o par anuncia conectividade. Os prefixos anunciados são codificados da mesma forma que os prefixos no campo `Unfeasable Routes`, conforme ilustra a imagem 2.10. Podem ser anunciados vários prefixos, sendo que para tal é necessário que os atributos do campo `Path Attributes` se apliquem a todos os prefixos anunciados.

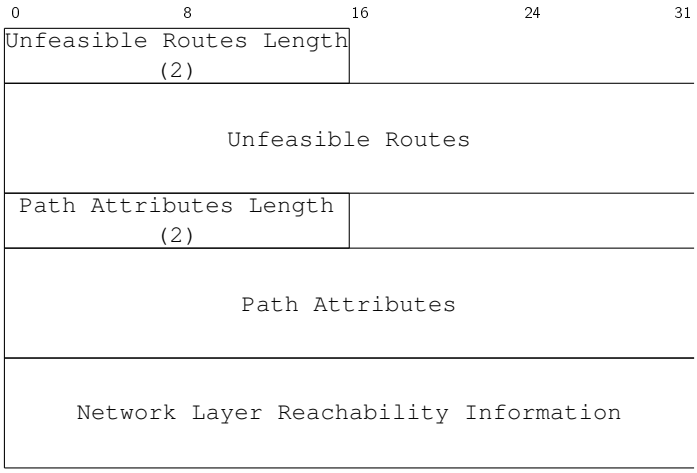

Figura 2.9: *Cabeçalho da mensagem* `UPDATE`

Figura 2.10: *Codificação dos prefixos IP na mensagem* `UPDATE`

O campo `Path Attributes` da mensagem `UPDATE`, como referido acima, contém a lista dos atributos BGP, associados aos prefixos anunciados no campo `NLRI`. Estes atributos caracterizam o prefixo em termos de `AS Path`, origem do prefixo, próximo salto, entre outros. Estes atributos influem na consideração das rotas para as tabelas de encaminhamento do par. O tipo e a utilização dos atributos será descrito mais adiante. A presente secção descreve como os atributos são integrados e codificados na mensagem `UPDATE`, mais propriamente no campo `Path Attributes`.

O campo `Path Attributes` é composto por três campos, como se verifica na figura 2.11, sendo eles:

Attribute Type O campo `Attribute Type`, que ocupa um octeto, representa o tipo do atributo e algumas definições do atributo. Este campo será descrito mais adiante.

Attribute Length O campo `Attribute Length`, pode ocupar um ou dois octetos, consoante o especificado no campo `Attribute Type`. Este campo define o tamanho, em octetos, do campo `Attribute Value`

Attribute Value O campo `Attribute Value` representa o valor do atributo em questão.

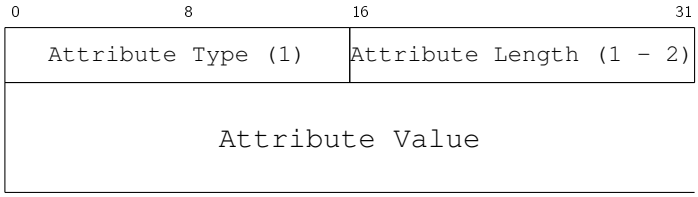

Figura 2.11: *Codificação dos atributos BGP na mensagem* `UPDATE`

O campo `Attribute Type` é composto por dois campos, como ilustra a figura 2.12. O primeiro campo, **Attribute Flags**, ocupa um octeto e especifica os diversos parâmetros do atributo. Este campo é interpretado *bit-a-bit*, onde os quatro *bits* mais significativos especificam os valores de quatro parâmetros e os restante quatro não têm uso. O significado de cada *bit*, por ordem decrescente de valor, é o seguinte:

1. O *bit* de maior valor é denominado `Optional`. Este valor especifica se o atributo é bem-conhecido[1] ou opcional. Para um atributo bem-conhecido o *bit* tem o valor 1, caso contrário tem o valor 0.

[1] *well-known*, segundo a especificação do BGP

2. O segundo *bit* de maior valor é denominado `Transitive`. Quando toma o valor de 1, define um atributo opcional e transitivo; quando toma o valor de 0, define um atributo opcional e não-transitivo. Para atributos bem-conhecidos o valor deste *bit* deve ser 0.

3. O terceiro *bit* mais significativo denomina-se `Partial`. Este *bit*, quando toma o valor de 1, indica que o atributo é opcional, transitivo e parcial. Caso seja um atributo opcional, transitivo e completo, ou qualquer um dos casos, deve tomar o valor de 0.

4. Por fim, o quarto *bit* denomina-se `Extended Length`. Este *bit* indica se o campo `Attribute Length`, referido acima, ocupa um octeto (`Extended Length` toma o valor de 0), ou dois octetos (`Extended Length` toma o valor de 1).

A figura 2.13 ilustra os *bits* acima referidos. Os atributos encontram-se des-

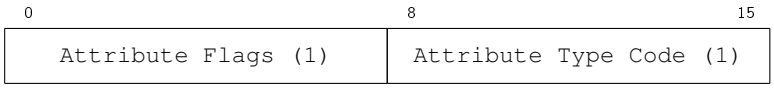

Figura 2.12: *Codificação do campo `Attribute Type` na mensagem `UPDATE`*

critos com mais detalhe na secção 2.4.3, no entanto segue-se uma descrição breve dos tipos dos atributos. Os atributos podem ser `well-known`, significando que devem ser reconhecidos por todas as máquinas, ou `optional`, significando que podem não ser reconhecidos. Um atributo `well-known` pode ainda ser obrigatório – `mandatory` – ou não – `discretionary`. Os atributos `optional` podem ainda ser `transitive`, implicando que deve persistir associado a uma rota, mesmo não sendo reconhecido, ou `non-transitive`, significando o oposto. Um atributo `optional transitive` diz-se `partial`

quando, a longo de uma rota, houve um encaminhador que não o reconheceu.

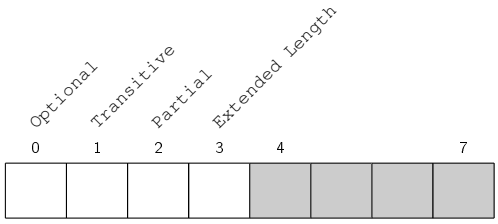

Figura 2.13: *Codificação do campo* `Attribute Flags` *na mensagem* `UPDATE`

2.4.1.3 Mensagem `NOTIFICATION`

A mensagem `NOTIFICATION` é enviada quando ocorre um erro na sessão BGP de um par. Após a ocorrência do erro, o par envia uma mensagem de erro – `NOTIFICATION` – com a informação relativa ao mesmo. Uma mensagem `NOTIFICATION` pode reportar:

- mensagens recebidas com erro;
- temporizadores que expiraram desde a ultima mensagem recebida;
- erro na máquina de estados BGP;
- fim de sessão, sem nenhum erro associado.

A figura 2.14 ilustra o formato da mensagem `NOTIFICATION`. A mensagem `NOTIFICATION` é composta pelos seguintes campos:

`Error Code` O campo `Error Code` especifica o tipo de erro encontrado, correspondendo cada tipo de erro a um código que será exposto de seguida.

Error Subcode O campo `Error Subcode` especifica, com maior detalhe que o campo `Error Code`, o erro que ocorreu na sessão BGP. O campo `Error Subcode` é, portanto, dependente do campo `Error Code`.

Data O campo `Data`, como o nome indica, contem toda a informação relacionada com o erro ocorrido. A presença deste campo está dependente do `Error Code` e do `Error Subcode`, pois apenas algumas combinações requerem o envio de dados adicionais.

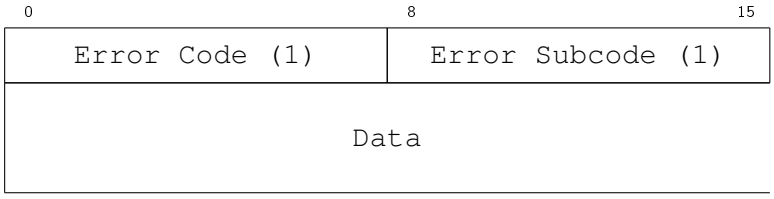

Figura 2.14: *Formato da mensagem* `NOTIFICATION`

Na especificação do BGP estão previstos seis `Error Code` diferentes, com vários `Error Subcode` associados. De Seguida expõem-se os vários tipos de erros previstos na especificação, com os valores associados.

Message Header Error (1) Verifica-se um `Message Header Error` quando surge um erro no processamento do cabeçalho comum das mensagens BGP. Este tipo de erro prevê os seguintes `Error subcode`:

Connection Not Synchronized (1) indica que o campo `Marker`, do cabeçalho comum, não corresponde ao previsto pelos mecanismos de sincronismo e segurança.

Bad Message Length (2) indica que o tamanho da mensagem é inferior ao mínimo especificado no protocolo, ou superior ao máximo suportado pela ligação entre os pares.

Bad Message Type (3) indica que o tipo da mensagem não é nenhum dos previstos — OPEN, UPDATE, NOTIFICATION ou KEEP-ALIVE.

OPEN message Header (2) O tipo de erro OPEN message Header indica um erro no processamento de mensagem OPEN. Este tipo de erro prevê os seguinte Error subcode:

Unsupported Version Number (1) indica que, o par que envia a mensagem NOTIFICATION, não suporta a versão BGP proposta pelo par que enviou a mensagem OPEN.

Bad Peer AS (2) indica que o par recebeu uma mensagem OPEN, cujo campo My Autonomous System não corresponde ao valor configurado internamente.

Bad BGP Identifier (3) indica um erro relativo ao campo BGP Identifier de uma mensagem OPEN recebida.

Unsupported Optional Parameter (4) indica que o par não suporta um parâmetro opcional proposto numa mensagem OPEN.

Authentication Failure (5) indica um erro na autenticação do par que envia a mensagem OPEN.

Unacceptable Hold Time (6) indica que o valor de Hold Time proposto por um par não é aceite pelo encaminhador que envia a mensagem NOTIFICATION.

UPDATE Message Error (3) O tipo UPDATE Message Error indica que ocorreu um erro relativo a uma mensagem UPDATE. Este tipo de erro possibilita os seguintes Error Subcode:

Malformed Attribute List (1) indica que ocorreu um erro na lista de atributos de uma mensagem UPDATE

Unrecognized Well-known Attribute (2) indica que foi recebida uma mensagem UPDATE, com um atributo identificado como bem-conhecido, mas não reconhecido pelo par.

Missing Well-known Attribute (3) indica que não constava, numa mensagem UPDATE, um atributo bem-conhecido.

Attribute Flags Error (4) indica que ocorreu um erro nos valores do campo Attribute Flags de um atributo.

Attribute Length Error (5) indica que ocorreu um erro relativo ao tamanho de um atributo.

Invalid ORIGIN Attribute (6) indica que o valor do atributo ORIGIN não corresponde a nenhum previsto na especificação do BGP.

Invalid NEXT-HOP Attribute (8) indica que o valor do atributo NEXT-HOP de uma mensagem UPDATE é invalido.

Optional Attribute Error (9) indica que ocorreu um erro no processamento de um atributo opcional.

Invalid Network Field (10) indica que ocorreu um erro a processar um prefixo de uma mensagem UPDATE.

Malformed AS_PATH (11) indica que foi recebida uma mensagem UPDATE, com um AS_PATH com erros.

Hold Timer Expired (4) Uma mensagem NOTIFICATION, com um Error Code Hold Timer Expired, é enviada quando o temporizador expira sem que o par tenha recebido qualquer mensagem do outro par.

Finite State Machine Error (5) O `Error Code Finite State Machine Error` é gerado quando acontece um evento não previsto na máquina de estados do BGP.

Cease (6) Quando ocorre um erro, não previsto pelos `Error Code` acima expostos, um encaminhador envia uma mensagem `NOTIFICATION` com o `Error Code Cease`.

2.4.1.4 Mensagem KEEPALIVE

O protocolo BGP implementa mecanismos para verificar o estado das ligações com os pares. Na troca de mensagens `OPEN`, os pares acordam o valor de um temporizador – `Hold Time`. Devem ser trocadas mensagens com um período nunca maior que o valor deste temporizador. As mensagens `KEEPALIVE` têm essa funcionalidade. Caso um par não tenha mensagens `UPDATE` que enviar, deve enviar mensagens `KEEPALIVE`, por forma a não permitir que o temporizador do par expire.

As mensagens de `KEEPALIVE` não transportam qualquer tipo de informação. O formato destas mensagens é composto apenas pelo cabeçalho comum do BGP, ilustrado na figura 2.6.

2.4.2 Armazenamento de Rotas

A especificação do protocolo BGP apresenta um modelo de como as rotas devem ser armazenadas. No entanto, nada obriga a que os sistemas implementem exactamente o modelo apresentado. Apenas é necessário que o sistema se comporte como previsto na especificação.

No modelo apresentado pela especificação, são apresentadas três tabelas de encaminhamento, sendo elas a `Adj-RIB`[2]`-In`, a `Loc-RIB` e a `Adj-RIB-Out`. O modelo descrito na especificação prevê, relativamente às tabelas de encaminhamento, o seguinte:

Adj-RIB-In A `Adj-RIB-In` contem as rotas válidas, anunciadas por um par. Deve portanto existir uma `Adj-RIB-In` associada a cada par e a cada sessão BGP activa. A validação das rotas será descrita *a posteriori*.

Loc-RIB A `Loc-RIB` corresponde à tabela de encaminhamento do par para os destinos descobertos por BGP. A composição da `Loc-RIB` resulta de um processo de selecção, aplicando as políticas encaminhamento, que tem como base as várias `Adj-RIB-In`. O processo de selecção de rotas será também descrito posteriormente no documento.

Adj-RIB-Out A `Adj-RIB-Out` contém as rotas que o encaminhador, através da aplicação das políticas de encaminhamento, anunciará a um par. Prevê-se, no modelo descrito na especificação, que exista uma implementação desta tabela para cada par com sessão BGP activa.

Os processos de aplicação de políticas de encaminhamento e selecção de rotas são fulcrais na composição das várias RIB's. No entanto, antes de descrever esses processos, é necessário um olhar sobre os atributos BGP e o seu sentido.

2.4.3 Atributos BGP

Os atributos BGP, conforme supra referido, caracterizam uma rota para um destino. Os atributos descritos na especificação base do BGP estão expostos

[2]RIB – Routing Information Base – tabela de encaminhamento

de seguida, estando ordenados pelo valor `Type Code`[3] que os representa.

1. **ORIGIN**
2. **AS_PATH**
3. **NEXT-HOP**
4. **MULTI-EXIT-DISC**
5. **LOCAL-PREF**
6. **ATOMIC-AGGREGATE**
7. **AGGREGATOR**

Os atributos BGP são divididos em duas categorias, quanto ao seu reconhecimento por parte dos encaminhadores. Os atributos podem ser:

Well-known Um atributo `Well-known` deve ser reconhecido por todos os sistemas que implementem o protocolo BGP. Sempre que um atributo deste tipo seja actualizado, o seu novo valor deve ser re-transmitido nas mensagens de `UPDATE`. Os atributos deste tipo podem ainda ser distinguidos quanto à obrigatoriedade da sua presença nas mensagens `UPDATE`. A especificação do BGP prevê as seguintes classificações para atributos `Well-known`:

mandatory Um atributo `Well-known mandatory` deve estar presente, obrigatoriamente, em todas as mensagens `UPDATE`.

discretionary Um atributo `Well-known discretionary` pode, ou não, estar presente numa mensagem `UPDATE`.

[3] O `Type Code` de um atributo é também o que representa o atributo nas mensagens `UPDATE`

Optional Um atributo `Optional` não tem, necessariamente, de ser suportado por todas as implementações do BGP. Este tipo de atributos permite ao BGP ser extendido para novas funcionalidades, sem alterar o funcionamento base do protocolo. Este tipo de atributos é ainda dividido em duas categorias, conforme o modo como um par que não reconhece o atributo o trata. Estas duas formas são:

transitive Um atributo `Optional transitive` deve ser retransmitido nas mensagens de `UPDATE` por um encaminhador que não o reconheça. Ainda neste cenário, o *bit* `Partial` referente ao atributo deve tomar o valor 1. Um atributo deste tipo pode ser adicionado por um encaminhador durante o caminho, neste caso o *bit* `Partial` referente ao atributo deve tomar o valor 1.

non-transitive Um atributo `Optional non-transitive` deve ser ignorado, e não re-transmitido, quando recebido por um encaminhado que não o reconheça. Adicionar um atributo deste tipo durante o caminho depende da implementação do atributo.

Os atributos BGP devem estar ordenados, numa mensagem BGP, por ordem crescente do valor `Type Code` referente a cada atributo. Não podem existir atributos duplicados numa mensagem `UPDATE`. Os atributos acima expostos serão de seguida descritos.

2.4.3.1 ORIGIN

O atributo `ORIGIN` é um atributo `Well-known mandatory`. Este atributo especifica a origem da informação da caminho de um prefixo. Este atributo pode tomar três valores, onde cada valor representa:

IGP (1) Uma rota descoberta por um protocolo IGP.

EGP (2) Uma rota descoberta por um protocolo EGP.

INCOMPLETE (3) Uma rota descoberta por outra forma que não IGP ou EGP – tipicamente representa rotas estáticas.

2.4.3.2 AS_PATH

O atributo AS_PATH é um atributo Well-known mandatory. Este atributo representa o caminho de AS's (AS path) que compõem a rota para um prefixo de destino. Este atributo permite detectar ciclos em termo de AS's, aplicar políticas que filtrem rotas de alguns AS's, e atribuir uma métrica – o número de AS's num AS_PATH – a uma rota. Este atributo é codificado segundo segmentos. Estes segmentos podem ser de dois tipos: AS-SET, que corresponde a um conjunto não-ordenado de AS's; ou AS-SEQUENCE, que corresponde a uma lista ordenada de AS's. Estes segmentos são codificados, no campo Attribute Value de uma mensagem UPDATE, conforme ilustra a figura 2.15, da seguinte forma:

path segment type O campo path segment type, que ocupa um octeto, especifica o tipo de segmento, tomando o valor de 1 para um AS-SET e tomando o valor de 2 para um AS-SEQUENCE.

path segment length O campo path segment length, que ocupa um octeto, indica o número de AS's que compõem o segmento em questão.

path segment value O campo path segment value, que tem tamanho variável, contém a lista dos AS's que compõem o segmento. Este

campo tem um comprimento máximo de 255 AS's, devido ao tamanho do campo path segment length. Caso seja necessário, um atributo AS_PATH pode ser composto por vários segmentos diferentes.

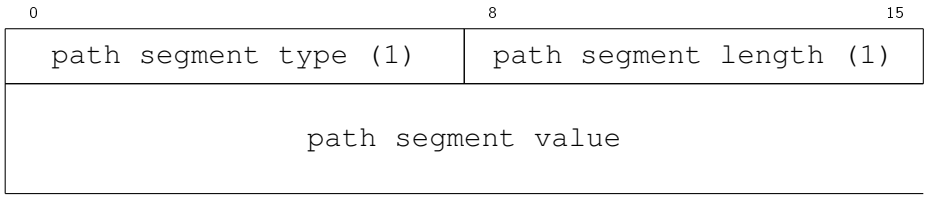

Figura 2.15: *Codificação do atributo AS_PATH numa mensagem UPDATE*

O propósito da existência de sequências de AS's não ordenadas – AS_SET – é suportar a agregação de rotas, em mensagens UPDATE, cujo AS_PATH difere em alguma parte. Desta forma a parte comum do AS_PATH é mantida como um AS-SEQUENCE e a parte não comum é concatenada num AS-SET.

2.4.3.3 NEXT-HOP

O atributo NEXT-HOP é um atributo Well-known mandatory que define o endereço do próximo salto para um destino. A codificação deste endereço é idêntica à codificação do campo NLRI e encontra-se ilustrada na figura 2.10. O NEXT-HOP tipicamente pode ter o endereço IP do par BGP, mas pode também ser um endereço diferente.

2.4.3.4 MULTI-EXIT-DISC

O atributo MULTI-EXIT-DISCRIMINATOR é um atributo Optional non--transitive. O seu valor é composto por quatro octetos. Este atributo pode ser utilizado pelo processo de decisão para discriminar pontos de entrada no AS. A figura 2.16 ilustra um exemplo onde o atributo MULTI-EXIT-DISC

pode ser utilizado. O AS 3 anuncia ao AS 4, uma rota para AS 1 pela ligação mais à esquerda, com um valor de MULTI-EXIT-DISCRIMINATOR mais reduzido, que no mesmo anuncio efectuado na ligação mais à direita. Isto deve-se ao facto de, no exemplo, o AS 1 ter um caminho mais curto através da ligação à esquerda. O atributo MULTI-EXIT-DISC deve ser utilizado em ligações BGP entre domínios diferentes.

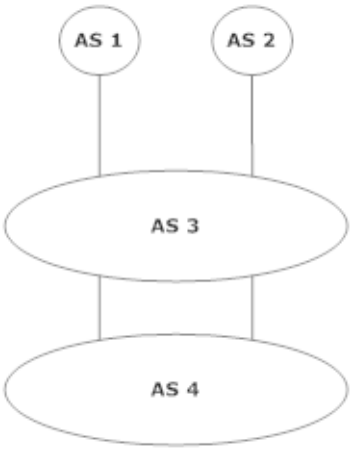

Figura 2.16: *Exemplo de uma topologia para o atributo MULTI-EXIT-DISC*

2.4.3.5 LOCAL-PREF

O atributo LOCAL-PREF é um atributo Well-known que se torna mandatory em mensagens UPDATE enviadas entre pares pertencentes ao mesmo AS. O atributo LOCAL-PREF não deve ser enviado a pares de AS's que não o AS do par que envia o UPDATE, excepto em casos especiais (RFC 5065 [16]). Este atributo é codificado com quatro octetos. Este atributo permite atribuir um grau de preferência a interfaces de saída locais para um destino.

2.4.3.6 ATOMIC-AGGREGATE

O atributo ATOMIC-AGGREGATE é classificado como Well-known discretionary e ocupa zero octetos. Este atributo deve ser incluído, sempre que se agregam rotas e é omitida informação do AS_PATH para facilitar essa agregação. A informação do AS_PATH, existindo o ATOMIC-AGGREGATE, apesar de não conter ciclos, pode não estar completa.

2.4.3.7 AGGREGATOR

O atributo AGREGATOR é classificado como Optional transitive. O valor deste atributo é composto por seis octetos, sendo eles o ASN do par que juntou o atributo e o seu BGP Identifier. Este atributo pode ser adicionado a uma mensagem UPDATE por um encaminhador que agregue rotas nessa mensagem.

2.4.4 BGP Interno e Externo

O protocolo BGP pretende dar conectividade global entre domínios. Faz portanto sentido que encaminhadores-fronteira, de AS's diferentes, tenham sessões BGP activas. No entanto devem também existir sessões activas entre os encaminhadores fronteira pertencentes ao mesmo AS. As sessões entre pares de AS's distintos denominam-se sessões de E-BGP (Exterior-BGP), ao passo que, as sessões entre pares pertencentes ao mesmo AS denominam-se de sessões I-BGP (Interior-BGP). As sessões I-BGP são o que permite, a todos os encaminhadores-fronteira de um AS, tomarem conhecimento de uma rota descoberta por um encaminhador fronteira. Recuando à figura 2.5, verifica-se uma ligação verde entre os encaminhadores R11 e R12. Esta ligação representa uma sessão de I-BGP e trata-se da forma de R12 conhecer

o AS 3 e R11 conhecer o AS 2. As sessões de E-BGP e I-BGP partilham o mesmo tipo de mensagens, no entanto o I-BGP traz umas nuances. Um encaminhador não deve anunciar a um par, de outro AS que não o seu, uma rota descoberta por I-BGP. Deve ainda existir uma sessão I-BGP entre **todos** os pares de encaminhadores-fronteira pertencentes ao mesmo AS[4].

2.4.5 Processo de Selecção de rotas

O protocolo BGP selecciona, de entre todas as rotas presentes em todas as Adj-RIB-In, as rotas que vão compor a Loc-RIB e consequentemente, serão copiadas para a Adj-RIB-Out de cada par. Este processo acontece em três fases distintas:

Fase 1 Cálculo do grau de preferência das rotas recebidas por cada par.

Fase 2 Seleccionar a melhor rota para cada destino e colocar essa rota na Loc-RIB.

Fase 3 Disseminar as rotas a cada par, de acordo com as politicas definidas no encaminhador.

Estas três fases serão descritas de seguida.

2.4.5.1 Fase 1: Cálculo do Grau de Preferência

A fase 1 tem início sempre que o encaminhador recebe uma mensagem UPDATE. Cada rota presente no campo Unfeasable Routes deve ser retirada da Adj-RIB-In do par que origina a mensagem de UPDATE. O encaminhador

[4] De facto existem estratégias que permitem contornar este facto, como por exemplo a reflexão de rotas, no entanto as ligações I-BGP não são o principal foco do trabalho.

aplica as políticas de encaminhamento para eleger as rotas válidas, por exemplo, barrando rotas de um determinado AS. Para cada rota anunciada válida o encaminhador deve calcular o grau de preferência dessa rota da seguinte forma:

- Caso a rota tenha sido descoberta por I-BGP, o valor de grau de preferência deve ser o valor do atributo `Local-Pref`
- Caso a rota tenha sido descoberta por E-BGP, o valor de grau de preferência deve ser calculado conforme as políticas de encaminhamento definidas.

É então actualizada a `Adj-RIB-In` do par em questão.

2.4.5.2 Fase 2: Selecção de Rotas

A fase 2 é invocada assim que termine a fase 1. A fase dois considera todas as `Adj-RIB-In`. Para cada destino que exista uma rota, numa `Adj-RIB-In`, é seleccionada aquela que restar da aplicação das regras abaixo expostas. Assim que numa regra apenas exista uma rota, esta é seleccionada. As regras, para seleccionar a rota para um destino são as seguintes:

1. Possua o maior grau de preferência.
2. Apresente o menor número de AS's no `AS_PATH`.
3. Quando definido, a rota que apresentar menor valor de `MULTI-EXIT-DISC`
4. Apresentar menor custo para o próximo salto, segundo o protocolo IGP.
5. Se todas as rotas forem descobertas por I-BGP deve-se ignorar este passo. Caso contrário deve-se seleccionar a rota descoberta por E-BGP, cujo `BGP Identifier` do par que a enviou seja o menor.

6. A rota descoberta por I-BGP, cujo `BGP Identifier` do par que a enviou seja o menor.

Este processo de selecção é relativamente complexo, e como foi observado, não toma nunca em consideração, exceptuando porventura na aplicação de políticas, o estado ou a capacidade da rede. As rotas seleccionadas devem então ser copiadas para a `Loc-RIB`.

2.4.5.3 Fase 3: Disseminação de Rotas

A fase 3 deve ser invocada sempre que:

- Termine a fase 2.
- Rotas na `Loc-RIB` se alterem.
- Rotas internas geradas por protocolos que não o BGP se alterem.
- Se estabeleça uma nova conexão BGP.

Nesta fase são aplicadas as políticas para decidir que rotas se podem anunciar a que pares. São depois preenchidas as `Adj-RIB-Out` de cada par.

2.5 Conclusão

O presente capítulo apresenta a temática do encaminhamento de tráfego em redes IP. Dentro desta é apresentada a necessidade do encaminhamento entre domínios distintos. Como padrão da Internet, o protocolo BGP, é fundamental para a compreensão do que é um protocolo de encaminhamento inter-domínios. O protocolo é apresentado no presente capítulo, sendo descrito o

seu comportamento, o formato das mensagens e atributos, as suas tabelas de encaminhamento e processos de selecção de rotas.

Esta descrição do protocolo BGP é a base de partida para a realização de um dos objectivos do trabalho a que se refere o presente documento – a implementação de uma bancada de trabalho com o protocolo BGP num simulador de redes.

3

Implementação do Protocolo BGP no Simulador NS-3

3.1 Introdução

O presente capítulo trata um dos objectivos do trabalho a que se trata o presente documento. Este objectivo é a implementação, num simulador de redes, do *standard de facto* para o encaminhamento inter-domínios, o protocolo BGP.

Este capítulo encontra-se dividido em duas partes. Numa primeira parte-se apresenta-se o simulador de redes optado para o trabalho e os motivos que levaram a essa mesma opção. Ainda na primeira parte, descreve-se de uma forma geral, a arquitectura do simulador e, de forma mais aprofundada, como estender o simulador a novos módulos.

Numa segunda parte do presente capítulo, é descrita a implementação do protocolo BGP no simulador em questão. Esta implementação tem a sua análise teórica no capítulo anterior – 2 – e descreve-se como é realizada a integração da implementação do BGP no simulador.

3.1.1 A escolha do Simulador NS-3

O custo, ao nível de tempo, de implementar qualquer tipo de extensão para o BGP, entendeu-se muito elevado. O desenvolvimento, em ambiente de simulação, é mais rápido, bem como o teste e obtenção de resultados.

Realizou-se uma pesquisa, na biblioteca digital do IEEE (*Institute of Electrical and Electronics Engineers*), por artigos científicos, cujo título contivesse o nome de um simulador de rede de código aberto. O simulador NS-2, versão anterior do simulador NS-3, foi aquele com, de longe, mais ocorrências – 926 artigos contêm no seu título a sequência "*NS-2 Simulator*".

O simulador NS-2, apesar de ainda largamente utilizado, em Novembro de 2012 verá passar um ano sobre o último lançamento de um versão, tendo a última sido lançada a 4 de Novembro de 2011. A versão seguinte ao NS-2, o simulador NS-3, é a versão que tem sido alvo da atenção dos programadores.

Tendo em conta o esforço realizado na introdução de uma nova versão, e também por forma a dar um contributo à comunidade académica e ao NS-3, tomou-se a decisão de implementar um modelo do protocolo BGP para o simulador NS-3. Até à data da escrita do presente documento, não existe um modelo do protocolo BGP, quer lançado oficialmente, quer na forma de contribuição por parte da comunidade académica.

3.2 O Simulador NS-3

O simulador NS-3 é composto por um conjunto de bibliotecas escritas de raiz para o propósito. O projecto arrancou a meados de 2006. As bibliotecas estão todas escritas na linguagem de programação C++, com alguns ficheiros de simulação escritos na linguagem Python. Pode-se dividir a escrita de pro-

gramas no NS-3 em dois níveis: a escrita de modelos e escrita de simulações. No primeiro caso desenvolvem-se modelos que representam tecnologias e ou protocolos de rede, como por o exemplo o IPv4, o LTE (*Long Term Evolution*, ou o 4G) ou um dispositivo de rede. No segundo caso escrevem-se ficheiro de simulação que utilizam os modelos desenvolvidos. Tipicamente, a utilização dos modelos nos ficheiros de simulação é realizada através de objectos de um tipo denominado `Helper`.

3.2.1 Os Objectos Chave do NS-3

O simulador NS-3 apresenta um conjunto de objectos chave para a escrita de ficheiros de simulação. Estes objectos correspondem a abstracções de entidades do mundo real. Os objectos chave são os seguintes:

Node Um Objecto do tipo `Node` representa um nó. De uma forma geral, contém um conjunto dos objectos chave expostos de seguida. Pode ser entendido, geralmente, como um encaminhador ou um terminal.

NetDevice Um objecto do tipo `NetDevice` representa uma interface de rede, como por exemplo uma interface *WiFi* ou *ethernet*. Um `Node` pode conter vários `NetDevice`.

Channel Entre dois `NetDevice` de um determinado tipo deve existir um `Channel` de um tipo específico. Um objecto do tipo `Channel` representa um canal de comunicação entre dois `NetDevice`. Se os `NetDevice` forem do tipo *WiFi*, o `Channel` deve também ser do tipo *WiFi*.

Packet Um objecto do tipo `Packet` representa um pacote de dados. São compostos por um conjunto de octetos contendo cabeçalhos de protocolos e dados.

Application Objectos do tipo `Application` representam processos (aplicações) que geram ou recebem tráfego usando a rede implementada na simulação. Geralmente estão associadas a um objecto do tipo `Node`.

Socket As ligações da camada de transporte são representadas, como de costume, através de objectos do tipo `Socket`. Estes objectos, que, geralmente, se associam a um `NetDevice`, possuem um conjunto de funções para envio e recepção de pacotes que são escalonadas e executadas por ordem de escalonamento. Estas funções denominam-se `Callback`.

3.2.2 Escalonador de Eventos

Com qualquer simulação do NS-3 corre um escalonador de eventos. Este este escalonador é responsável pela execução de qualquer processo no NS-3. Ao contrário de implementações reais, onde quem escreve o código deve permitir e controlar múltiplos processos e interacção entre estes, no NS-3 os processos são definidos como eventos e escalonados e executados sequencialmente. Deste comportamento vem a denominação do NS-3 como um simulador baseado em eventos.

Um evento, ao ser escalonado, deve ter associado um tempo para execução e uma função associada. Estas funções associadas são, geralmente, objectos do tipo `Callback`. Os processos referentes a alguns eventos podem assim ser modificados através da alteração da `Callback` associada.

3.2.3 O Encaminhamento no NS-3

O protocolo NS-3, implementa vários protocolos de encaminhamento. Entre estes protocolos, na sua maioria protocolos de encaminhamento de redes *ad-hoc*, encontra-se um protocolo de encaminhamento global centralizado. Este protocolo preenche as tabelas de encaminhamento dos encaminhadores consoante o estado da rede.

O simulador NS-3 permite a incorporação de novos protocolos de encaminhamento. Estes protocolos de encaminhamento devem ser implementados como sub-classes da classe abstracta `Ipv4RoutingProtocol`. A figura 3.1 ilustra como se realiza o encaminhamento de tráfego, através de um `Ipv4RoutingProtocol`.

O NS-3 especifica uma camada entre o protocolo de encaminhamento, `Ipv4RoutingProtocol` e os `NetDevice` de um `Node`. Esta camada é simulada através de objectos da classe `IPv4L3Protocol`. Uma instância de `IPv4L3Protocol` delega as decisões de encaminhamento num `Ipv4RoutingProtocol`.

Um `Ipv4RoutingProtocol` deve implementar duas funções para o encaminhamento de tráfego:

`RouteOutput` A função `RouteOutput` é invocada pelos protocolos de transporte. Esta função retorna um rota para um destino, contendo uma interface de saída e o endereço do próximo salto. O protocolo de transporte, com a informação da rota, invoca a função `SendOut` do `Ipv4RoutingProtocol` associado ao `Node`.

`RouteInput` A função `RouteInput` é invocada pelo `Ipv4RoutingProtocol` assim que um `NetDevice` receba algum pacote. Esta função

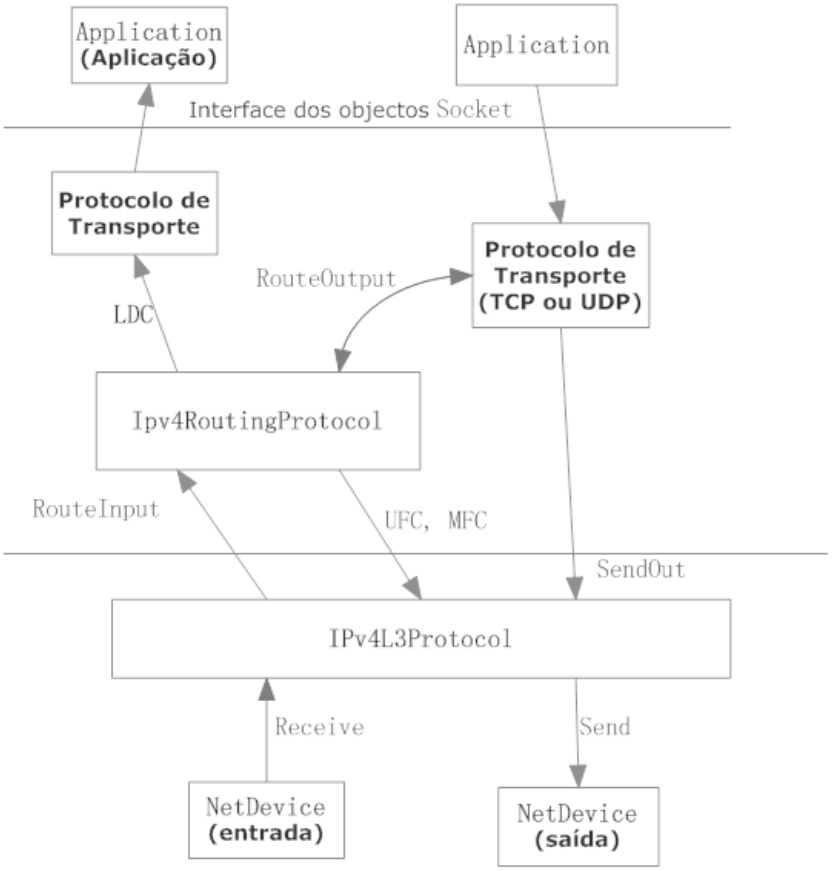

Figura 3.1: *Modelo do encaminhamento de tráfego no simulador NS3*

encaminha cada pacote usando quatro `Callback` que lhe são passadas como parâmetros. Cada uma destas `Callback` representa um encaminhamento diferente, pelo que a função decide para que `Callback` vai encaminhar o pacote. As `Callback` em questão podem ser as seguintes:

Encaminhamento Local Se o pacote se destina ao `Node` em questão, a função `RouteInput` deve invocar a `LocalDeliveryCallback`

(LDC, na figura 3.1), com o pacote recebido. Esta `Callback` permite aos dados chegarem à camada de transporte que, eventualmente, os entregará numa `Application`.

Encaminhamento Unicast Se o pacote se destina a um endereço de um `Node` que não o nó em questão, deve ser invocada a `UnicastForwardCallback` (UFC, na figura 3.1). A função `RouteInput` deve associar a rota que o pacote deve seguir e associar à `UnicastForwardCallback` o pacote e a rota.

Encaminhamento Multicast Caso o pacote seja de tráfego *multicast*, a função `RouteInput` deve, da mesma forma que o encaminhamento *Unicast*, associar a rota do pacote e o pacote à `Callback` que trata este caso – `MulticastForwardCallback` (MFC, na figura 3.1).

A implementação dos tipos de encaminhamento é da responsabilidade do `Ipv4RoutingProtocol`. Se, por exemplo, se implementar um protocolo que não permita o encaminhamento *Multicast*, a função `RouteInput` deve ignorar a recepção de pacotes deste tipo. Ambas as funções acima referidas obtêm as rotas para um destino através de consultas a tabelas de encaminhamento que, no caso de uma nova implementação de um protocolo, devem também ser implementadas.

3.2.4 O Encaminhamento com Vários Protocolos – `Ipv4ListRouting`

O NS-3 permite que possam existir vários protocolos de encaminhamento num `Node`. Desta forma existe a classe `Ipv4ListRouting` que permite associar

vários `Ipv4RoutingProtocol` definindo uma prioridade a cada um. Estes `Ipv4RoutingProtocol` são invocados, por ordem crescente de prioridade, para calcular rotas para um destino. Assim que se obtenha uma rota o processo é terminado sem percorrer todos os `Ipv4RoutingProtocol`.

3.2.5 Objectos do Tipo `Header`

O simulador NS-3 implementa a classe abstracta `Header` que permite a incorporação e extracção de dados em cabeçalhos. Esta classe permite que se especifiquem sub-classes referentes ao cabeçalho de um protocolo. Desta forma, existe uniformidade na criação e leitura dos parâmetros dos pacotes que percorrem a rede. Cada sub-classe deve implementar os métodos de serialização e desserialização os cabeçalhos.

3.3 Implementação do Protocolo BGP

O protocolo BGP, no simulador NS-3, deve ser implementado como um objecto da classe `Ipv4RoutingProtocol`. Pelo exposto nas anteriores secções do presente capítulo, entende-se que a implementação do protocolo BGP no simulador NS-3 divide-se em três fases:

- Implementação das mensagens BGP, tendo em conta os formatos especificados em [14].
- Especificação das tabelas de encaminhamento – as `RIB`, conforme [14].
- Implementação do protocolo BGP como um objecto do tipo `Ipv4RoutingProtocol` que tenha um comportamento semelhante ao apontado em [14].

Como referido acima, a implementação destas fases é realizada recorrendo à linguagem de programação C++.

3.3.1 Implementação das Mensagens BGP

As mensagens BGP, descritas no capítulo 2.4.1, são a forma que os pares têm de comunicar entre si. Existem três tipos de mensagens:

1. `OPEN`
2. `UPDATE`
3. `NOTIFICATION`
4. `KEEPALIVE`

Como referido nas secções anteriores do presente capítulo, os cabeçalhos das mensagens são definidos como sub-classes da classe abstracta `Header`. Esta decisão baseia-se no facto de as mensagens BGP serem compostas apenas por um cabeçalho e dados da mensagem.

Para implementar as mensagem BGP no NS-3 definiram-se as seguintes subclasses da classe abstracta `Header`:

`BgpOpenHeader` A classe `BgpOpenHeader` encontra-se implementada no ficheiro `bgp-open-packet.cc`.

`BgpUpdateHeader` A classe `BgpUpdateHeader` encontra-se implementada no ficheiro `bgp-update-packet.cc`.

`BgpNotificationHeader` A classe `BgpNotificationHeader` encontra-se implementada no ficheiro `bgp-notification-packet.cc`.

BgpKeepAliveHeader A classe `BgpKeepAliveHeader` encontra-se implementada no ficheiro `bgp-keepalive-packet.cc`.

Cada uma das classes acima definidas apresenta um método construtor, que recebe como parâmetros os vários valores das mensagens, que é utilizado para criar as entidades dos objectos. Cada classe implementa, também os métodos `Serialize()` e `Deserialize()` que permitem transformar um objecto num conjunto de octetos e o contrário respectivamente.

3.3.1.1 Atributos das Mensagens `UPDATE`

As mensagens `UPDATE`, contrariamente aos restantes tipos de mensagens, apresentam uma construção que pode ser bastante variada, devido aos atributos BGP. Os atributos são codificados nas mensagens BGP de uma só forma, no entanto o campo referente ao valor de cada atributo tem uma codificação própria. Desta forma entendeu-se implementar os vários atributos como objectos de uma classe `BgpAttribute`, estando esta classe implementada no ficheiro `bgp-attribute.cc`.

Um objecto da classe `BgpAttribute` tem um construtor vazio que cria o objecto. Os vários tipos de atributos são então criados com uma função específica do atributo que o transforma, recebendo o seu valor como parâmetro, num atributo desse tipo. Como exemplo para estratégia descrita, se se pretender criar um atributo `ORIGIN`, deve-se criar um objecto `BgpAttribute` com o construtor vazio e de seguida invocar o método `NewOrigin`, enviando-lhe o código respectivo à origem, no objecto. No caso do atributo `AS_PATH`, a construção é ligeiramente diferente dos restantes atributos.

O atributo AS_PATH é um atributo composto por mais que um campo, como descrito na secção 2.4.3.2. Desta forma, para representar o atributo AS_PATH criaram-se três classes, implementadas no ficheiro bgp-as-path.cc:

BgpAsSet A classe BgpAsSet representa um AS-SET que compõem um atributo AS_PATH.

BgpAsSequence A classe BgpAsSequence representa um AS-SEQUENCE que compõem um atributo AS_PATH.

BgpAsPath A classe BgpAsPath é composta por vectores de instâncias de BgpAsSequence e BgpAsSet e representa um atributo AS_PATH.

3.3.2 Tabelas de Encaminhamento BGP

A implementação do protocolo BGP no simulador NS-3 contém a implementação das diversas tabelas previstas na norma. Como referido no capítulo 2.4.2, existem três tabelas definidas na norma do BGP:

- Adj-RIB-In
- Loc-RIB
- Adj-RIB-Out

De uma forma geral, o que difere entre as tabelas referidas, é a quantidade de tabelas – existem várias Adj-RIB-In e Adj-RIB-Out e apenas uma Loc-RIB – e as rotas nestas contidas – todas as rotas válidas estão contidas nas Adj-RIB-In, ao passo que apenas as seleccionadas estão contidas nas Loc-RIB e Adj-RIB-Out. Desta forma, optou-se por implementar apenas uma tipo de tabela, sendo que várias instâncias deste tipo terão significados diferentes no protocolo de encaminhamento.

Para implementar as tabelas de encaminhamento desenvolveram-se duas classes:

BgpRibEntry A classe `BgpRibentry` encontra-se implementada no ficheiro `BgpRib.cc`. Esta classe representa uma entrada na tabela de encaminhamento. Cada entrada é composta por um prefixo de destino – endereço do destino e máscara de rede – e os atributos BGP associados a cada rota.

BgpRib A classe `BgpRib` encontra-se implementada no ficheiro `BgpRib.cc`. Esta classe representa uma RIB, pois é composta por uma tabela de objectos `BgpRibEntry`, onde o campo correspondente ao prefixo do destino é chave nesta tabela.

Entre as várias funções que compõem a classe `BgpRib`, apresentam-se as seguintes:

Print A função `Print` imprime os valores que compõem a tabela de encaminhamento.

GetUpdateMessage A função `GetUpdateMessage` permite compor os campos `Withdrawn Routes` e `NLRI` de uma mensagem Update

Lookup A função `Lookup`, como expectável numa tabela de encaminhamento, permite descobrir o próximo salto para um destino. Esta procura pelo destino é realizada percorrendo todas as entradas e procurando a melhor correspondência.

3.3.3 O protocolo de Encaminhamento

A modelação do comportamento do protocolo BGP, com referido acima, é realizada através da implementação de uma sub-classe da classe abstracta `Ipv4-RoutingProtocol`. Esta subclasse denominou-se `BgpRoutingProtocol` e encontra-se implementada no ficheiro `bgp-routing-protocol.cc`.

3.3.3.1 Funções Implementadas da Classe Abstracta `Ipv4Routing-Protocol`

Uma sub-classe da classe abstracta `Ipv4RoutingProtocol` deve implementar, no mínimo os métodos `RouteOutput` e `RouteInput`, por forma permitir o encaminhamento de tráfego, como se ilustra na figura 3.1. No entanto a classe abstracta `Ipv4RoutingProtocol` permite que se implementam mais funções que são invocadas pelo simulador na ocorrência de determinados eventos. Segue-se uma descrição geral das funções, deste tipo, implementadas na classe `Ipv4RoutingProtocol`.

`RouteOutput` A função `RouteOutput` retorna uma rota para um destino, após invocar a função `Lookup` na instância de `BgpRIB` correspondente à `Loc-RIB`.

`RouteInput` A função `RouteInput` processa um pacote recebido entregando-o à camada superior, ou encaminhando-o por `uincast` após consultar a `Loc-RIB`.

`NotifyInterfaceUp` A função `NotifyInterfaceUp` é invocada pelo escalonador assim que uma interface de um `NetDevice` fica activa. Na classe `BgpRoutingProtocol` esta função cria uma conexão TCP com os pares aos quais tem conectividade. Após estabelecida a conexão TCP

precede-se à troca de mensagens `OPEN` e posteriormente `UPDATE`. Esta função despoleta o protocolo BGP numa interface.

NotifyInterfaceDown A função `NotifyInterfaceDown` é invocada pelo escalonador sempre que uma interface transita de um estado activo para inactivo. Esta função, se possível termina as conexões TCP com os pares e elimina das RIB as rotas descobertas pelos pares com os quais se perdeu conectividade. Caso se perca conectividade, ou se alterem os atributos, de uma rota, essa informação é registada para compor o campo `WithdrawnRoutes` de uma mensagem `UPDATE`.

NotifyAddAddress A função `NotifyAddAddress` é invocada sempre que exista uma associação de um endereço IP numa interface. Esta função, quando invocada, adiciona o endereço associado à tabela de encaminhamento para poder anunciá-lo aos pares.

NotifyRemoveAddress A função `NotifyRemoveAddress` é invocada sempre que exista um endereço IP seja desassociado de uma interface. A entrada desse endereço é retirada da tabela e regista-se essa informação para compor o campo `WithdrawnRoutes` de uma mensagem `UPDATE`.

3.3.3.2 Estruturas de Dados

A classe `BgpRoutingProtocol` implementa um conjunto de estruturas de dados, por forma a conter toda a informação necessária ao protocolo BGP. As principais estruturas de dados, implementadas na referida classe, são as seguintes:

m_peerInterface A estrutura `m_peerInterface` contém a informação do endereço dos pares que se encontram ligados a um determinado in-

terface. É composto por uma lista de tuplos com o código da interface e o endereço IP do par que está atingível por essa interface. Esta informação deve ser passada ao protocolo na simulação através da função `AddPeer`.

m_socketList A estrutura `m_socketList` representa a lista dos objectos do tipo `Socket` associados a cada interface. Cada entrada desta lista é composta por um tuplo com o código da interface e um apontador para um objecto do tipo `Socket`.

m_activePeers A estrutura `m_activePeers` representa a informação dos pares com os quais se tem uma ligação BGP activa. É composta por uma lista de tuplos com o endereço IP dos pares e um estado da ligação – em início, estabelecida ou com erros.

m_unfeasable A estrutura `m_unfeasable` representa rotas que vão compor o campo `Withdrawn Routes` de uma mensagem `UPDATE`. Esta estrutura é composta por uma lista de prefixos IP.

m_adjRibIn A estrutura `m_adjRibIn` representa, como o nome indica, as várias `Adj-RIB-In` do BGP. A estrutura é composta por uma lista de tuplos com o endereço IP de um par e um objecto do tipo `BgpRib`.

m_adjRibOut A estrutura `m_adjRibOut` é a correspondente da estrutura `m_adjRibIn` para as várias `Adj-RIB-Out`. A composição é idêntica à da estrutura acima descrita.

m_locRib A estrutra `m_locRib` representa, como o nome indica, a `Loc-RIB` do protocolo. É composta por um objecto do tipo `BgpRib`.

3.3.3.3 Comportamento Geral da Implementação

O objectivo da implementação do protocolo BGP no simulador NS-3 é obter um comportamento, o mais semelhante possível, ao especificado na norma do BGP. O comportamento do protocolo BGP tem de ser adaptado para os eventos do NS-3. Desta forma apresentam-se, de uma forma muito geral, os eventos implementados e a sequência entre estes.

SendOpenMsg O evento `SendOpenMsg` corresponde a enviar uma mensagem `OPEN` para um par. Este evento accionado sempre que um par inicia o estabelecimento de uma ligação TCP, ou recebe uma mensagem `OPEN`, numa conexão TCP que não iniciou.

RecvOpen O evento `RecvOpen` corresponde a receber uma mensagem `OPEN`, é invocado sempre que:

1. Um par inicia uma ligação TCP e executa `SendOpenMsg`.
2. Um par tem uma conexão estabelecida que não a iniciou.

StartUpdating O evento `StartUpdating` corresponde a enviar o primeiro conjunto de mensagens `UPDATE`, numa sessão já activa. Neste evento criam-se as instâncias das estruturas de dados relativas à sessão BGP. Este evento é invocado a recepção ou envio de uma mensagem `OPEN`.

RecvUpdate O evento `RecvUpdate` corresponde a receber uma mensagem do tipo `UPDATE`. Neste evento, a mensagem é processada e a informação da `Adj-RIB-In` referente ao par que envia a mensagem é actualizada. Após este evento inicia-se o evento `DoPCalc`.

DoPCalc O evento `DoPCalc` refere-se ao cálculo do grau de preferência das rotas de uma `Adj-RIB-In`. Este evento é escalonado sempre que ocorre um `RecvUpdate`. Este evento termina com a invocação de um evento `RouteSelection`.

RouteSelection O evento `RouteSelection` corresponde à segunda fase do processo de selecção do BGP. Este evento é invocado sao terminar o evento acima descrito. Neste evento são seleccionadas as rotas das `Adj-RIB-In` referente a cada par para a `Loc-RIB`. Ao terminar este evento, invoca-se o evento `RouteDissemination`.

RouteDissemination O evento `RouteDissemination` corresponde à fase 3 do processo de selecção do BGP. Neste evento são preenchidas as `Adj-RIB-Out` de cada par e construídas as mensagens `UPDATE` a enviar. Este evento invoca o evento `SendUpdate`. Este evento é invocado pelos eventos `StartUpdating`, `RouteSelection` e na função `NotifyAddAddress`.

SendUpdate O evento `SendUpdate` corresponde a enviar uma mensagem de `UPDATE`. Este evento é invocado pelo evento `RouteDissemination`. Este evento tem um temporizador interno, por forma a não enviar mensagens com uma frequência superior à fixada.

SendNotification O evento `SendNotification` corresponde ao envio de uma mensagem `NOTIFICATION`. Sempre que ocorre um erro num dos eventos acima, invoca-se este evento. Este evento cria a mensagem `NOTIFICATION` e termina a ligação BGP com o par em questão. Após este evento, transita-se para o estado inicial do BGP.

SendKeepALive O evento `SendKeepALive` é invocado sempre que um temporizador expira sem que se esteja no estado `SendUpdate`. Este evento corresponde ao envio de uma mensagem `KEEPALIVE`.

3.4 Teste da Implementação

O simulador NS-3 tem conjunto razoável de sub-classes da classe `Application`. Estas sub-classes referem-se a aplicações que têm como objectivo testar os modelos de rede implementados e as tecnologias a que estes se referem. Para testar o protocolo BGP equacionou-se a criação de uma aplicação, no entanto, decidiu-se utilizar aplicações já existentes no simulador NS-3, pois entendeu-se que estas cumpriam os requisitos dos testes.

Criou-se um ficheiro de simulação onde se implementou a topologia ilustrada na imagem 3.2. A topologia contém sete sistemas autónomos e treze encaminhadores, o que faz com que existam ligações I-BGP e E-BGP.

A aplicação escolhida para o teste foi a `OnOffApplication`. Esta aplicação gera tráfego para um destino com uma determinada frequência. Foi instanciado um objecto da sub-classe `OnOffApplication` no interface, com endereço IP `10.1.1.1`, de um encaminhador do AS 65000, a vermelho na figura 3.2. Definiu-se que o destino do tráfego seria o interface `10.10.1.1` de um encaminhador do AS 650006, a azul na figura 3.2.

Os resultados foram medidos através da ferramenta *Wireshark*. A ferramenta permite obter amostras do tráfego entre nós da rede implementada. A figura 3.3 ilustra uma amostra do tráfego que percorreu a simulação. Pode-se verificar que existiu uma conexão TCP entre os interfaces `10.1.1.1` e `10.1.1.2` – a verde na figura 3.2. Após a conexão TCP estar estabelecida, existiu uma troca de mensagens `OPEN` e logo após a troca das mensagens

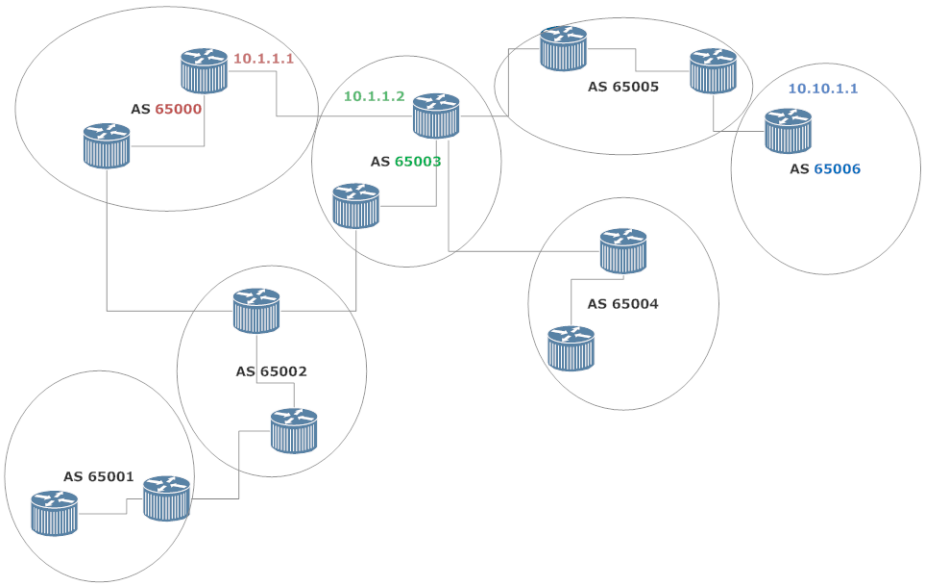

Figura 3.2: *Topologia de rede usada no teste à implementação do BGP no NS-3*

`UPDATE`. Esta troca de mensagens permitiu a `10.1.1.1` ter conhecimento de uma rota para `10.10.1.1`, começando dessa forma o tráfego da aplicação `OnOffApplication`.

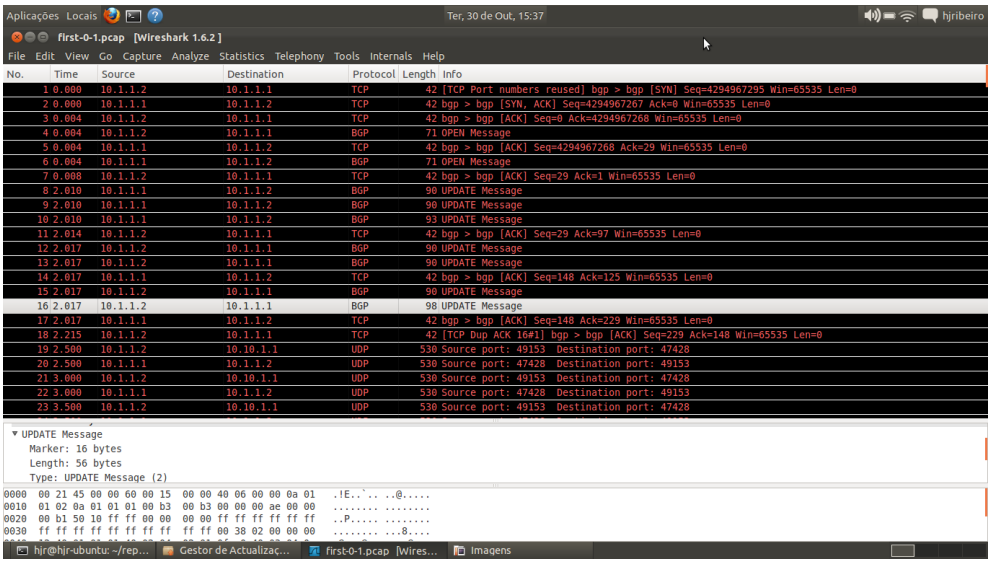

Figura 3.3: *Visualização da simulação do teste no programa* Wireshark

Os resultados permitem classificar o comportamento do implementação do protocolo BGP para o simulador NS-3 como adequado ao especificado na norma.

3.5 Conclusão

No presente capítulo descreveu-se a implementação do protocolo BGP no simulador NS-3 e como esta se integra no mesmo. É descrita, de uma forma geral, a arquitectura do NS-3, e principalmente, como é realizado o encaminhamento neste simulador e como o encaminhamento pode ser estendido.

Até à data da escrita do presente documento, não existia, uma implementação do BGP no NS-3. Esta lacuna, associada ao facto de o NS-3 fornecer um suporte mais completo para simulação de redes – por exemplo, ao nível

do endereçamento – que o seu antecessor, NS-2, foi o principal móbil para a implementação do protocolo no simulador em questão.

Pelos vários testes efectuados, entre eles o teste descrito na secção 3.4, entende-se que a implementação do BGP tem um comporto de acordo com o expectável e descrito na norma do protocolo.

4

Encaminhamento Inter-Domínio com QoS

"To enable end-to-end quality of service guarantees in the Internet, based on the border gateway protocol(BGP), interdomain QoS routing information advertising, and routing are important. However, little research has been done in the area so far." [18]

4.1 Introdução

No presente capítulo apresentam-se o estudo do trabalho relacionado com o encaminhamento inter-domínio com informações de QoS. Pelo referido no primeiro capítulo, o trabalho a que se refere o presente documento, tem como foco esta mesma temática. Pretende-se neste capítulo apresentar uma visão global do estado do encaminhamento entre domínios com QoS.

O presente capítulo divide-se em quatro partes. Numa primeira parte apresentam-se os conceitos relativos a qualidade de serviço. Estes conceitos incluem as métricas e estratégias usadas, bem como a sua aplicação prática. Nesta primeira parte introduz-se o encaminhamento com informações de qua-

lidade de serviço e apresenta-se um caso particular deste – o encaminhamento inter-domínio com QoS.

Numa segunda parte do presente capítulo apresenta-se trabalhos relacionados, sobre a utilização de classes de serviço como solução para o problema em questão. Estes trabalhos apresentam cinco extensões ao protocolo BGP, sendo as extensões apresentadas neste capítulo.

Na terceira parte do presente capítulo apresenta-se uma proposta de uma extensão ao protocolo BGP, por forma a permitir, também o encaminhamento com classes de serviço. A proposta em questão foca-se nos aspectos das métricas e na extensão aos atributos do BGP.

Por fim, na última parte do presente capítulo apresenta-se um trabalho sobre a utilização de métricas estatísticas para representar parâmetros de QoS. Nesta parte é apresentada a definição destas métricas, que influência têm no encaminhamento inter-domínio e como permitem lidar com heterogeneidade das redes inter-domínio.

4.2 Qualidade de Serviço

> *"Corollary of Moore's Law: As you increase the capacity of any system to accommodate user demand, user demand will increase to consume system capacity."* [18]

O tráfego no IP, geralmente, é encaminhado pacote-a-pacote, tendo em consideração o endereço IP. No entanto existem diferentes fluxos, com requisitos diferentes, que são sempre encaminhados da mesma forma. Uma aplicação de correio electrónico tem requisitos diferentes de uma aplicação de videoconferência. Estes requisitos diferentes permitem especificar a Qualidade de Serviço (QoS - *Quality of Service*) de cada aplicação. A tabela 4.1 indica a

importância de alguns parâmetros de QoS em fluxos de algumas aplicações sobre a Internet. Os parâmetros de QoS podem os mais variados, como por exemplo:

- largura de banda,
- atraso dos pacotes,
- número de pacotes perdidos,
- número de pacotes com erros, etc.

Estes requisitos de QoS são críticos, pois, por exemplo, se o atraso numa comunicação de voz for demasiado elevado, a comunicação torna-se impraticável. Com o encaminhamento realizado pacote-a-pacote, não é muito fácil para a rede, se não mesmo impossível, satisfazer os requisitos de QoS para as aplicações.

Tabela 4.1: *Importância de parâmetros de QoS em Aplicações sobre a Internet*

	Fiabilidade	Atraso	Largura de Banda
Correio Electrónico	Elevada	Reduzida	Reduzida
Transferência de Ficheiros	Elevada	Reduzida	Média
Video-conferência	Reduzida	Elevada	Elevada

Uma solução para suportar todos os requisitos de QoS é aumentar a capacidade da rede. Se existir uma rede com um capacidade bastante elevada, ao nível do atraso, da largura de banda e do número de pacotes perdidos, essa rede deve poder suportar os requisitos de QoS, desde que a utilização na rede na aumente também. No entanto, prevê-se que, durante os próximos anos, o tráfego na Internet continue a duplicar de ano para ano [2]. Esta tendência de crescimento no tráfego na Internet coloca entraves ao sobre-dimensionamento

das redes, para suportar os parâmetros de QoS exigidos pelas aplicações. Implementar mecanismos, aos vários níveis da pilha protocolar da Internet, que permitam suportar os requisitos de QoS é, no mínimo, uma solução que deve ser equacionada [5].

4.2.1 Mecanismos de QoS

O sobre-dimensionamento das redes, já referido acima, pode-se considerar um mecanismo para alcançar melhor QoS. No entanto existem outros mecanismos [5]. Estes mecanismos incluem algoritmos de controlo de filas, algoritmos de escalonamento com QoS, algoritmos de congestionamento, e operam nos encaminhadores, ou na entrada de pacotes, ou na saída. Estes mecanismos são utilizados por duas arquitecturas distintas, que pretendem incorporar o QoS nas redes IP. As duas arquitecturas referidas são as seguintes:

Serviços Integrados A arquitectura de serviços integrados, ou IntServ, baseia-se na reserva de recursos de fim-a-fim.

Serviços Diferenciados A arquitectura de serviços diferenciados, ou DiffServ, baseia-se, como o nome indica, na diferenciação do tráfego por classes.

4.2.1.1 Serviços Integrados

A arquitectura de serviços integrados baseia-se na reserva de recursos de fim-a-fim. O protocolo RSVP (The Resource reSerVation Protocol - RFC 2205 [19] e outros, RFC 2210 [20]) tem como objectivo permitir essa reserva. Todos os encaminhadores, ao longo do percurso de fim-a-fim, devem suportar esta arquitectura e o protocolo RSVP. As aplicações são responsáveis por reservar e

especificar as características do tráfego que vão gerar, bem como os recursos que pretendem reservar.

Esta arquitectura fornece bom QoS a um ou mais fluxos, pois os recursos requisitados são realmente alocados para um fluxo. No entanto é necessário que se efectue a reserva anterior à troca de informação, troca essa que não é muito escalável. O facto de cada encaminhador manter, no mínimo, um entrada na tabela de encaminhamento, por cada fluxo, também não é muito escalável. A necessidade de todos os encaminhadores suportarem a arquitectura é por si só, também, limitativa, pois pode nunca ser possível descobrir um caminho com QoS de fim-a-fim.

4.2.1.2 Serviços Diferenciados

A arquitectura de serviços diferenciados (RFC 2474 [21], RFC 2475 [22], entre outros) baseia-se na diferenciação do tráfego através da atribuição de classes de tráfego. O tráfego é classificado, conforme as necessidades de QoS, num número conhecido de classes de tráfego. O tráfego deve ser marcado com uma determinada classe. Os protocolos Ipv4 e Ipv6 especificam, nos seus cabeçalhos, campos para definição da classe de tráfego. O encaminhamento, nesta classe, é efectuado conforme a classe de serviço, e não pelo fluxo, como acontece nos serviços diferenciados. Existem duas abordagens que são mais utilizadas actualmente, sendo elas:

Expedited Forwarding (EF) O perfil EF (RFC 3246 [23]) requer valores reduzidos de atraso, variação do atraso, e perdas de pacotes. Este tráfego é bastante policiado, pois causa deterioramento da qualidade de serviço no restante tráfego. Geralmente limita-se este tráfego a 30% da capacidade máxima de uma ligação.

Assured Forwarding (AF) O perfil AF (RFC 2597 [24] e RFC 3260 [25]) tem como objectivo permitir fornecer garantias de entrega de pacotes, com determinados requisitos de QoS, desde que o número de pacotes não ultrapasse um valor estabelecido. Este perfil especifica quatro diferentes prioridades e três probabilidades de descarte de pacotes. Combinando estes valores obtêm-se doze diferentes classificações para o tráfego.

4.2.2 Encaminhamento com QoS

Os dois mecanismos apresentados acima – reserva de recursos (IntServ), ou classes de serviço (DiffServ) – introduzem qualidade de serviço nos encaminhadores. No entanto, os protocolos de encaminhamento típicos ignoram diferentes requisitos de QoS [26]. As decisões para o melhor caminho são feitas tendo em conta o número de saltos, ou políticas de encaminhamento. Este facto faz com que possa existir um caminho, capaz de fornecer os requisitos de QoS necessários a uma aplicação, mas que não pertença à tabela de encaminhamento de um encaminhador, em favor de um caminho que apenas tem um menor número de salto, mas atraso e perdas consideravelmente maiores. Para um melhor funcionamento das estratégias definidas acima, é necessário haver uma descoberta de rotas consciente das diferentes necessidades das aplicações. Esta consciência, para além de suportar melhor os mecanismos de QoS, permite também melhorar a performance de uma rede, apesar de introduzirem um acréscimo na computação dos protocolos [27]. Estes protocolos de encaminhamento com informações de QoS devem introduzir o mínimo de alterações ao actual esquema de encaminhamento IP, até por forma a facilitar a introdução dos mesmos protocolos [26].

4.2.2.1 Métricas de QoS

Por forma a existirem protocolos conscientes das necessidades de QoS das aplicações, é necessário definir as medidas em que essas necessidades são expressas. Por métricas de QoS entende-se isso mesmo. Medidas que se referem a atributos que representam o estado da rede. Estas métricas podem ser das mais variadas, desde a largura de banda instantânea disponível, ao atraso médio, *etc*. Numa forma geral, um protocolo de encaminhamento com informações de QoS deve calcular o melhor caminho, segundo uma ou mais métricas de QoS.

As métricas de QoS são divididas em dois conjuntos: estáticas e dinâmicas. As métricas estáticas são constantes e podem representar, por exemplo, a capacidade de transmissão da ligação ou o número de AS's que compõem o caminho. As métricas dinâmicas, variam de acordo com a carga e podem representar a largura de banda disponível ou até atraso médio de pacotes.

Os protocolos de encaminhamento com informações de QoS calculam o QoS de uma rota. A natureza das métricas, muitas vezes, é um entrave a estes protocolos. Pode existir mais que uma métrica nos requisitos, por exemplo largura de banda e atraso, neste caso, o calculo do melhor caminho pode resultar em rotas ambíguas. A natureza dinâmica e altamente dependente do tráfego, também é um entrave à estabilidade dos protocolos. É também necessário combinar métricas de ligações para calcular a métrica de um conjunto de ligações. As métricas apresentam formas de associação diferentes e são classificadas segundo essa associação:

Aditivas Entende-se por métricas aditivias, as métricas cuja associação resulta da soma aritmética das métricas referentes a cada ligação. Exemplos

deste tipo tipo de métrica são o atraso e a variação dos atrasos[1]. De uma forma geral, a métrica resultante da associação, $m_T = m_0 \oplus m_1 \oplus ... \oplus m_n$, onde m_T representa a resultante e m_i representa a métrica referente a cada ligação, é dada pelo representado na equação 4.1.

$$m_T = m_0 + m_1 + ... + m_n \tag{4.1}$$

Multiplicativas Métricas multiplicativas são métricas cuja associação resulta da multiplicação das métricas referentes a cada ligação. De uma forma geral, a métrica resultante da associação, $m_T = m_0 \oplus m_1 \oplus ... \oplus m_n$, onde m_T representa a resultante e m_i representa a métrica referente a cada ligação, é dada pelo representado na equação 4.2.

$$m_T = m_0 \times m_1 \times ... \times m_n \tag{4.2}$$

A taxa de pacotes perdidos é um exemplo deste tipo de métrica.

Côncavas Métricas côncavas são métricas cuja resultante da associação é o máximo ou o mínimo das métricas referentes às ligações. De uma forma geral, uma métrica côncava, de máximos, resultante da associação, $m_T = m_0 \oplus m_1 \oplus ... \oplus m_n$, onde m_T representa a resultante e m_i representa a métrica referente a cada ligação, é dada pelo representado na equação 4.3.

$$m_T = max(m_0, m_1, ..., m_n) \tag{4.3}$$

[1] Também conhecidas como *delay* e *jitter*.

4.2.2.2 Estratégias de encaminhamento

Um protocolo de encaminhamento com informações de Qos pode ser implementado recorrendo a duas estratégias diferentes. Uma estratégia baseia-se no cálculo de rotas a pedido, por fluxo. Nesta estratégia, mediante um pedido da fonte, o encaminhador mais próximo desencadeia um processo de cálculo de um caminho que consiga satisfazer os requisitos de QoS especificados pela fonte. A outra estratégia possível baseia-se no cálculo antecipado de rotas alternativas. Estas rotas são calculadas e mantidas nos encaminhadores antes de qualquer fluxo os percorrer.

O encaminhamento por fluxo é mais adequado à estratégia de serviços integrados, uma vez que se tem de realizar a reserva de recurso *a priori*. Quando esta reserva se realiza, pode-se também calcular a melhor rota para um conjunto de requisitos de QoS. Na arquitectura dos serviços diferenciados é mais adequado calcular a melhor rota para cada classe. Faz assim sentido calcular antecipadamente o melhor caminho pra cada classe de tráfego. Neste último cenário existe também a vantagem de um encaminhador poder alterar as suas rotas, sem que isso afecte, pelo menos em demasia, o QoS da rota.

No encaminhamento inter-domínio com informações de QoS uma estratégia de cálculo de rotas a pedido pode ter um custo muito elevado, do ponto de vista do número de mensagens trocadas. Geralmente um domínio tem uma quantidade muito grande de tráfego e de fluxos, pelo que os encaminhadores teriam de calcular o melhor caminho muitas vezes.

4.3 Encaminhamento Inter-Domínio com Informações de QoS

> *"Internet routing can be categorized into intra domain routing and inter domain routing, and QoS should be guaranteed both at the intra and inter domains for end-to-end QoS guarantee."* [28]

Num cenário de qualidade de serviço fim a fim, é fundamental equacionar a troca de informações de QoS entre domínios distintos. Existem diversas abordagens para encaminhamento com QoS dentro de um domínio, tais como [29], [30] e [31] no entanto pouco desenvolvimento surgiu para o encaminhamento com QoS num cenário inter-domínio [3].

Existem três grandes vantagens em introduzir informações de QoS no standard de facto para o encaminhamento inter-domínio. Em primeiro lugar, as informações de QoS optimizam a performance do encaminhamento inter-domínio, pois permitem a instalação de rotas com melhor QoS num momento nas tabelas de encaminhamento. Em segundo lugar torna toda a engenharia de tráfego inter-domínio mais eficiente. Por fim, permite fornecer serviços a outros protocolos que requeiram suporte de QoS da camada de rede. [3]

Os principais entraves à introdução de mecanismos de QoS no encaminhamento inter-domínio são a escalabilidade, heterogeneidade e também a estabilidade dos mesmos. A natureza dinâmica da informação de QoS não pode comprometer a escalabilidade do encaminhamento inter-domínio com uma troca de mensagens com elevada frequência. As métricas de QoS devem também ser capazes de suportar a heterogeneidade das ligações inter-domínio. [3]

O objectivo para o qual são pretendidas as informações de QoS também é um desafio para a introdução dos referidos mecanismos. Porque é possível ter

aplicações com requisitos de QoS muito distintos, é necessário ter em conta que pode não existir uma única rota com informações de QoS num determinado momento, mas sim um conjunto de rotas dependendo do objectivo [1].

4.4 Extensões de QoS no BGP para Suporte de Múltiplas Classes de Serviço

> *"There are five enhancements to BGP (...): exchanging potentially multiple paths per prefix, maintaining QoS parameters for each path, pruning the set of known paths to a dominant set (...), choosing a particular path from this dominant set (...), and enforcing the selected path."* [32]

Em [33] [34] [32] são propostas cinco extensões ao protocolo BGP para suportar o encaminhamento diferenciado com múltiplas classes de serviço.

Como o protocolo BGP restringe um encaminhador a anunciar apenas uma rota por destino, não há forma de um par ter conhecimento de uma rota em particular que que forneça parâmetros de QoS para uma dada classe. Para resolver este problema os autores propõem o anúncio de mais de uma rota para cada destino, consoante os parâmetros de QoS de cada rota. A informação contida nestes anúncios também não se restringe apenas ao conjunto de AS's que constituem a rota. Cada AS anuncia então uma lista de rotas dominantes – rotas que garantem melhor QoS para uma determinada classe, e os parâmetros de QoS de cada rota.

4.4.1 Extensões ao protocolo BGP

O trabalho defende cinco extensões ao protocolo BGP por forma a permitir o encaminhamento com múltiplas classes de serviço. Estas extensões são:

1. Anuncio de um conjunto de rotas por cada prefixo.
2. Preservar os parâmetros de QoS de cada Rota.
3. Selecção de rotas dominantes.
4. Escolher uma rota para um requisito de QoS em particular.
5. Garantir que a rota escolhida é percorrida.

De seguida, é apresentado de que forma se implementam as extensões ao protocolo BGP.

4.4.1.1 Anúncio de múltiplas rotas

Como referido anteriormente, os autores defendem que, um anúncio do protocolo BGP contém, no máximo, uma rota para cada destino anunciado, juntamente com os atributos da mesma rota. Se várias rotas existirem para cada destino, para permitir o encaminhamento com QoS, as mensagens de `UPDATE` deverão ser extendidas por forma a permitir que cada rota tenha uma lista de atributos.

As extensões não se ficam por aqui, pois o processo de decisão do BGP deve também ser actualizado. Idealmente, devem também poder ser consideradas válidas todas as rotas de todos os vizinhos. Para tal os atributos `Local_Pref` e `AS_Path` devem poder ser alvo de uma política. Esta política deve assegurar que, no processo de decisão, as rotas aprendidas dos vizinhos, idealmente de todos, tenham o mesmo valor do atributo `Local_Pref` e o mesmo número de AS's que compõem o atributo `AS_Path`.

No estado actual do BGP, se existir pelo menos uma rota aprendida por eBGP, as rotas aprendidas por iBGP devem ser eliminadas. Para suportar

múltiplas rotas por destino é necessário actualizar este comportamento. Todas as rotas recebidas num encaminhador fronteira devem ser alvo de um processo de selecção de onde se extraem as rotas dominantes. Estas rotas dominantes são mantidas pelo encaminhador e não são mais filtradas.

4.4.1.2 Preservar os Parâmetros de QoS

Para além de anunciar várias rotas, com os seu atributos, para cada destino, as mensagens de `UPDATE` devem também ser estendidas para conter os parâmetros de QoS de cada rota. Estas métricas podem incluir a largura de banda, atraso, variação do atraso, e até mesmo a segurança da rota e a disponibilidade da rota. Sempre que se recebe um anúncio de uma rota é necessário fazer a junção dos parâmetros de QoS da rota com os mesmos do encaminhador que recebe o anúncio. As próprias ligações entre pares podem variar entre ligações eBGP – tipicamente uma ligação física entre os pares, e ligações iBGP – tipicamente múltiplas ligações físicas dentro de um domínio.

A natureza das métricas é também uma questão identificada. Estas podem ser aditivas – o atraso, multiplicativas – a taxa de perdas, e côncavas – a largura de banda e segurança de um caminho.

Identifica-se ainda o problema da natureza dinâmica de certas métricas. O compromisso entre precisão das métricas e a escalabilidade e estabilidade do protocolo.

Os autores defendem ainda que, em algumas métricas, devido à sua natureza dinâmica, é crítico decidir o compromisso entre a precisão dos valor de QoS e o custo que esse valor preciso traz ao protocolo. Para tal os autores sugerem que cada parâmetro deve ter associado uma margem de erro.

4.4.1.3 Selecção de rotas dominantes

Devido ao facto de existirem diversos requisitos de QoS, pode não existir uma rota só que garanta todas as combinações de requisitos de QoS. No entanto, pode ser possível que uma rota possa garantir um conjunto grande de combinações de requisitos de QoS. Não é então necessário que um encaminhador fronteira anuncie todas as rotas que conhece, apenas as rotas dominantes. Diz-se que a rota r domina as rotas do conjunto S se fornecer melhor QoS do que qualquer rota de S para todas as métricas de QoS.

Desta forma, um encaminhador apenas anuncia as suas rotas dominantes e não perde desempenho. No entanto é necessário controlar o número de rotas anunciadas para conservar o protocolo escalável.

4.4.1.4 Selecção de uma rota

A selecção de rotas dominantes permite obter um conjunto de rotas para cada destino. Após existir um conjunto de rotas dominantes é necessário saber que rota utilizar para cada destino, tendo em conta os parâmetros de QoS desejados. O processo de decisão do BGP deve no entanto ser estendido para lidar com a nova informação presente nesse momento:

- As rotas dominantes para um destino.
- As métrica de Qos de cada rota dominante.
- Os parâmetros de QoS da classe em questão.

Numa primeira etapa é necessário filtrar as rotas que cumprem os requisitos de QoS em questão. Desse subconjunto de rotas dominantes, qualquer que seja a rota escolhida é válida para a classe em questão. No entanto deve ser

escolhida uma só rota. Essa rota deve ser a que cumpra os requisitos de QoS com mais margem, ou seja, a rota com melhor QoS possível. Utilizando este método garante-se, para além do melhor QoS, a não existência de ciclos. Para cada classe de tráfego é incluída na `RIB` uma entrada com a rota seleccionada.

4.4.1.5 Assegurar utilização da rota escolhida

No trabalho em questão, e relacionado com a utilização efectiva da rota escolhida, são levantadas duas questões:

1. Como podem os encaminhadores IGP tomar conhecimento das rotas dominantes para um destino.

2. Depois de seleccionada uma rota para um destino, como garantir que os pacotes seguem essa mesma rota.

A resposta à primeira questão depende, logicamente, do protocolo de IGP usado, e se este protocolo faz encaminhamento com QoS. Tendo em conta que se está a estender o protocolo inter-domínio para suportar encaminhamento com QoS, faz sentido assumir que o mesmo encaminhamento é suportado pelos protocolos de IGP utilizados. Assim sendo, cada encaminhador fronteira injecta no IGP uma ligação por cada rota dominante, sendo os parâmetros de QoS da ligação os mesmos que na rota dominante relacionada. Isto permite que cada encaminhado IGP possa seleccionar a rota apropriada, para um destino específico, através de um protocolo de estado de ligação. Para manter a escalabilidade podem ser importados apenas algumas rotas, para as quais o encaminhamento com QoS seja mais benéfico.

A resposta à segunda questão depende do modelo de encaminhamento disponível: encaminhamento por rota explícita (EPF – Explicit Path Forwarding)

ou encaminhamento salto-a-salto (HHF – Hop by Hop Forwarding). No caso de se utilizar uma rota explícita, cada pacote tem no seu cabeçalho a rota que deve percorrer. No encaminhamento salto-a-salto não é possível definir nem controlar a rota utilizada. Divide-se então, a resposta à segunda questão, em três cenários possíveis:

1. Encaminhamento por rota explícita fim-a-fim.

2. Encaminhamento por rota explícita ao nível do AS.

3. Encaminhamento salto-a-salto.

No primeiro cenário a rota é explicita no cabeçalho IP. No segundo cenário segue uma indicação dos AS's que o caminho contém, cada AS encaminha o pacote para o AS seguinte na lista. No encaminhamento salto-a-salto é necessário que o IGP suporte as classes utilizadas no encaminhamento inter-domínio.

4.4.2 Conclusões

Nos trabalhos [33], [34] e [32] são propostas cinco extensões ao protocolo BGP para suportar o encaminhamento diferenciado com múltiplas classes de serviço. As extensões encontram-se, de uma forma geral, devidamente descritas, no entanto existem algumas questões que não são referidas nos documentos.

O trabalho em questão não refere como se pode estender as mensagens de `UPDATE` por forma a conter a informação de QoS. Em relação às métricas de QoS, também apenas é referido que é crítico definir o compromisso entre a precisão dos valores das métricas e o custo do aumento da troca de mensagens. Não é apontada nenhuma estratégia para alcançar este objectivo.

O trabalho sugere ainda que aplique uma política aos atributos LOCAL−PREF e AS−PATH. Esta política deve garantir que estes atributos perdem a sua influência no processo de decisão. Ao autor do presente documento, esta estratégia parece em parte desvantajosa. Esta opinião deve-se ao facto de o atributo AS−PATH, apesar de não transportar nenhuma informação de QoS, estar geralmente directamente relacionado com o número de encaminhadores, logo também do número de saltos. Ignorando o AS_PATH perde-se também informações sobre a topologia de rede. Pelo exposto acima, o autor do presente documento defende que a influência deste atributo no processo de decisão do BGP, deve ser revisto, mas não se pode desvalorizar em demasia.

4.5 O Protocolo EQ-BGP

> *"The objective of EQ-BGP is to establish end-to-end paths that offer most suitable QoS guarantees taking into account both the QoS capabilities of particular domains as well as inter-domain links."* [1]

No trabalho apresentado em [1], parte integrante do trabalho [35], é proposto uma extensão ao protocolo BGP. Esta extensão designa-se EQ-BGP(Enhanced QoS Border Gateway Protocol) e tem como objectivo permitir o encaminhamento inter-domínio com QoS, mantendo uma rota para cada classe de tráfego. A extensão ao BGP é composta por quatro componentes:

1. O Atributo QOS_NLRI.
2. Função de agregação de QoS.
3. Algoritmo de decisão de QoS.
4. Múltiplas tabelas de encaminhamento.

O protocolo EQ-BGP, como referido acima, permite o encaminhamento inter-domínio com QoS, com uma tabela para cada classe de tráfego. Para tal, os encaminhadores devem anunciar os destinos aos quais fornecem conectividade juntamente com informação das classes de serviço suportadas para cada rota e do seu nível de QoS. Cada encaminhador deve associar, à informação de QoS presente numa mensagem `UPDATE`, as métricas de QoS da ligação com o par que enviou essa mensagem. O protocolo EQ-BGP utiliza, como métricas, o atraso, a variação do atraso e a taxa de perdas.

4.5.1 O atributo `QOS_NLRI`

No trabalho em questão, defende-se a adaptação de um atributo `QoS_NLRI` [36], apresentado num esboço do IETF (Internet Engineering Task Force).O atributo, descrito em [36], tem como objectivo transportar a informação de QoS de um nó. Na proposta apresentada em [1] este atributo permite transportar informação de QoS referente a várias classes de serviço, como ilustrado na figura 4.1. O atributo é composto pelos seguintes campos:

Número de Classes Neste campo, que ocupa um octeto, indicam-se o número de classes que o atributo `QOS_NLRI` contém.

Identificador da Classe Neste campo de um octeto indica-se um identificador da classe a que se referem os três campos seguintes.

`IPTD` O campo `IPTD` (*IP Packet Transfer Delay*) especifica um atributo de QoS – o atraso.

`IPDV` O campo `IPDV` (*IP Packet Delay Variation*) especifica um atributo de QoS – a variação do atraso.

H

Tabela 4.2: *Classes de Serviço previstas no trabalho apresentado em [1] e seus requisitos de QoS*

	IPTD [ms]	IPDV [ms]	IPLR
Telefonia	100	50	10^{-3}
Tempo-Real Interactivo	100	50	10^{-3}
Multimédia em Fluxo	1000	–	10^{-3}
Dados de Alto Débito	1000	–	10^{-3}
Padrão	–	–	–

IPLR O campo IPLR (*IP Packet Loss Rate*) especifica um atributo de QoS – a taxa de perdas de pacotes.

O atributo contém os valores dos parâmetros de QoS das vária classes. A cada classe corresponde um identificador de classe, um valor de IPTD, um valor de um valor de IPDV e um valor de IPLR. As classes são ordenadas por ordem crescente do respectivo indicador de classe.

4.5.2 Função de agregação de QoS

O trabalho apresentado em [1] sugere a implementação de cinco classes de serviço, apresentadas, com seus requisitos de QoS, e em função das métricas propostas no trabalho, na tabela 4.2.

Para as métricas apresentadas, é sugerida, como função de agregação das mesmas, a soma. No caso do IPTD a natureza aditiva da métrica, por si só, sugere a soma como função de agregação. O autor de [1] sugere que se utilize também a soma como função de agregação para o IPDV e o IPLR. Esta sugestão baseia-se no valor reduzido destas métricas, tipicamente inferiores a 10^{-2} ms.

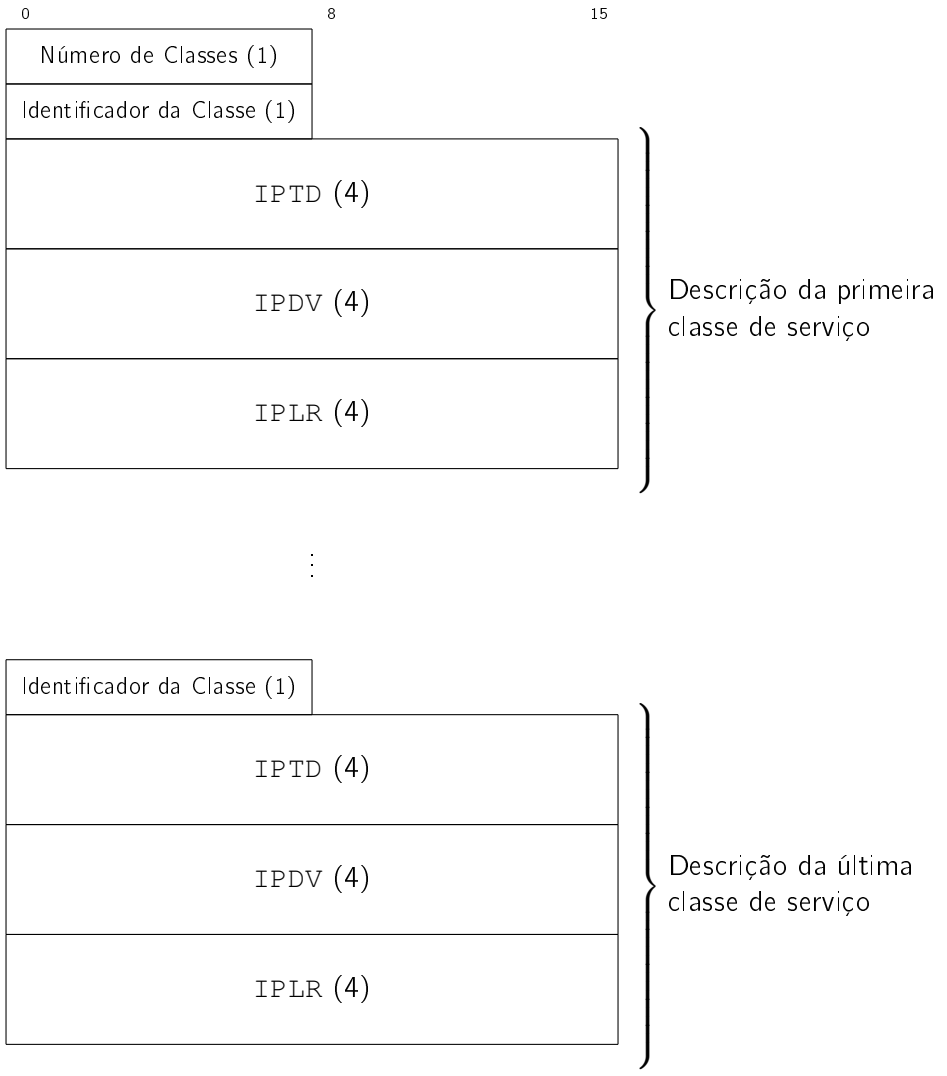

Figura 4.1: *Codificação do atributo* `QOS_NLRI`*, segundo a proposta [1]*

4.5.3 Algoritmo de Decisão de QoS

O trabalho apresenta um algoritmo para decidir o grau de preferência de uma rota. Este algoritmo, sugere-se, deve substituir o critério do tamanho atributo `AS_PATH` no cálculo do grau de preferência no processo de decisão do BGP. O valor do grau de preferência, DoP (*Degree of Preference*), é dado pela equação 4.4, onde i representa um determinado parâmetro de QoS de uma rota, T_i representa um requisito do parâmetro i (ver tabela 4.2), q_i representa o valor de QoS do parâmetro i, e f_i representa um valor atribuído ao parâmetro que permite controlar o comportamento do algoritmo.

$$DoP = \sum_{i \in \texttt{IPTD,IPDV,IPLR}} \frac{f_i}{max(0, T_i - q_i)} \tag{4.4}$$

A rota que apresentar menor valor como grau de preferência é a seleccionada. Se em algum caso, o valor q_i do parâmetro i for superior ao requisito T_i da classe, a rota é tratada como não estando de acordo com os requisitos de QoS.

4.5.4 Múltiplas Tabelas de Encaminhamento

O trabalho em questão defende a utilização de várias classes de serviço com requisitos de QoS distintos (ver tabela 4.2). Este comportamento necessita que existam várias tabelas de encaminhamento nos encaminhadores, mais propriamente uma por cada classe de serviço suportada.

4.5.5 Conclusões

O trabalho descrito em [1] descreve de forma muito sumária um protocolo que consiste no protocolo BGP acrescido de um conjunto de extensões. Da mesma forma que o trabalho descrito em [33], [34] e [32], e apresentado na secção 4.4, também esta proposta se baseia na utilização de classes de serviço. A proposta sugere uma forma de incluir as informações de QoS na mensagem `UPDATE`, recorrendo para isso à utilização de um novo atributo – *QoS_NLRI* [36].

A proposta EQ-BGP não inclui, como métrica, a largura de banda, ou qualquer outra métrica relacionada com o débito da rede. Esta ausência afigura-se, ao autor do presente documento, como crítica, pois existem aplicações que requerem explicitamente elevada largura de banda, onde o atraso pode ser pouco significativo. As funções de agregação de QoS e o algoritmo de decisão, que são adaptados para as métricas sugeridas, não conseguem lidar com a inclusão de uma métrica côncava como o a largura de banda disponível. O facto de o algoritmo de decisão se sobrepor ao número de AS's que compõem o `AS_PATH`, da mesma forma que para [33], [34] e [32], faz com que se perca a informação relativa à topologia de AS's.

4.6 Representação Estatística das Métricas de QoS

> *"(...)ABI and DI can represent the fine-grained statistical property of the available bandwidth and delay efficiently. With ABI and DI, the major statistical property of the instantaneous values can be captured with acceptable processing overhead."* [37]

Em [3] e [37] introduzem-se novas métricas para o encaminhamento inter-domínio com QoS e define-se como essas métricas permitem estender o pro-

tocolo BGP. As métricas apresentadas não representam valores instantâneos de um atributo numa ligação, – largura de banda disponível, por exemplo – mas sim uma representação estatística desse atributo. Os atributos são representados como um intervalo de valores – $\bar{\omega}$ – e uma probabilidade – ρ, significando isto que o valor instantâneo pertence ao intervalo $\bar{\omega}$ com uma probabilidade ρ. Esta representação estatística das métricas é mais estável que a representação instantânea, reduzindo assim o número de anúncios necessários, não prejudicando a escalabilidade do BGP.

A representação estatística permite também suportar melhor a heterogeneidade ao nível das ligações entre AS's. Diferentes intervalos $\bar{\omega}$ podem diferenciar numa ligação directa entre AS's de numa ligação que contenha encaminhadores que não fazem encaminhamento com QoS.

4.6.1 Modelo de Rede

O modelo de rede apresentado é descrito pelo grafo $G(V, E)$, onde V representa o conjunto de encaminhadores que suportam o encaminhamento com QoS e E o conjunto das ligações entre esses encaminhadores. Existem três tipos de ligações lógicas:

`TYPE-1` Uma ligação lógica do tipo TYPE-1 representa uma ligação directa entre encaminhadores que suportem QoS, tipicamente estas ligações ocorrem entre AS's vizinhos.

`TYPE-2` Uma ligação lógica TYPE-2 representa uma ligação IGP dentro de um AS, ligando dois encaminhadores pertencentes ao mesmo AS.

TYPE-3 As ligações lógicas TYPE-3 são compostas por ligações físicas através de múltiplos AS's, em que todos os encaminhadores intermédios não suportam QoS.

O processo de obtenção dos atributos de cada $e \in E$ varia conforme o tipo de ligação. No caso de e se tratar de uma ligação lógica TYPE-1 pode-se obter os parâmetros de QoS através da monitorização directa. Se e for do tipo TYPE-2, os protocolos de IGP podem fornecer essa informação. Quando e é uma ligação lógica TYPE-3 é necessário calcular os parâmetros de QoS fim-a-fim.

4.6.2 Métricas propostas: ABI e DI

Como referido acima, a proposta apresenta métricas estatísticas para representar os parâmetros de QoS. Estas métricas são representadas como um intervalo $\bar{\omega}$ de valores e uma probabilidade ρ de o valor instantâneo pertencer a esse intervalo. São definidas então duas métricas: o ABI (Available Bandwidth Index) e o DI (Delay Index).

O ABI define-se como $\hat{b} = b_l, b_u, \rho$, significando que a probabilidade de a largura de banda instantânea b pertencer ao intervalo $\bar{\omega} = [b_l, b_u]$ não é menor que ρ, i.e., $P_r[b \in \bar{\omega} = [b_l, b_u] \geq \rho]$.Analogamente, DI define-se como $\hat{d} = d_l, d_u, \rho$, significando que a probabilidade de o atraso instantâneo d pertencer ao intervalo $\bar{\omega} = [d_l, d_u]$ não é menor que ρ, i.e., $P_r[d \in \bar{\omega} = [d_l, d_u] \geq \rho]$.

Nestas métricas, $\bar{\omega}$ representa a variação dos valores instantâneos e ρ a sua precisão. Existem várias vantagens no uso de ABI e DI como métricas de encaminhamento. O ABI e DI permitem representar as variações instantâneas dos parâmetros sem um custo muito grande, evitando recorrer a funções de densidade de probabilidade. Permitem também obter muito mais informação

que métricas estáticas como a capacidade da ligação ou o número de saltos. Outra vantagem do uso do ABI e do DI é o facto de manter a extensão ao BGP escalável. Mesmo que o valor instantâneo de um parâmetro varie frequentemente, a sua distribuição estatística varia com uma frequência inferior, podendo até não variar. Assim sendo, o número de anúncios necessários no BGP será muito menor do que o necessário caso se utilize métricas instantâneas.

A heterogeneidade das ligações lógicas também é melhor suportada com estas métricas estatísticas. Uma ligação lógica TYPE-3 pode ser representada por um intervalo $\bar{\omega}$ muito largo ou por uma probabilidade ρ reduzida.

4.6.3 Calculo do ABI e do DI de um Percurso

Uma rota compõe-se agregando várias ligações lógicas. Assim para calcular as métricas de QoS de uma rota pode-se calcular os parâmetros de QoS de fim-a-fim ou obter a métrica da rota através da junção das métricas das ligações lógicas que compõem a rota. No entanto calcular o QoS de fim-a-fim para cada rota não é escalável no actual cenário do BGP. A operação de junção é diferente no caso de se tratar de ABI ou DI porque a largura de banda é uma métrica côncava e o atraso é uma métrica aditiva.

4.6.3.1 Operação de junção do ABI

Dados dois ABI's, $\hat{b}_1$ e $\hat{b}_2$ define-se a operação de junção do ABI como $\hat{b} = \hat{b}_1 \oplus \hat{b}_2$. Assim sendo, o ABI de uma rota $v_1 v_2 ... v_n$ é $\hat{b}_{v_1 v_2} \oplus ... \oplus \hat{b}_{v_{n-1} v_n}$. Assume-se ainda que os ABI's de diferentes ligações são independentes e que a distribuição de largura de banda fora de $\bar{\omega} = [b_l, b_u]$ é simétrica em torno de

$\bar{\omega}$. São propostos então, dois métodos de cálculo para a operação de junção de ABI.

Define-se $\hat{b_n}.\bar{\omega}$ como notação para o intervalo com probabilidade ρ que tem como valor médio $\hat{b}$. Define-se também, por comodidade, $l_n = \hat{b}_n.b_l$ como o limite inferior do intervalo $\hat{b_n}.\bar{\omega}$ e $u_n = \hat{b}_n.b_u$ como o limite superior do intervalo $\hat{b_n}.\bar{\omega}$.

Método 1: Sabendo que $\hat{b}_1, \hat{b}_2$ são dois ABI's para as ligações 1 e 2 e $\hat{b} = \hat{b}_1 \oplus \hat{b}_2$, então $\hat{b}.\bar{\omega} = [min(\hat{b}_1.b_l, \hat{b}_2.b_l), min(\hat{b}_1.b_u, \hat{b}_1.b_u)]$.

O intervalo resultante do método 1 para cálculo da operação de junção tem como valores inferior e superior os mínimos dos valores inferiores e superiores dos intervalos da operação. O valor da probabilidade $\hat{b}.\rho$ depende da relação entre $\hat{b}_1.u$ e $\hat{b}_2.u$ da seguinte forma:

- $\hat{b}_1.u \leq \hat{b}_2.u$:
  $\hat{b}.\rho = \dfrac{\hat{b}_1.\rho(1 + \hat{b}_2.\rho)}{2}$
- $\hat{b}_1.u \geq \hat{b}_2.u$:
  $\hat{b}.\rho = \dfrac{\hat{b}_2.\rho(1 + \hat{b}_1.\rho)}{2}$

Método 2: Partindo dos mesmos princípios do método 1 a operação de junção seguido o método 2 resulta em $\hat{b}.\bar{\omega} = [min(\hat{b}_1.b_l), \hat{b}_2.b_l).max(\hat{b}_1.b_u), \hat{b}_2.b_u)]$

O método 2 faz com que $\hat{b}.\bar{\omega}$ cubra todo o intervalo ocupado por $\hat{b}_1.\bar{\omega}$ e $\hat{b}_2.\bar{\omega}$, nunca $\hat{b}.\bar{\omega}$ sendo menor que $\hat{b}_1.\bar{\omega}$ e $\hat{b}_2.\bar{\omega}$. O valor da probabilidade $\hat{b}.\rho$ resultante da junção é dado por $\dfrac{\rho 1 + \rho 2}{2}$.

A vantagem do método 2 sobre o anterior deve-se ao facto de, a probabilidade ρ resultante da junção nunca ser menor que as probabilidades ρ_1 e ρ_2 simultaneamente. Esta vantagem pode levar também a um aumento significativo do intervalo $\bar{\omega}$.

Os autores sugerem ainda em que situações se deve utilizar cada um dos métodos.

- Deve-se utilizar o método 1 quando os conjuntos são disjuntos ou pouco coincidentes
- Deve-se utilizar o método 2 quando os conjuntos tem uma grande área coincidente.

4.6.3.2 Operação de junção do DI

A operação de junção do DI, sendo aplicada a uma métrica aditiva, baseia-se na soma dos DI's de cada ligação lógica. Dados dois DI's, $\hat{d}_1$ e $\hat{d}_2$ define-se a operação de junção do DI como $\hat{d} = \hat{d}_1 \oplus \hat{d}_2$. Assim sendo, o DI de uma rota $v_1 v_2 ... v_n$ é dado por $\hat{d}_{v_1 v_2} \oplus ... \oplus \hat{d}_{v_{n-1} v_n}$. Partindo da mesma ideia da operação de junção do ABI, define-se o intervalo $\hat{d}.\bar{\omega}$ com limites inferiores e superiores $\hat{d}.d_l$ e $\hat{d}.d_u$ e probabilidade $\hat{d}.\rho$.

Método 1: Sabendo que $\hat{d}_1, \hat{d}_2$ são dois DI's para as ligações 1 e 2 e $\hat{d} = \hat{d}_1 \oplus \hat{d}_2$, então $\hat{d}.\bar{\omega} = [\hat{d}_1.d_l + \hat{d}_2.d_l, \hat{d}_1.d_u + \hat{d}_1.d_u]$. A probabilidade $\hat{d}.\rho$ é dada por $\hat{d}.\rho = \hat{d}_1.\rho.\hat{d}_2.\rho$

Método 2: Partindo das mesmas condições do método 1, calcula-se a junção como $\hat{d}.\bar{\omega} = [max(\hat{d}_1.d_l, \hat{d}_2.d_l), \hat{d}_1.d_u + \hat{d}_1.d_u]$. A probabilidade $\hat{d}.\rho$ depende da relação entre $\hat{d}_1.d_l$ e $\hat{d}_2.d_l$ da seguinte forma:

- $\hat{d}_1.d_l < \hat{d}_2.d_l$:
  $\hat{d}.\rho = \frac{\hat{d}_2.\rho(1 + \hat{d}_1.\rho)}{2}$
- $\hat{d}_1.d_l > \hat{d}_2.d_l$:
  $\hat{d}.\rho = \frac{\hat{d}_1.\rho(1 + \hat{d}_2.\rho)}{2}$

Quanto à utilização de cada um dos métodos, os autores defendem que geralmente deve ser utilizado o método 1, no entanto, o método 2 deve ser utilizado quando a diferença entre $\hat{d}_1.d_l$ e $\hat{d}_2.d_l$ é elevada.

4.6.4 Extensões ao protocolo BGP

No trabalho apresentado em [3] e [37], para além das métrica DI e ABI, são também apresentadas formas de estender o protocolo BGP, de forma a suportar as métricas referidas. São apontadas as seguintes extensões:

1. Estender as mensagens de `UPDATE` por forma a anunciarem informação de QoS.

2. Seleccionar rotas conforme a informação de QoS.

3. Monitorizar e actualizar as informações de QoS das rotas.

4.6.4.1 Extensão das mensagens `UPDATE`

Propõe-se, para as mensagens `UPDATE` permitirem a troca de informação de QoS, dois novos atributos de caminho. Um primeiro atributo, opcional e transitivo, que contém a informação de QoS. Um segundo atributo, opcional e transitivo, que contém o endereço do último par BGP com encaminhamento por QoS. O segundo atributo permite a um encaminhador descobrir o tipo de ligação, em termos de encaminhamento com QoS, que tem com o seu par - ligação tipo `TYPE1` ou `TYPE3`. Recebendo uma mensagem `UPDATE`, um par verifica se o par que origina a mensagem coloca o seu endereço no segundo atributo. Existem então três possibilidades:

1. Caso o par coloque o endereço, e o ASN do par que origina seja diferente do ASN que do encaminhador que recebe a mensagem, a ligação entre os pares é do tipo `TYPE 1`.

2. Caso o par coloque o endereço, e o ASN do par que origina seja igual ao ASN que do encaminhador que recebe a mensagem, a ligação entre os pares é do tipo `TYPE 2`.

3. Caso o par não coloque o seu endereço no atributo, a ligação entre o último par com encaminhamento com informações de QoS e o encaminhador que recebe a mensagem é do tipo `TYPE 3`.

4.6.4.2 Selecção de rotas com melhor QoS

Propõe-se, no trabalho em questão, que a selecção da melhor rota seja segundo o melhor ABI ou o melhor DI. Apenas se considera uma métrica para decidir a melhor rota. Devido à composição das métricas em questão – um intervalo $\bar{\omega}$ com uma probabilidade ρ, a comparação destas não é imediata. A estratégia é então normalizar as métricas e calcular um peso para cada rota consoante o QoS pretendido.

Para normalizar as métricas, o parâmetro $|\bar{\omega}|$ é estendido, conforme o valor de ρ, assumindo uma distribuição normal para $|\bar{\omega}|$. Assume-se então que o valor da largura de banda instantânea b segue uma distribuição normal $\mathcal{N}((b_l + b_u)/2, \sigma^2)$, em que b_l e b_u representam os limites inferior e superior do parâmetro $|\bar{\omega}|$.

O intervalo normalizado é dado por $|\bar{\omega'}| = [b_m - \delta, b_m + \delta]$, onde $b_m = (b_l + b_u)/2$ e $\delta = (b_u - b_l)/2\rho$. A normalização do DI realiza-se da mesma forma que no ABI.

É ainda necessário calcular o peso de cada rota tendo em conta o ABI ou o DI – $W_b e W_d$ respectivamente. O peso do ABI define-se como $W_b = b_m - \eta\delta$, onde $\eta > 0$ representa o compromisso entre a preferência pela estabilidade dos valores e o seu valor médio. Da mesma forma, o peso de uma rota tendo em conta o DI define-se como $W_d = (d_u + d_l + \eta(d_u - d_l)/\rho)/2$. São seleccionadas as rotas que apresentem maior W_b ou menor W_d.

4.6.4.3 Actualização da informação de QoS

Como referido anteriormente, as métricas de QoS em questão são representadas tendo em conta a sua distribuição estatística e não o valor instantâneo, por forma a reduzir a variação dos anúncios. Da mesma forma, a actualização de informação deve ser anunciada de forma criteriosa.

Existem dois tipos de alterações nos parâmetros de QoS. Um primeiro cenário surge quando se altera a informação de QoS de uma ligação lógica. O segundo cenário trata o caso de se receber uma mensagem `Update` com informação actualizada. são então definidos dois valores limite para cada cenário a partir dos quais se procede à actualização das informações de QoS. Os dois limites são:

1. *Link-State Threshold (T_l)*: Quando ocorre uma alteração numa ligação lógica e produz uma diferença no peso da rota superior a este valor limite, considera-se que a informação foi alterada.

2. *Route Update Threshold (T_r)*: Uma rota substitui outra, na tabela de encaminhamento, quando o seu peso traz uma diferença superior a este limite.

4.6.5 Extensão do ABI e DI para Histogramas

As métrica propostas, ABI e DI, representam informação dinâmica de QoS. Os autores do trabalho sugerem estender essa informação, por forma a anunciar não um intervalo e uma probabilidade, mas sim um conjunto de intervalos e a probabilidade de cada intervalo. Apresenta-se então no trabalho as métricas ABH (*Available Bandwidth Histogram*) e DH (*Delay Histogram*).

4.6.5.1 Definição de ABH e DH

No trabalho em questão define-se ABH e DH como um conjunto $(\bar{\omega}_i, \rho_i)$, onde $\bar{\omega}_i = [l_i, u_i]$ e ρ_i representam o intervalo e a probabilidade de índice i. A largura de cada intervalo reflecte o compromisso entre precisão e custo computacional. As métricas ABH e DH resultam, portanto, de dividir um ABI ou DI em sub-intervalos. Devem ser respeitadas as seguintes condições:

- Quaisquer diferentes w_i não se devem sobrepor.
- Quaisquer intervalos consecutivos devem partilhar um limite.
- Todos os intervalos devem ter a mesma largura – $|\bar{\omega}|$.

A figura 4.2 ilustra graficamente um histograma. Como são compostos da mesma forma, um ABH, para os mesmos intervalos com a mesma probabilidade, equivale a um DH.

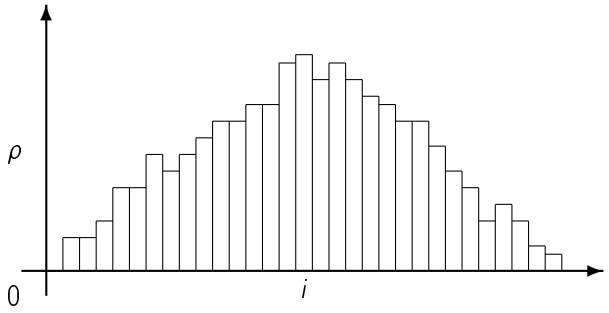

Figura 4.2: *Exemplo gráfico de um histograma com 60 intervalos, representado a probabilidade ρ e o índice do intervalo i.*

4.6.5.2 Operações de junção do ABH e DH

Seguindo a mesma notação das operações de junção do ABI e DI, define-se a operação de junção do ABH como $abh = abh1 \oplus abh2$ e do DH como $dh = dh1 \oplus dh2$. Para o caso da junção da largura de banda, $b = min(b1, b2)$, a função de distribuição acumulada é dada pela equação 4.5.

$$F_b(x) = F_{b1}(x) + F_{b2}(x) - F_{b1}(x)F_{b2}(x) \tag{4.5}$$

Na equação 4.5 $F_b(x)$ representa a função de distribuição acumulada da variável b. Sabendo as funções de distribuição de b_1 e b_2 é possível calcular a função de distribuição da junção. Os autores defendem que a probabilidade de a resultante da junção do ABH pertencer ao intervalo $[x, x + \delta x]$ –

$P[b \in [x, x+\delta x]]$ é dada pela equação 4.6.

$$\begin{aligned} P[b \in [x, x+\delta x]] = & P[b_1 \in [x, x+\delta x]](1 - F_{b2}(x)) \\ & + P[b_2 \in [x, x+\delta x]](1 - F_{b1}(x)) \\ & - P[b_1 \in [x, x+\delta x]]P[b_2 \in [x, x+\delta x]] \end{aligned} \tag{4.6}$$

Partindo da equação 4.6, os autores sugerem uma aproximação para a junção de ABH, $abh = abh1 \oplus abh2$, descrita no o algoritmo que se encontra especificado na listagem 4.1.

```
1 s ← min{i : abh1.ρi > 0 ou abh2.ρi > 0}
  t ← max{i : abh1.ρi > 0 ou abh2.ρi > 0}
3 ω1 ← 0.0
  ω2 ← 0.0
5 para i ← s até t
  fazer
7     abh.ρi ← (1 − ω2)abh1.ρi + (1 − ω1)abh2.ρi − abh1.ρi.abh2ρi
      ω1 ← ω1 + abh1.ρi
9     ω2 ← ω2 + abh2.ρi
```

Listing 4.1: *Algoritmo para calcular* $abh = abh1 \oplus abh2$

A função de distribuição da resultante da junção do atraso pode também ser definida segundo a função de distribuição dos atrasos em junção como definido na equação 4.7, onde f_d representa a função de distribuição do atraso d.

$$f_d(x) = \int_0^x f_{d1}(t) f_{d2}(x-t)\,dt \tag{4.7}$$

Partindo da equação 4.7 e tomando $dh_1 \oplus dh_2$, onde dh_i é um atraso operando da junção, os autores defendem que é possível aproximar a junção de DH, pela

equação 4.8.

$$dh.\rho_k = \sum_{i=0}^{k} dh_1.\rho_i dh2.\rho_{k-i} \tag{4.8}$$

4.6.5.3 Selecção de rotas com melhor QoS

Uma vez que as métricas por histogramas fornecem uma informação mais precisa que o ABI e DI, é possível obter indicadores matemáticos mais fiáveis com a esperança matemática ou a variância. Desta forma, os autores defendem que o peso de um rota, adaptado do método descrito na secção 4.6.4.2, tendo em conta a largura de banda, é dado pela equação 4.9, ao passo que o peso de uma rota, tendo em conta o atraso, deve ser dado pela equação 4.10. Nas equações 4.9 e 4.10, **E** representa a esperança matemática e **STD** representa o desvio padrão.

$$W_b = \mathbf{E}[abh] - \eta\mathbf{STD}(abh) \tag{4.9}$$

$$W_d = \mathbf{E}[dh] - \eta\mathbf{STD}(dh) \tag{4.10}$$

4.6.5.4 Anúncio da Média e Variância das Métricas

Os autores defendem a possibilidade de anunciar, ao invés dos histogramas completos de uma métrica, as suas média e variância. Desta forma, no caso do atraso, anunciar-se-ia o atraso médio e variância do atraso. Como o atraso é uma métrica aditiva, e assumindo uma distribuição normal, a média resultante da junção é dada pela soma das médias, da mesma forma que a variância resultante da junção é dada pela soma das variâncias.

Anunciando o valor médio e variância da largura de banda, devido a esta ser uma métrica côncava, a junção destes parâmetros apresenta uma particularidade. Tomando o valor médio da junção como o menor das médias em junção, pode resultar num valor superior ao esperado. A desigualdade de Jensen, $min(E[b_1], E[b_2] \geq E[min(b_1, b_2)])$, demonstra isso mesmo.

4.6.6 Conclusões

O trabalho proposto em [3] e [37] apresenta uma abordagem bastante completa para a implementação de QoS no protocolo BGP. São utilizados dois conceitos de métricas estatísticas: ABI, DI, ABH e DH. A natureza estatística destas métricas coaduna-se com a estabilidade e escalabilidade que se pretendem para o protocolo BGP. No trabalho são apenas consideradas as representações estatísticas da largura de banda instantânea e do atraso instantâneo, sendo que o melhor caminho é calculado tendo em conta apenas uma métrica. O número de AS's que compõem o atributo `AS_PATH` também não é considerado no trabalho em análise.

Quanto ao cálculo do peso de uma rota, utilizando o ABI, a fórmula para o W_b(ver secção 4.6.4.2), não parece, ao autor do presente documento, estar bem especificada. No trabalho refere-se que o intervalo normalizado, para as métricas, é dado por $|\bar{\omega'}| = [b_m - \delta, b_m + \delta]$, onde $b_m = (b_l + b_u)/2$ e $\delta = (b_u - b_l)/2\rho$. Refere-se também que peso do peso do ABI define-se como $W_b = b_m - \eta\delta$. Olhando mais atentamente à definição de W_b e tomando

$\eta = 1$:

$$W_b = \frac{b_l + b_u}{2} - \frac{b_u - b_l}{2\rho} \Leftrightarrow \quad (4.11)$$

$$2W_b = b_u + b_l - \frac{b_u - b_l}{\rho} \Leftrightarrow \quad (4.12)$$

$$2W_b = \frac{\rho b_u - b_u + \rho b_l + b_l}{\rho} \Leftrightarrow \quad (4.13)$$

$$W_b = \frac{b_u(\rho - 1) + b_l(\rho + 1)}{2\rho} \Leftrightarrow \quad (4.14)$$

$$lim(W_b)_{\rho \to 1} = \frac{b_u(1 - 1) + b_l(1 + 1)}{2} = b_l \quad (4.15)$$

A equação 4.15 revela que, quando o valor de ρ tende a aproximar-se de 1, o peso do limite superior do intervalo normalizado do ABI tende a desaparecer e o peso do limite superior tende a aproximar-se de 2. Este comportamento faz com que, existindo duas rotas r_1, r_2 com os seguintes intervalos: $b_{r1} =< 90, 300 >$ e $b_{r2} =< 95, 110 >$, para um $\rho \approx 1$ é seleccionada a rota r_2, o que não é desejável.

4.7 Conclusão

No presente capítulo foi efectuada uma introdução ao tema de qualidade de serviço e ao encaminhamento com qualidade serviço, nomeadamente entre diferentes domínios. Após a introdução apresentam-se três propostas para o extensões de QoS para o protocolo BGP.

As duas primeiras proposta defendem o encaminhamento inter-domínio com classes de serviço. Este facto, aliado à investigação realizada para tratar o tema, permite ao autor concluir que a abordagem das classes de serviço pode ser a mais indicada para o encaminhamento inter-domínio. O autor conclui também que, o uso de métricas estatísticas permite não degradar

em demasia a estabilidade e escalabilidade do protocolo BGP. Desta forma, entende-se que seria pertinente, equacionar uma solução para o encaminhamento inter-domínio com informações de qualidade de serviço, que permitisse o encaminhamento por classes de serviço recorrendo métricas estatísticas.

Conclui-se também que, a junção de métricas ao longo de uma rota e, o cálculo do grau de preferência de uma rota, tendo em conta os parâmetros de QoS, são áreas críticas de um protocolo de encaminhamento deste tipo. As conclusões retiradas ao longo deste capítulo dão o mote ao capitulo que segue onde se propões uma solução para o problema do encaminhamento inter-domínio.

5 Solução Proposta

5.1 Introdução

No presente capítulo apresenta-se uma proposta para o encaminhamento inter-domínio com informações de QoS. Esta proposta baseia-se no encaminhamento com classes de serviço e inclui a utilização de várias métricas, entre elas métricas estatísticas.

O presente capítulo encontra-se dividido em seis partes. Numa primeira parte são identificados os objectivos da proposta e faz uma caracterização geral da mesma.

Na segunda parte do presente capítulo definem-se as métricas sugeridas na proposta e que parâmetros de QoS representam. Na terceira parte apresenta-se a forma como as mensagens do protocolo BGP são estendidas, por forma a suportar as métricas sugeridas. Na quarta parte, do presente capítulo, descreve-se como se defende que deve ser a operação de junção de cada uma das métricas especificadas.

Na quarta parte do presente capítulo expõe-se como as tabelas de encaminhamento são também estendidas, para suportar a solução proposta. Por

fim apresentam-se as extensões, necessárias ao processo de selecção de rotas do BGP, para a solução.

5.2 Objectivos

O presente documento tem como objectivo, entre outros, apresentar uma proposta, para uma extensão ao protocolo BGP, que permita o encaminhamento de tráfego com informações de QoS. Ao conceber a proposta, houve uma tendência para que esta obedecesse aos seguintes princípios:

1. A solução deve permitir um conjunto significativo de métricas, que possam cumprir os mais variados requisitos por parte das aplicações.

2. A solução proposta não deve degradar a estabilidade e escalabilidade do protocolo BGP.

3. A solução proposta deve ter como objectivo a interoperabilidade com o actual estado do protocolo BGP.

4. Para além da eventual criação de atributos opcionais, a proposta deve ter com objectivo minimizar as alterações efectuadas ao protocolo BGP.

5. A solução proposta deve permitir o encaminhamento por classes de serviço.

Os requisitos apresentados, na opinião do autor, têm como objectivo, facilitar uma eventual implementação da proposta numa implementação do protocolo BGP. A proposta, sugerida no presente documento, tem como base as propostas apresentadas em em [3], [37], [34], [33], [32] e [1]. De uma forma muito

geral, pretende-se conceber uma extensão ao BGP, que permita o encaminhamento de tráfego por classes de serviço, introduzindo os conceitos de métricas estatísticas.

Na solução proposta no presente documento defende-se que o cálculo do melhor seja efectuado para cada classe de tráfego, conforme defendido, por sua vez, em [1] e [34]. Esta estratégia permite suportar os requisitos de QoS de cada classe, desta forma classes com requisitos opostos, podem ver os seus requisitos cumpridos, caso hajam rotas para tal, ao contrário do cenário onde o melhor caminho é calculado, tendo em conta informações de QoS, apenas uma vez para todos os fluxos.

Sugere-se neste documento, por forma a facilitar a interoperabilidade com o actual padrão do protocolo BGP, a utilização do modelo rede proposto em [37]. Este modelo, como descrito na secção 4.6.1, pressupõe a existência de três tipos de ligações, sendo eles:

TYPE-1 Uma ligação lógica do tipo `TYPE-1` representa uma ligação directa entre encaminhadores que implementam a solução proposta.

TYPE-2 Uma ligação lógica `TYPE-2` representa uma ligação IGP dentro de um AS, ligando dois encaminhadores que implementam a solução proposta.

TYPE-3 As ligações lógicas `TYPE-3` são compostas por ligações físicas através de múltiplos AS's, em que todos os encaminhadores intermédios não suportam a solução proposta.

O modelo de rede apresentado acima, na opinião do autor, retrata com uma precisão bastante aceitável, as várias ligações possíveis num cenário de conectividade inter-domínios. As ligações do tipo `TYPE-3` prevêem a existência de

encaminhadores, e AS's, que apenas implementem o padrão actual do protocolo BGP, não introduzindo a solução proposta para o encaminhamento com QoS. A consciência da existência destes encaminhadores é vital para permitir a interoperabilidade entre as duas versões do protocolo BGP.

5.3 Métricas de QoS

Por forma ao encaminhamento inter-domínio permitir a troca de informações de QoS, deve existir unanimidade, por parte dos sistemas autónomos, nas métricas utilizadas, quer para que se possa associar uma métrica a uma rota de fim-a-fim, quer para a definição das classes de tráfego. Para a proposta apresentada no presente documento, defende-se a utilização das seguintes métricas:

- IPLR, conforme descrito em [1], a taxa de pacotes perdidos.
- ABH, conforme descrito em [37], a representação estatística da largura de banda disponível.
- DH, conforme descrito em [37], a representação estatística do atraso.
- Segurança de um caminho, conforme sugerido em [34].

As métricas sugeridas, na opinião do autor, permitem caracterizar um estado da rede com bastante precisão. Em [1] sugere-se para além da métrica IPLR, a utilização de mais duas métricas, o atraso médio e a variação do atraso. Estas duas métricas, como se pode verificar em [37], podem-se extrair da métrica DH. O trabalho apresentado em [34] sugere precisamente, que se utilizem as métricas referentes a perdas de pacotes, atraso e largura de banda. O trabalho sugere também uma métrica adicional que pode ser a segurança da

rota. Apesar de a utilização desta métrica não ser mais defendida ao longo do trabalho referido, foi alvo de investigação por parte do autor e sugere-se a sua utilização na proposta apresentada.

Como referido anteriormente, a métrica IPLR diz respeito à taxa de pacotes perdidos. Esta métrica é importante em classes de serviço que tenham restritos requisitos em termos de perdas. As métricas ABH e DH relacionam-se com o débito e latência da rede. Estas métricas são importantes para classes que pretendam servir aplicações para interacção humana.

Cada métrica referida dá lugar a um atributo BGP. Estes atributos serão descritos em secções seguintes.

5.3.1 As Métricas ABH e DH

As métricas ABH e DH, sugeridas no presente relatório, encontram-se descritas em [3] e [37] e na secção 5.3.1 do presente documento. Como já referido estas métricas referem-se à representação do histograma da largura de banda disponível. Cada ABH é composto por um intervalo de largura de banda, dividido por sua vez em sub-intervalos de igual largura. Cada sub-intervalo tem associado a si uma probabilidade. A figura 4.2 ilustra um exemplo de uma representação por histograma, onde se representam o índice de cada sub-intervalo, i, no eixo horizontal e a probabilidade de cada sub-intervalo, ρ, no eixo vertical. Da mesma forma que o ABH, o DH representa um histograma de um intervalo do tempo de atraso.

No trabalho apresentado em [37] não se define a forma como as métricas são codificadas, apenas se indica que intervalos consecutivos, de igual probabilidade devem estar colapsados, por forma a reduzir o tamanho das mensagens `UPDATE`. No presente trabalho apresenta-se uma forma de representar estas

métricas. O ABH e DH dependem, para a sua representação, da largura dos sub-intervalos, $|\bar{\omega}|$, do valor inicial do primeiro sub-intervalo, $\bar{\omega}_0.l$, e do número total de intervalos, N. Sugere-se então que se represente primeiro o valor inicial do primeiro sub-intervalo, $\bar{\omega}_0$, seguido da largura dos intervalos, $|\bar{\omega}|$, seguido do índice de cada intervalo e a sua probabilidade, $< \bar{\omega}_i, \rho_i >$. Havendo intervalos consecutivos de iguais valores de ρ, deve apenas representar-se o primeiro dos intervalos. Concretizando, existindo três intervalos $\bar{\omega}_1, \bar{\omega}_2, \bar{\omega}_3$ com probabilidades $\rho_1, \rho_2 = \rho_1, \rho_3$ respectivamente, deve-se representar o seguinte $< \bar{\omega}_1, \rho_1 >, < \bar{\omega}_3, \rho_3 >$. Qualquer intervalo $\bar{\omega}_i$ não representado toma o valor do último intervalo, com índice menor que $\bar{\omega}_i$, representado. O último intervalo, quer tenha uma probabilidade associada idêntica à probabilidade do intervalo anterior, deve se encontrar representado no histograma. Desta forma é possível identificar o número total de sub-intervalos. Sendo $\bar{\omega}_n$ o último intervalo, o número de sub-intervalos, N é dado por $n + 1$. A figura 5.1 ilustra a representação das métricas ABH e DH.

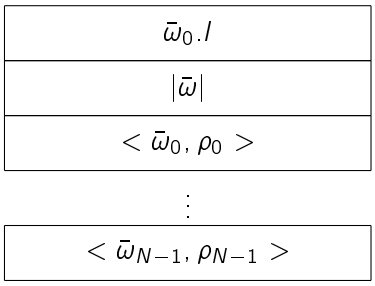

Figura 5.1: *Exemplo da representação das métricas ABH e DH*

5.3.2 A Métrica IPLR

A métrica IPLR, como referido anteriormente, retrata a taxa de perdas de pacotes e pretende representar a probabilidade de uma rota, ou parte de uma

rota, perder, ou descartar, um determinado pacote.

Foi considerado utilizar, como métrica referente à taxa de perdas de pacotes, a representação em histograma desta mesma métrica, no entanto optou-se pela não utilização desse tipo de representação. A taxa de perdas de pacote relaciona-se directamente com a taxa de erros de *bit* [38], com falhas nas máquinas e descarte de pacotes devido a congestão dos canais [39]. Os erro de *bit* apresentam uma natureza impulsiva, pelo que modelos mais actuais prevêem a modelação deste tipo de erros, não segundo as distribuições apresentadas em [3] e [37], mas com modelos baseados em estados, prevendo estados com erro e estados sem erro, como são exemplo [40] e [41]. Esta natureza impulsiva dos erros de *bit* influencia a taxa de perdas de pacote [38]. A implementação destes modelos de erros resulta numa probabilidade, aproximada, de perda de pacote, pelo que utilização desta valor parece, ao autor, adequada para representação deste atributo.

A métrica IPLR tem uma representação bastante simples, em relação às restantes métricas sugeridas no presente documento. A sua representação não é mais do que o valor do taxa de perdas de pacotes, como ilustra a figura 5.2.

Figura 5.2: *Exemplo da representação da métrica IPLR.*

5.3.3 A Métrica Segurança de um Caminho

Na exposição das métricas, realizada na secção anterior, referiu-se uma métrica, a segurança de caminho, que se defende a utilização na solução proposta. Existem diversos trabalhos defendendo a inclusão de métricas relativas à segurança de uma rota, como são exemplo [42] e [43]. Não sendo a questão

da segurança de redes IP o foco do trabalho a que se refere o presente documento, sugere-se uma estratégia para transportar a informação de segurança de um caminho e agregar informação de duas ou mais rotas.

A métrica relativa à segurança de um caminho denomina-se, na solução apresentada, por `PS` (*Path Security*[1]). A informação de segurança é relativa a parâmetros de segurança, que embora referidos no presente trabalho, não são especificados. Propõe-se, então, que o atributo seja composto por um conjunto de parâmetros, onde cada parâmetro representa um parâmetro de segurança da rede. A figura 5.3 ilustra como se constitui a métrica `PS`.

Figura 5.3: *Exemplo da representação da métrica `PS`*

Apesar de não se pretender apresentar qualquer tipo de sugestão para os parâmetros em questão, estes podem ser relativos à confidencialidade do um caminho, à autenticidade, à segurança, ou mesmo quanto à segurança das técnicas utilizadas.

[1]Segurança do Caminho em português.

5.4 Extensão das Mensagens UPDATE

No protocolo BGP, as mensagens UPDATE são a forma de anunciar rotas, e com estas rotas, os parâmetros de QoS. Estas mensagens devem portanto ser estendidas por forma a incluir a informação relativa às métricas de QoS propostas no presente documento. Em [37] sugere-se, conforme descrito na secção 4.6.4.1 do presente documento, que para lidar com as ligações do tipo TYPE-3, deve ser criado um atributo que contenha o endereço do ultimo encaminhador que implementa a solução proposta. Este atributo, apesar de apresentado em [37] como optional transitive, não está descrito como é representada a informação relativa ao endereço do último par. O autor do presente documento considera que o atributo tem um papel muito preponderante na descoberta do tipo de ligação entre os pares, pelo que defende-se a sua inclusão na solução proposta.

5.4.1 O Atributo QoS-Peer

A extensão, ao protocolo BGP, que permite o encaminhamento de tráfego inter-dom+inio com informações de QoS deve, como amplamente referido acima, ser capaz tratar ligações onde os pares não implementam a mesma solução. Tendo como ponto de partida o atributo defendido em [37] e apresentado na secção anterior (5.4), sugere-se nomear o atributo de QoS-Peer[2]. Este atributo, como referido acima, indica o endereço do par que implementa a solução proposta, mais próximo do par que recebe a mensagem, para uma determinada rota. Caso um encaminhador receba uma mensagem UPDATE, onde o atributo QoS-Peer não transporte o endereço do par que envia a mensagem, o encaminhador que recebe a mesma está em posição de assumir

[2]QoS-Peer ou par de QoS em português.

que a ligação entre os dois pares é uma ligação do tipo `TYPE-3`. Como referido em [37], este atributo deve ser definido como `optional transitive`. Propõe-se, no presente documento, que a informação relativa ao endereço do último par, contida no atributo, seja representada no mesmo, da mesma forma que o campo `NLRI`, de uma mensagem `UPDATE`, é representado no padrão do BGP. Como se encontra exposto na secção 2.4.1.2, o representa-se no início o número de *bits* que compõem o endereço e de seguida. O atributo `QoS-Peer` difere da representação exposta em 2.4.1.2, no sentido em que, o atributo apenas é composto por um endereço. O facto de este ser um endereço de um encaminhador, pode levar a pensar na não necessidade de introduzir a informação relativa ao número de *bits* que compõem o endereço. No entanto esta informação permite, entre outras coisas, descobrir se o par tem um endereço IPv4 ou IPv6.

O atributo é então composto por dois campos, sendo eles:

Length O campo `Length`, que ocupa um octeto, contém a informação relativa ao número de *bits* que compõem o endereço.

Prefix O campo `Prefix`, que tem tamanho variável, contém o endereço do último par, de uma rota, que implementa a solução proposta.

Uma ilustração do atributo pode-se verificar na imagem 5.4.

Figura 5.4: *Codificação do atributo `QoS-Peer` numa mensagem `UPDATE`*

5.4.2 O atributo `QoS_NLRI`

O anúncio de rotas com informações de QoS é um tópico defendido na grande maioria dos trabalhos estudados, em [37] sugere-se que exista um atributo `Optional transitive` criado para o efeito. Em [34] e [1] apresenta-se o atributo `QoS_NLRI`, originalmente apresentado em [36]. Este atributo, estendido em [1] para suportar a troca de informações de QoS por classes de serviço, encontra-se descrito na secção 4.5.1 do presente relatório. Como o presente capítulo apresenta métricas de QoS diferentes das apresentadas em [1], este atributo deve também ser modificado para acomodar as métricas aqui defendidas.

O atributo apresentado em [1] divide a informação de QoS por cada classe de serviço. Esta divisão da informação é, no entender do autor do presente documento, bastante adequada ao problema. No entanto é necessário definir os campos do atributo referentes às métricas propostas. Sugere-se que a informação de QoS, relativa a cada classe de tráfego, seja composta pelos seguintes campos:

Identificador da Classe Este campo pretende, como o nome indica, identificar a classe a que se referem os parâmetros de QoS que seguem.

QoS-ABH Informação relativa à métrica ABH.

QoS-DH Informação relativa à métrica DH.

QoS-IPLR Informação relativa à métrica IPLR.

QoS-PS Informação relativa à métrica PS.

A figura 5.5 ilustra os campos acima expostos e a sua ordem. Devido à complexidade dos campos, estes serão descritos nas secções seguintes do presente documento.

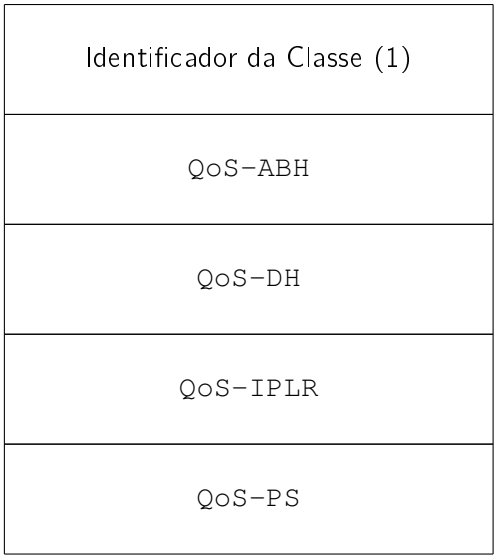

Figura 5.5: *Exemplo da representação das métrica relativas a uma classe de serviço.*

No trabalho apresentado no presente documento, apresenta-se um cabeçalho comum para cada um dos campos acima expostos. Este cabeçalho é composto, como ilustra a figura 5.6, pelos seguintes campos:

ID O campo `ID`, que ocupa um octeto, representa um identificador da métrica presente nos octetos seguintes do campo. Prevê-se a existência dos seguintes identificadores, com a métrica associada:

1. `QoS-ABH`
2. `QoS-DH`
3. `QoS-IPLR`

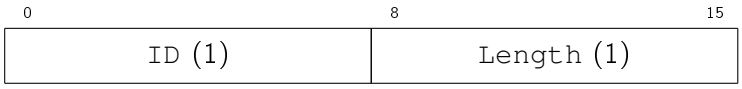

Figura 5.6: *Cabeçalho comum dos campos do atributo* `QoS_NLRI`.

4. `QoS-PS`

Length O campo `Length`, que ocupa um octeto, especifica número de instâncias que compõem a métrica de QoS em questão. Este número de instâncias pode referir-se ao número de intervalos que compõem os histogramas das métricas ABH e DH caso se trate de campos `QoS-ABH` e `QoS-DH`, respectivamente; o número de instâncias, no caso de se tratar de um campo `QoS-PS`, refere-se ao número de parâmetros que compõem a métrica. Caso se trate de um campo `QoS-IPLR`, o número de instâncias deve tomar o valor de 1. O autor optou por introduzir o número de instâncias, e não o número de octetos do campo, pois estas instâncias podem ocupar mais de octeto, permitindo assim que este campo `Length` possa ocupar, por sua vez, um octeto. Devido a esta decisão, o número de instâncias de cada métrica pode ser no máximo duzentos e cinquenta e seis, o que, na opinião do autor, é adequado ao problema. Caso, num trabalho futuro, surja necessidade de alongar este campo, sugere-se que este campo indique o número de octetos seguintes que compõem a informação de tamanho.

5.4.2.1 Os Campos `QoS-ABH` e `QoS-DH`

No atributo `QoS_NLRI`, proposto no presente documento, os campos `QoS-ABH` e `QoS-DH` representam a informação relativa à métrica ABH e DH respectivamente. Os campos que compõem estas métricas, ilustrados na figura 5.1,

são transpostos para os campos `QoS-ABH` e `QoS-DH` do atributo `QoS_NLRI`. Desta forma cada campo é composto, como se ilustra na figura 5.7, pelos seguintes campos:

Valor Inicial Este campo indica o valor inicial do primeiro sub-intervalo que compõe a métrica em questão. Este campo é o correspondente de $\bar{\omega}_0.l$, definido em 5.3.1. Este campo ocupa quatro octetos.

Largura dos Intervalos Este campo especifica a largura de cada sub-intervalo do histograma referente à métrica em questão. Corresponde a $|\bar{\omega}|$, definido em 5.3.1. Este valor ocupa quatro octetos.

Índice Este campo especifica o índice do sub-intervalo em questão, correspondendo a $\bar{\omega}_i$ como definido na secção 5.3.1. Este valor ocupa um octeto, pois o número máximo de instâncias permitido pelo campo `Length` do cabeçalho geral dos campos `QoS-ABH` e `QoS-DH` é 255.

Probabilidade Este campo indica a probabilidade relativa ao intervalo indicado em `Índice`, ρ_i, como exposto em 5.3.1. Este valor ocupa um octeto. Devido à probabilidade variar entre 0 e 1, e tendo em conta que este campo pode apresentar 256 valores diferentes, o valor da probabilidade associada, ρ_i, deve ser calculado segundo a equação 5.1, onde P representa o valor do campo `Probabilidade`.

$$\rho_i = \frac{P}{255} \tag{5.1}$$

Esta codificação permite obter uma resolução de sensivelmente 0.004, ou 0.4%, o que, na opinião do autor, trata-se de um valor bastante razoável.

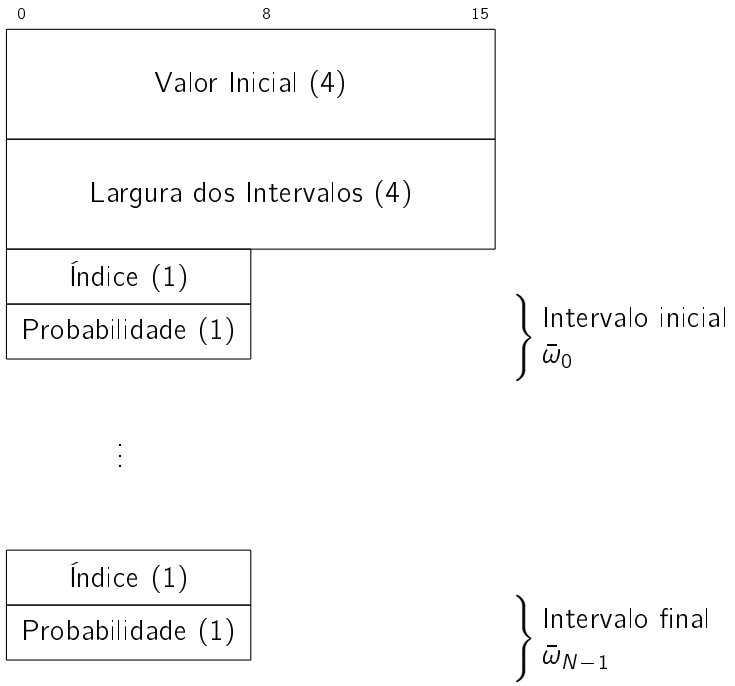

Figura 5.7: *Composição dos campos* `QoS-ABH` *e* `QoS-DH` *do atributo* `QoS_NLRI`.

Como referido na secção 5.3.1, é possível agregar valores de intervalos consecutivos de probabilidades iguais. Devido ao valor da resolução ser inferior ao valor do erro introduzido pela operação de junção desta métrica, que será apresentada numa secção seguinte, pode-se equacionar a agregação de intervalos cujas probabilidades não sejam aritméticamente iguais, mas sim, com uma determinada e reduzida diferença. Defende-se que essa decisão deve ser tomada pelas várias implementações da solução, no entanto, sugere-se que intervalos consecutivos, $\bar{\omega}_1, \bar{\omega}_2$, com probabilidades ρ_1, ρ_2, respectivamente, possam ser agregados sempre que se verificar a equação 5.2, ou seja, sempre que exista uma diferença de 10% entra as probabilidades.

$$0.9 \leq \frac{\rho_2}{\rho_1} \leq 1.1 \tag{5.2}$$

No entanto desaconselha-se a realização desta aproximação mais que duas vezes consecutivas, pois corre-se o risco de descaracterizar as distribuições dos histogramas.

5.4.2.2 O Campo `QoS-IPLR`

O campo `QoS-IPLR`, como o nome indica, transporta informação relativa à métrica IPLR. Esta métrica representa, conforme exposto na secção 5.3.2, a taxa de perdas de pacotes.

A métrica IPLR varia entre os valores 0 e 1. Se determinadas aplicações, como é exemplo o VoIP, permitem algumas perdas de pacotes, uma taxa de perdas de pacotes de 10% pode ser suficiente para degradar significativamente uma comunicação de qualquer tipo de aplicação [44]. Desta forma, o rigor, e a resolução, do campo `QoS-IPLR`, tem mais influência em valores reduzidos de

IPLR, do que em valores não reduzidos – acima de 10%. Pretende-se atribuir um octeto para representar este valor.

Distribuindo uma probabilidade, entre 0 e 1, por um octeto, obtêm-se uma resolução de sensivelmente 0.004, ou 0.4%. Apesar de este valor não ser, na opinião do autor, curto para a métrica em questão, é possível, sem elevado custo computacional, obter uma maior resolução para valores mais reduzidos de IPLR. Propõe-se que existam duas escalas de IPLR, uma escala para valores de IPLR entre 0% e 10% e outra escala para valores superiores a 10% e até 100%. Atribuindo o primeiro *bit* do octeto para definir a escala, restam 7 *bits* para representar o valor de IPLR. Esta estratégia permite uma resolução de, sensivelmente, 0.08 pontos percentuais para valores entre 0% e 10% e uma resolução de, aproximadamente, 0.7 pontos percentuais para valores de IPLR superiores a 10%. Com esta estratégia aumenta-se a resolução, da zona de valores críticos de IPLR, mais de cinco vezes, reduzindo a resolução da zona onde qualquer aplicação sofrerá os efeitos do elevado IPLR.

Para permitir este comportamento, propõe-se que, o atributo `QoS-IPLR` seja composto, como ilustra a figura 5.8, pelos seguintes campos:

Escala O primeiro *bit* do octeto representa então a escala em que se encontra o valor representados nos *bits* seguintes. Este *bit* segue a seguinte codificação:

- **0** O valor representado nos restantes *bits* encontra entre 0% e 10%.
- **1** O valor representado nos restantes *bits* é superior a 10%.

IPLR Os restantes *bits* representam o valor do IPLR na escala indicada. O valor de IPLR é calculado da seguinte forma:

- Caso o valor seja inferior a 10% o valor do IPLR, representado entre 0 e 1, é dado pela equação 5.3, onde P representa o valor dos sete *bits* que compõem este campo.

$$IPLR = \frac{0.1P}{127} \tag{5.3}$$

- Caso o valor seja superior a 10% o valor do IPLR, representado entre 0 e 1, é dado pela equação 5.4, onde P representa o valor dos sete *bits* que compõem este campo.

$$IPLR = \frac{0.9P}{127} + 0.1 \tag{5.4}$$

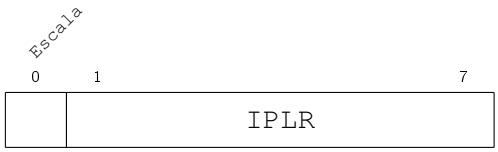

Figura 5.8: *Codificação do campo `QoS-IPLR` no atributo `QoS_NLRI`*

5.4.2.3 O Campo `QoS-PS`

O campo `QoS-PS`, de um atributo `QoS_NLRI`, contém a informação relativa à métrica PS. Esta métrica, como referido acima, refere-se à segurança de uma rota, não se pretendendo, no presente documento, especificar os parâmetros de segurança, apenas sugerir a forma de os incorporar no atributo `QoS_NLRI`. Como descrito na secção 5.3.3, a métrica PS é composta o por um conjunto de parâmetros de segurança, sendo o número destes parâmetros indicado no

campo `Length`, no cabeçalho comum dos campos que compõem o atributo `QoS_NLRI`. Sugere-se que cada parâmetro seja composto, como ilustra a figura 5.9, pelos seguintes campos:

Header O campo `Header`, que ocupa um octeto, tem dois objectivos: Especificar o parâmetro de segurança em questão e o número de octetos que compõem o seu valor. Para tal este octeto é dividido em dois campos de quatro *bits* cada, sendo eles:

ID Os quatro primeiros *bits* do campo `Header` representam o campo `ID`. Este campo é um identificador do parâmetro de segurança em questão. Segundo esta codificação é possível especificar dezasseis parâmetros de segurança diferentes, o que na opinião do autor, é adequado ao problema.

Length Os quatro *bits* menos significativos do campo `Header` especificam o tamanho em octetos do campo `Value`. Este campo permite especificar um tamanho máximo de quinze octetos para o campo `Value`. Pode, eventualmente, existir um parâmetro de segurança com necessidade de mais que quinze octetos para representação do seu valor, sendo que nesse caso, desde que conste na especificação do parâmetro de segurança, alguns octetos do campo `Value` podem conter informação quanto ao tamanho do mesmo.

Value O campo `Value` contém a informação relativa ao valor do parâmetro de segurança. A codificação deste campo deste campo deve ser referida na especificação de cada parâmetro de segurança.

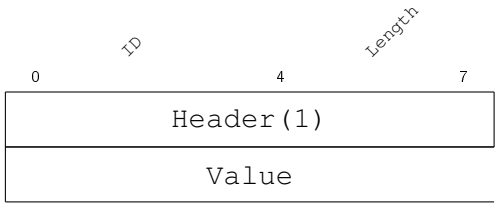

Figura 5.9: *Codificação dos parâmetros do campo `QoS-PS` no atributo `QoS_NLRI`*

5.5 Operação Junção das Métricas de Qos

Uma área crítica, na definição das métricas de QoS, é especificar a operação de junção das métricas referidas. A correcta definição desta operação é essencial para que a métrica de uma rota represente o que na verdade acontece nessa mesma rota. No trabalho proposto apresentam-se quatro métricas, cujas associações são de tipos diferentes:

- A métrica IPLR é uma métrica multiplicativa.
- As métricas ABH PS são métricas côncavas.
- A métrica DH é uma métrica aditiva.

Tendo em conta a natureza das métricas, não é possível definir uma operação de junção de métricas, $m_r = m1 \oplus m2$. Esta operação deve portanto ser especificada para cada tipo de métrica. As métricas PS e ABH, apesar de ambas serem métricas côncavas, têm naturezas distintas, pois a métrica ABH é uma representação estatística da distribuição da largura de banda, logo devem também ter operações de junção distintas.

Quando se apresentam as métricas ABH e DH, em [37], apresentam-se também propostas para a operação de junção destas métricas. Quanto à

métrica IPLR, sugerida em [1], defende-se no trabalho referido que a operação de junção desta métrica pode passar pela simples soma dos vários IPLR. Do ponto de vista do autor, esta operação não se adequa, pois, mesmo podendo não introduzir um erro significativo, o IPLR é uma métrica multiplicativa, nesse caso a operação de junção deve reflectir esse facto.

As operações de junção de métricas, que se apresentam no presente documento, são independentes do tipo de ligação entre os encaminhadores, sejam elas tipo `TYPE-1`, `TYPE-2` ou `TYPE-3`. A forma como o tipo de ligação influencia as métricas, está na aquisição das mesmas. Em [37], defende-se que as métricas devem ser adquiridas segundo o tipo de ligação e da seguinte forma:

TYPE-1 Numa ligação do tipo `TYPE-1`, os parâmetros de QoS obtêm-se através de monitorização directa, pois este tipo de ligações representa uma ligação directa entre dois encaminhadores que implementam a solução proposta.

TYPE-2 Numa ligação do tipo `TYPE-2`, as métricas de QoS são obtidas através dos protocolos de IGP.

TYPE-3 Numa ligação do tipo `TYPE-3` as métricas de QoS, devido a não haver uma ligação directa, nem por IGP, devem ser obtidas por medição de QoS de fim-a-fim. Em [37] não se apresentam soluções, nem sugestões, para a medição de QoS de fim-a-fim. O trabalho a que se refere o presente documento, não pretende, também, sugerir forma de efectuar esta medição, pois considera-se que esta medição é da responsabilidade de cada implementação da solução. No entanto identificaram-se eventuais soluções para este problema, entre elas [45] e [46].

Sugere-se, que para a solução proposta no presente documento, se utilizem as estratégia de monitorização das métricas acima definidas. Cada encaminhador deve conhecer as métricas relativas à ligação lógica com cada par que implemente a solução proposta. Assim que receba um mensagem `UPDATE` um encaminhador deve associar as métricas das sua ligação com as métricas presentes na mensagem. Esta associação deve ser feita no encaminhador que recebe a mensagem pois, por entre outros motivos, um encaminhador que implemente a solução proposta pode não saber se os seus pares implementam, por sua vez, o encaminhamento com QoS. A operação de junção das métricas referidas é diferente para cada métrica sendo apresentadas de seguida as várias operações.

5.5.1 Operação de Junção da Métrica ABH

A métrica ABH, bem como a sua operação de junção encontram-se expostas em [37] e nas secções 4.6.5.1 e 4.6.5.2 do presente documento, respectivamente. A operação de junção da métrica ABH baseia-se na fórmula para o função distribuição acumulada do mínimo de duas variáveis aleatórias. Esta função encontra-se na equação 5.5, onde $F_b(x)$ representa a função distribuição acumulada resultante do mínimo de duas variáveis aleatórias cujas respectivas funções distribuição acumuladas são $F_{b1}(x)$ e $F_{b1}(x)$.

$$F_b(x) = F_{b1}(x) + F_{b2}(x) - F_{b1}(x)F_{b2}(x) \tag{5.5}$$

A equação 5.5 é a base para obtenção do algoritmo apresentado na listagem 4.1 presente na secção 4.6.5.2 do presente documento e replicada de seguida.

```
1 s ← min{i : abh_1.ρ_i > 0 ou abh_2.ρ_i > 0}
  t ← max{i : abh_1.ρ_i > 0 ou abh_2.ρ_i > 0}
```

```
3 ω1 ← 0.0
  ω2 ← 0.0
5 para i ← s até t
  fazer
7       abh.ρi ← (1 − ω2)abh1.ρi + (1 − ω1)abh2.ρi − abh1.ρi.abh2ρi
        ω1 ← ω1 + abh1.ρi
9       ω2 ← ω2 + abh2.ρi
```

Listing 5.1: *Algoritmo para calcular* $abh = abh1 \oplus abh2$

5.5.2 Operação de Junção da Métrica DH

A métrica DH e sua operação de junção são definidas no trabalho apresentado em [37]. No presente documento, a operação de junção da métrica DH está descrita na secção 4.6.5.2.

A operação de junção do DH baseia-se na propriedade da soma de duas variáveis aleatórias, visto o atraso ser uma métrica aditiva. A distribuição , $F_r(x)$, resultante da soma de duas variáveis aleatórias com distribuições $F_1(x)$ e $F_2(x)$ é dada pela convolução das funções de distribuição $F_1(x)$ e $F_2(x)$. A equação 5.6 ilustra a esta propriedade.

$$F_r(x) = \int_{-\infty}^{+\infty} F_1(x-t)F_2(t)dt \tag{5.6}$$

Da equação 5.6 obtém-se a formula para, computacionalmente, realizar a operação de junção do DH, $DH = DH_1 \oplus DH_2$, onde DH é o resultado da junção de DH_1 com DH_2. Definindo DH^i como o valor da probabilidade, ρ, do sub-intervalo de índice i de um DH, a operação de junção de DH, $DH =$

$DH_1 \oplus DH_2$ está apresentada na equação 5.7.

$$DH^i = \sum_{j=0}^{i} DH_1^j DH_2^{(i-k)} \tag{5.7}$$

Para reduzir a carga causada pelo cálculo da equação 5.7, sugere-se aplicar o algoritmo FFT (*Fast Fourier Transform*) na implementação da solução proposta.

5.5.3 Operação de Junção da Métrica IPLR

A métrica IPLR, como referido acima, é uma métrica multiplicativa, no entanto a operação de junção não se limita à simples multiplicação de dois valores. O valor da métrica IPLR, corresponde, à razão entre pacotes enviados e pacotes perdidos, representando também uma probabilidade teórica de se perder um determinado pacote. Analogamente, a probabilidade de não se perder um determinado pacote é dada por $1 - IPLR$. Desta forma, a probabilidade de não se perder um pacote em duas rotas, com IPLR's distintos, $IPLR_1$ e $IPLR_2$, é dada por $(1-IPLR1)(1-IPLR_2)$. Isto permite obter a fórmula para a probabilidade de perda de pacotes em duas rotas com IPLR's distintos. A operação de junção de dois IPLR, $IPLR_1$ e $IPLR_2$, $IPLR = IPLR_1 \oplus IPLR_2$, encontra-se definida na equação 5.8.

$$IPLR = IPLR_1 + IPLR_2 - IPLR_1 IPLR_2 \tag{5.8}$$

Quando um encaminhador efectua esta operação deve ter em atenção a escala de valores da métrica IPLR. Da mesma forma, é necessário, sempre que o valor resultante da operação de junção ultrapasse os 10%, sem que nenhum dos operandos o faça, alterara a escala do valor do IPLR.

5.5.4 Operação de Junção da Métrica PS

A métrica PS apresenta-se como uma métrica côncava. Isto na medida em que, qualquer parâmetro de segurança em questão, a segurança de uma rota é dada pelo menor segurança de qualquer parte da rota. Quaisquer parâmetros que venham a ser especificados devem ter em conta este facto.

A operação de junção da métrica PS deve ser efectuada parâmetro a parâmetro. Desta forma, para cada parâmetro, PS^i, de uma métrica a sua operação de junção, $PS^i = PS_1^i \oplus PS_2^i$ é dada pela equação 5.9.

$$PS^i = min(PS_1^i, PS_2^i) \tag{5.9}$$

A métrica PS está representada no campo `QoS-PS` no atributo `QoS_NLRI`. Este atributo é definido como `Optional transitive`, por forma a lidar com os encaminhadores que não implementam a solução apresentada. No entanto pretende-se um comportamento ligeiramente diferente para a métrica PS. É possível que existam parâmetros de segurança apenas utilizados em certas implementações, nesse caso, na operação de junção da métrica PS, um parâmetro que esteja apenas representado em parte de uma rota, sugere-se que deva ser retirado da informação da rota. Este comportamento não deve ter influencia no caso de um encaminhador estar perante uma ligação do tipo `TYPE-3`. Nesse caso, como referido acima, o encaminhador deve iniciar um processo de monitorização dos parâmetros de segurança, de fim-a-fim, com o último encaminhador, numa determinada, rota que implementa a solução proposta.

5.6 Armazenamento de Rotas

Na norma do protocolo, especificam-se três tabelas que, conceptualmente, compõem as tabelas de encaminhamento do BGP. Estas tabelas são a `Loc-RIB`, a `Adj-RIB-In` e a `Adj-RIB-Out`. Em [1] sugere-se a introdução de uma tabela de encaminhamento por cada classe de serviço. Esta decisão entende-se como necessária, para permitir o BGP encaminhar o tráfego consoante a classe de tráfego, no entanto, criar N `Adj-RIB-In`, consoante o número de classes pode não ser suficiente.

Sugere-se, no presente documento, que deve existir uma instância de cada tabela por cada classe de tráfego. De uma forma geral cada `Adj-RIB-In`, `Adj-RIB-Out` e `Loc-RIB` devem ser divididas em múltiplas tabelas, uma por cada classe. Desta forma, um encaminhador que tenha ligações activas com três pares e tenha informação de quatro diferentes classes de serviço a anunciar e receber de cada par, tem um total de 12 `Adj-RIB-In` (quatro classes repetidas por três pares), 12 `Adj-RIB-Out` e quatro `Loc-RIB` (uma para cada classe). Estas tabelas são apenas conceptuais, pois podem existir implementações onde hajam tabelas colapsadas, apontadores para tabelas diferentes, etc. No entanto, para facilitar a exposição da solução não se sugere nenhum desses processos.

Cada uma das tabelas acima referidas deve conter, para além da informação já nela contida, a descrição da classe de serviço a que se refere, a informação dos atributos `QoS-Peer` e `QoS_NLRI`.

5.7 Selecção e Anuncio de Rotas

O processo de selecção e disseminação de rotas do protocolo BGP encontra-se descrito na secção 2.4.5 do presente documento. Este processo, ou conjunto de processos, deve ser alterado por forma a suportar o encaminhamento por classes de tráfego, com informações de QoS. De uma forma geral, os objectivos das três fases referidas na secção 2.4.5 mantêm-se os mesmos, no entanto, devem incluir as métricas de QoS para cada classe. As várias alterações sugeridas para as fases do processo de selecção do BGP encontram-se expostas de seguida.

5.7.1 Fase 1: Cálculo do Grau de Preferência

Na fase 1 do processo de selecção os encaminhadores elegem as rotas que passam à fase seguinte. Cada encaminhador deve inspeccionar o atributo `AS_Path` por forma a detectar ciclos na rota. Sugere-se, no presente documento duas alterações nesta fase: Filtragem de rotas que não cumpram requisitos e QoS e alteração do cálculo do grau de preferência. Estas alterações serão descritas na seguinte secção.

5.7.1.1 Filtragem de Rotas

Quando um encaminhador recebe uma mensagem `UPDATE`, esta mensagem contém a informação de QoS até ao último encaminhador que implementa a solução proposta. O encaminhador que recebe a mensagem `UPDATE` deve então, juntar as métricas de QoS da ligação com o último par às métricas anunciadas numa mensagem. Esta operação pode degradar algumas métricas, fazendo com que uma rota não cumpra os requisitos de uma classe de serviço.

Desta forma, um encaminhador, após receber uma mensagem `UPDATE`, aplicar as políticas e verificar os cíclos, deve efectuar as operações de junção das métricas de QoS e apenas colocar nas `Adj-RIB-In` as rotas que, para uma determinada classe, cumprem os requisitos de QoS.

Na solução proposta defende-se que as métricas PS e IPLR tenham apenas influência nas extensões propostas nesta secção. Defende-se que não se deve entrar com as referidas métricas para o cálculo do melhor caminho, mas apenas para filtrar as rotas elegíveis para uma classe com determinados requisitos de IPLR e PS. Este facto não significa que as métricas ABH e DH não sejam, também utilizadas para filtrar rotas, pois nesta fase as quatro métricas propostas são utilizadas.

5.7.1.2 Cálculo do Grau de Preferência de uma Rota

Na actual norma do BGP existe uma função para o cálculo do grau de preferência de uma rota. Caso uma rota tenho sido descoberta por I-BGP, o valor do grau de preferência toma o valor do atributo `Local-Pref`. Caso uma rota tenha sido descoberta por E-BGP, o valor do grau de preferência é obtido através das políticas de encaminhamento. Sugere-se que, por forma a permitir ao BGP encaminhar tráfego, pelo caminho que cumpre melhor determinados requisitos de QoS, é necessário modificar significativamente a fórmula usada pelo BGP para cálculo do grau de preferência. Sugere-se também, no presente documento, que os encaminhadores que implementam a solução proposta utilizem apenas os atributos `Local-Pref` e `Multi-Exit-Disc` quando for estritamente necessário dar um maior grau de preferência a determinadas rotas, pois o valor destes atributos sobressai sobre o cálculo do grau de preferência

descrito abaixo. Sugere-se portanto, que o valor dos atributos referidos seja 0.

No trabalho apresentado em [37], defendem-se duas fórmulas para o cálculo do peso de rotas conforme os valores de ABH e DH, no entanto não existe uma fórmula que relacione estas duas métricas. Nas equações 5.10 e 5.11 encontram-se as fórmulas sugeridas em [37] para cálculo dos pesos de uma rota segundo a métrica ABH, W_b e segundo a métrica DH, W_d, respectivamente. Nestas equações, **E** representa o operador esperança matemática, **STD** representa o operador desvio padrão e η_b e η_d representam parâmetros que permitem dar mais importância ao valor médio ou à variação das métricas no peso calculado.

$$W_b = \mathbf{E}[abh] - \eta_b \mathbf{STD}(abh) \tag{5.10}$$

$$W_d = \mathbf{E}[dh] - \eta_d \mathbf{STD}(dh) \tag{5.11}$$

O autor do presente documento entende que é necessário existir uma fórmula que tenha em conta as duas métricas para calcular o melhor caminho. Entende-se também, que para determinadas classes de serviço a métrica ABH possa ter um peso superior à métrica DH, e que para outras classes esses papéis possam se inverter. Sugere-se, portanto, no presente documento, que existam diferentes fórmulas para diferentes classes de serviço, tendo em conta os requisitos de QoS das mesmas.

Um parâmetro que em todos os trabalhos relacionados estudados, sobre extensões de QoS para o BGP, não foi incorporado no cálculo do grau de preferência das rotas, foi o tamanho do atributo `AS_Path` – `AS_Path-Len`. As soluções propostas nos trabalhos estudados sugerem algumas métricas que ultrapassam por completo o parâmetro `AS_Path-Len`. O autor do presente

documento entende que o número de AS's que compõem uma rota, não só traz informação útil, como pode até desempenhar um papel muito importante no cálculo do grau de preferência. Entende-se também que a não utilização do atributo `AS_Path` para cálculo do melhor caminho é descaracterizar, em demasia, o protocolo BGP. Defende-se a incorporação do atributo `AS_Path` pelos seguintes motivos:

1. Quanto maior o número de AS's, geralmente, é maior o número de encaminhadores que compõem uma rota, o que, devido às aproximações nas operações de junção, traz mais incerteza às métricas de QoS.

2. Com o aumento do número de AS's, tendencialmente, tendem a aumentar os AS's que não implementam a solução proposta e com isso as ligações do tipo `TYPE-3`. Este facto aumenta também a incerteza quanto ao valor das métricas e a possibilidade de, simplesmente, não existir encaminhamento com QoS, pois os encaminhadores-fronteira não o realizam.

3. Numa rota com muitos AS's, a probabilidade de um encaminhador estar com problemas ou a descartar pacotes é maior.

4. Devido à natureza côncava da métrica ABH, quanto maior for número de AS's de uma rota, mais "à esquerda" se vai colocar a distribuição da probabilidade do ABH [47] [48].

Pelo acima exposto, sugere-se que o parâmetro `AS_Path-Len` seja incluído no cálculo do grau de preferência de uma rota. Defende-se que este parâmetro seja incorporado na fórmula de cálculo do grau de preferência de uma rota, DoP, obtendo-se a equação 5.12, onde:

- W_b representa o peso da métrica ABH para o grau de preferência da rota. Calculado pela equação 5.10.

- β_b representa um parâmetro para especificar o peso de W_b em DoP.

- W_d representa, analogamente a W_b, o peso da métrica DH para o grau de preferência da rota calculado pela equação 5.10.

- β_d representa, de forma semelhante a β_b, um parâmetro para especificar o peso de W_d em DoP.

- N representa o número de AS's que compõem um `AS_Path`.

- β_a representa, analogamente a β_b e β_d, um parâmetro que permite definir o peso de N em DoP.

$$DoP = \beta_b W_b + \frac{\beta_d}{W_d} + \frac{\beta_a}{N} \tag{5.12}$$

Como apresentado na equação 5.12, os pesos W_b e W_d, bem como N, encontram-se, cada um deles, acompanhado por um factor. Este factores permitem obter equações diferentes, o que se torna importante, no caso de classes de serviço com requisitos de QoS distintos. Conjugando as equações 5.12, 5.10 e 5.11, conclui-se que, para além dos valores das métricas de QoS é necessário especificar os parâmetros η_b, β_b, η_d, β_d e β_a. Defende-se que este parâmetros devem ser especificados para classe, no entanto, no presente documento, não se apresenta nenhuma proposta para a definição dos parâmetros. Defende-se que devem ser realizados testes numéricos intensivos à formula, por forma a obter os parâmetros, ou relação entre parâmetros, relativos aos vários comportamentos desejados.

Volta-se a referir, porém, que as métricas IPLR e PS não entram na fórmula do grau de preferência de uma rota, apenas na filtragem de rotas para entrada na `Adj-RIB-In`.

5.7.2 Fase 2: Selecção de Rotas

A fase 2 do processo de selecção de rotas do BGP tem como objectivo seleccionar as rotas das RIB de entrada para a RIB local. Para a execução desta fase não são apresentadas modificações, no entanto o comportamento é ligeiramente diferente, devido a modificações já referidas.

Em primeiro lugar, devido à fórmula do grau de preferência aplicada a cada rota, muito dificilmente existiram várias rotas empatadas nesta fase. No entanto, caso existam rotas empatadas, sugere-se que se apliquem os critérios de desempate da norma do BGP. Outro comportamento diferente, é a existência de várias `Loc-RIB` por encaminhador, o que implica que se efectue esta fase mais vezes que o indicado na norma do BGP.

5.7.3 Fase 3: Disseminação de Rotas

A fase 3 do processo de selecção de rotas consiste na disseminação das rotas, através de mensagens `UPDATE`. Esta fase é despoletada sempre que:

- Termine a fase 2.
- Se estabeleça uma nova conexão BGP.
- Rotas na `Loc-RIB` se alterem.
- Rotas internas geradas por protocolos que não o BGP se alterem.

Devido à natureza dinâmica das métricas de QoS, mesmo utilizando a representação estatística das mesmas, a fase 3 pode ocorrer bastante mais vezes que o previsto na norma do BGP. Esta número de ocorrências degrada substancialmente a estabilidade do protocolo BGP. Devem, portanto, ser introduzidas alterações com o objectivo de reduzir o numero de mensagens `UPDATE` enviadas.

Sempre que uma nova ligação BGP se estabeleça esta fase deve iniciar. Este comportamento não deve ser alterado, no entanto, nos outros acontecimentos que despoletam a fase 3, deve ser testado se existem alterações suficientes, nas métricas e nos atributos BGP, para anunciar uma rota.

No presente documento sugere-se que exista uma margem até à qual, qualquer alteração no QoS de uma rota não seja considerada significativa. Quer isto dizer que, sempre que uma rota, em qualquer das RIB, altera os seus parâmetros de QoS numa proporção menor que margem especificada, o processo de decisão do BGP deve ocorrer normalmente, mas não se devem enviar mensagens `UPDATE`. No entanto, sempre que esta alteração for suficiente para seleccionar outra rota para a `Loc-RIB`, ou alterar qualquer outro atributo BGP, ou mesmo a métrica PS, devem ser enviados os anúncios.

5.8 Conclusão

No presente capítulo foi apresentada uma solução para o encaminhamento inter-domínios com classes de serviço. Esta solução baseia-se na utilização da representação por histogramas dos valores da largura de banda e atraso. O autor entende que se apresentou uma solução que complementa os trabalhos apresentados em [37], [1] e [34], para além do que se modelou a solução, por forma a permitir a sua implementação, por exemplo, num simulador de rede.

Conclui-se também que, para a solução proposta, é necessário alterar áreas sensíveis do protocolo BGP, como o processo de decisão e as tabelas de encaminhamento, não sendo possível criar apenas um atributo novo, por exemplo.

6 Conclusão

6.1 Síntese do Trabalho Desenvolvido

O trabalho desenvolvido, no âmbito da dissertação a que se refere o presente documento, encontra-se descrito nos capítulos anteriores e sintetizado no presente.

Foi realizada uma introdução ao protocolo BGP, bem como um estudo sobre o mesmo, em termos de comportamento geral, mensagens trocadas, armazenamento e selecção de rotas. Este estudo foi parte importante e fundamental para o desenvolvimento de uma bancada de teste do BGP no Simulador NS-3. Este desenvolvimento é composto pela implementação do protocolo BGP neste simulador, e encontrando-se descrita a integração da implementação no simulador no capítulo 3.

Numa fase posterior à implementação do protocolo BGP, foi efectuado um estudo sobre o encaminhamento inter-domínio com informações de qualidade de serviço. No capítulo 4 apresentam-se as propostas que autor entendeu mais significativas, bem como a base para o que se seguiu. Após o estudo realizado, foi concebida uma proposta para o encaminhamento inter-domínios

com classes de serviço. Esta proposta encontra-se descrita no capítulo 5.

6.2 Contribuições

Como referido na secção anterior, as maiores contribuições para a comunidade académica são a implementação de uma bancada de teste para o BGP no simulador NS-3 e a concepção de uma proposta para o encaminhamento de tráfego entre domínios com informações de QoS.

O simulador NS-3, versão que pretende substituir, o largamente utilizado e citado, NS-2, não inclui, da mesma forma que o seu antecessor não incluia, suporte para o protocolo BGP. Se para o NS-2 já tinham sido desenvolvida, pela comunidade académica, uma implementação do BGP, no mais recente simulador da família não existia, até à data da escrita do presente documento, nenhuma implementação deste protocolo. Desta forma o autor considera uma contribuição relevante a implementação do protocolo no simulador NS-3.

Durante o estudo efectuado encontraram-se algumas propostas para o encaminhamento inter-domínio com informações de QoS. No entanto, a maioria destas propostas, não referiam algumas partes que, ao autor do presente documento, se identificam como chave para o mesmo. As propostas estudadas para o encaminhamento com classes de serviço, muitas vezes, não referiam que métricas utilizar, como as combinar, como não degradar a estabilidade e escalabilidade do protocolo BGP e como a solução deve interagir com uma implementação clássica do BGP. No entanto, no terceiro trabalho apresentado no capítulo 4, as questões acima levantadas são respondidas, com uma estratégia que, no entender do autor, se adequa bastante ao problema. No entanto, nesse trabalho, apenas se equaciona o encaminhamento segundo uma métrica e não por classes de serviço. Desta forma, a concepção de uma pro-

posta para o encaminhamento inter-domínio com classes de serviço e métricas estatísticas, apresentou-se como o caminho a seguir ao autor do presente documento. Esta proposta encontra-se descrita no capítulo 5. O facto de a proposta ter em consideração dois campos distintos – classes de serviço e métricas estatísticas – onde não foram encontrados trabalhos que conjugassem estes campos, na opinião do autor, define a solução proposta no capítulo 5 como uma contribuição relevante, no âmbito do encaminhamento de tráfego inter-domínios.

6.3 Trabalho Futuro

Devido a questões de tempo, relacionadas com a dissertação ao qual se refere o presente documento, existem algumas áreas que não foram alvo de atenção por parte do autor. Existem ainda outros temas que foram apenas identificados, mas não descrita a sua concepção. Neste capítulo apresentam-se essas áreas.

A implementação do protocolo BGP no simulador NS-3, que está n8um estado funcional, não cumpre as especificações para a escrita de novos módulos do simulador. Este incumprimento está relacionado com a escrita do código e não com o que foi implementado. Desta forma, e para permitir a submissão da implementação e possibilitar a presença deste em futuras versões do simulador, deve ser revisto todo o código-fonte e alterado conforme as especificações do simulador.

A implementação do BGP, teve como ponto de partida a norma do mesmo, conforme [14]. No entanto, existem outras extensões ao BGP, como por exemplo a troca de informações de capacidades ou o espelhamento de rotas. Estas extensões não foram, devido a requisitos de tempo, incorporadas na implementação do BGP. O autor considera que a incorporação de extensões

na implementação do BGP no NS-3 torna a bancada de testes muito mais completa, pelo que se considera a análise e implementação das extensões trabalho a realizar.

Quanto à segunda contribuição do presente documento, alguns parâmetros desta foram deixados um pouco em claro. Um primeiro parâmetro é a métrica PS, referente à segurança de um caminho. Esta métrica deve ser estudada, bem como os eventuais parâmetro que a componham. Principalmente deve-se equacionar que parâmetros de segurança faz sentido utilizar no âmbito do encaminhamento inter-domínios e como se podem relacionar com a segurança implementada nas camadas de rede e transporte.

A proposta apresentada tem como base o encaminhamento com classes de serviço. No entanto, no presente documento não se especificam as classes que devem ser implementadas. Para o correcto funcionamento da solução proposta, entende-se que a definição das classes é importante, bem como a especificação dos requisitos de QoS de cada classe.

Na solução proposta, apresenta-se um fórmula genérica para o cálculo do grau de preferência de uma rota, tendo em conta as várias métricas de QoS sugeridas, representada na equação 5.12, do capítulo 5. Esta fórmula depende de um conjunto de parâmetros que modelam o seu comportamento e não estão especificados no presente documento. O autor entende que é necessário um estudo numérico intensivo a estes parâmetros, por forma a identificar que valores e que relações entre estes podem existir

Por fim, entende-se que se deve implementar e testar a solução proposta na bancada de teste implementada no NS-3. Esta implementação faz todo o sentido, pois permite medir a eficiência da solução proposta, no entanto corresponde a um gasto, em termos de tempo, muito elevado, pelo que não

foi possível incorporar no trabalho a que se refere o presente documento.

Acrónimos

ABH Available Bandwidth Histogram

ABI Available Bandwidth Index

AF Assured Forwarding

AS Autonomous System

ASN Autonomous System Number

DH Delay Histogram

DI Delay Index

DoP Degree of Preference

E-BGP Exterior-BGP

EF Expedited Forwarding

EPF Explicit Path Forwarding

EQ-BGP Enhanced QoS Border Gateway Protocol

FFT Fast Fourier Transform

FTP File Transfer Protocol

HHF Hop by Hop Forwarding

HTTP Hypertext Transfer Protocol

I-BGP Interior-BGP

IEEE Institute of Electrical and Electronics Engineers

IETF Internet Engineering Task Force

IP Internet Protocol

IPTD IP Packet Transfer Delay

IPDV IP Packet Delay Variation

IPLR IP Packet Loss Rate

ISO International Organization for Standardization

LAN Local Area Network

LDC Local Delivery Callback

LTE Long Term Evolution

MFC Multicast Forwarding Callback

NS Network Simulator

OSI Open System Interconnect

OSPF Open Shortest Path First

PS Path Security

PHB Per Hop Behavior

QoS Quality of Service

RIB Routing Information Base

RIP Routing Information Protocol

RSVP The Resource reSerVation Protocol

TCP Transfer Control Protocol

UDP User Datagram Protocol

UFC Unicast Forwarding Callback

Bibliografia

[1] A. Beben, "Eq-bgp: an efficient inter-domain qos routing protocol," in *Proceedings of the 20th International Conference on Advanced Information Networking and Applications - Volume 02*, AINA '06, (Washington, DC, USA), pp. 560–564, IEEE Computer Society, 2006. xiv, xv, 75, 81, 82, 83, 84, 86, 104, 105, 106, 113, 123, 128, 135

[2] K. G. Coffman and A. M. Odlyzko, "Handbook of massive data sets," ch. Internet growth: is there a "Moore's law"for data traffic?, pp. 47–93, Norwell, MA, USA: Kluwer Academic Publishers, 2002. 1, 67

[3] L. Xiao, K.-S. Lui, J. Wang, and K. Nahrsted, "Qos extension to bgp," in *Proceedings of the 10th IEEE International Conference on Network Protocols*, ICNP '02, (Washington, DC, USA), pp. 100–109, IEEE Computer Society, 2002. 2, 74, 86, 92, 99, 104, 107, 109

[4] D. Comer, *Internetworking with TCP/IP: Principles, protocols, and architecture*. Internetworking with TCP/IP, Pearson Prentice Hall, 2006. 5, 15

[5] A. Tanenbaum, *Computer Networks*. Prentice Hall Professional Technical Reference, 4th ed., 2002. 6, 10, 40, 68

[6] R. Braden, "Requirements for Internet Hosts - Communication Layers." RFC 1122 (Standard), Oct. 1989. Updated by RFCs 1349, 4379, 5884, 6093, 6298, 6633. 6

[7] J. Postel, "Transmission Control Protocol." RFC 793 (Standard), Sept. 1981. Updated by RFCs 1122, 3168, 6093, 6528. 7

[8] J. Postel, "User Datagram Protocol." RFC 768 (Standard), Aug. 1980. 7

[9] J. Postel, "Internet Protocol." RFC 791 (Standard), Sept. 1981. Updated by RFCs 1349, 2474. 7, 8

[10] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification." RFC 2460 (Draft Standard), Dec. 1998. Updated by RFCs 5095, 5722, 5871, 6437, 6564. 8

[11] G. Malkin, "RIP Version 2." RFC 2453 (Standard), Nov. 1998. Updated by RFC 4822. 10

[12] J. Moy, "OSPF Version 2." RFC 2328 (Standard), Apr. 1998. Updated by RFCs 5709, 6549. 10

[13] J. Hawkinson and T. Bates, "Guidelines for creation, selection, and registration of an Autonomous System (AS)." RFC 1930 (Best Current Practice), Mar. 1996. 11, 19

[14] Y. Rekhter, T. Li, and S. Hares, "A Border Gateway Protocol 4 (BGP-4)." RFC 4271 (Draft Standard), Jan. 2006. Updated by RFCs 6286, 6608. 14, 50, 139

[15] R. Chandra and J. Scudder, "Capabilities Advertisement with BGP-4." RFC 3392 (Draft Standard), Nov. 2002. Obsoleted by RFC 5492. 20

[16] P. Traina, D. McPherson, and J. Scudder, "Autonomous System Confederations for BGP." RFC 5065 (Draft Standard), Aug. 2007. 36

[17] J. W. Stewart, III, *BGP4: Inter-Domain Routing in the Internet*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1998. 40

[18] P. Ferguson and G. Huston, *Quality of service: delivering QoS on the Internet and in corporate networks*. New York, NY, USA: John Wiley & Sons, Inc., 1998. 65, 66

[19] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin, "Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification." RFC 2205 (Proposed Standard), Sept. 1997. Updated by RFCs 2750, 3936, 4495, 5946, 6437. 68

[20] J. Wroclawski, "The Use of RSVP with IETF Integrated Services." RFC 2210 (Proposed Standard), Sept. 1997. 68

[21] K. Nichols, S. Blake, F. Baker, and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers." RFC 2474 (Proposed Standard), Dec. 1998. Updated by RFCs 3168, 3260. 69

[22] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An Architecture for Differentiated Services." RFC 2475 (Informational), Dec. 1998. Updated by RFC 3260. 69

[23] B. Davie, A. Charny, J. Bennet, K. Benson, J. L. Boudec, W. Courtney, S. Davari, V. Firoiu, and D. Stiliadis, "An Expedited Forwarding PHB (Per-Hop Behavior)." RFC 3246 (Proposed Standard), Mar. 2002. 69

[24] J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski, "Assured Forwarding PHB Group." RFC 2597 (Proposed Standard), June 1999. Updated by RFC 3260. 70

[25] D. Grossman, "New Terminology and Clarifications for Diffserv." RFC 3260 (Informational), Apr. 2002. 70

[26] D. Ghosh, V. Sarangan, and R. Acharya, "Quality-of-service routing in ip networks," *Trans. Multi.*, vol. 3, pp. 200–208, June 2001. 70

[27] G. Apostolopoulos, R. Guérin, S. Kamat, and S. K. Tripathi, "Quality of service based routing: a performance perspective," *SIGCOMM Comput. Commun. Rev.*, vol. 28, pp. 17–28, Oct. 1998. 70

[28] T.-I. Kim, H.-W. Jung, M. Y. Chung, and S.-I. Jin, "Inter-domain routing based on link state information for end-to-end qos guarantee," in *Proceedings of the 11th international conference on Advanced Communication Technology - Volume 3*, ICACT'09, (Piscataway, NJ, USA), pp. 1729–1732, IEEE Press, 2009. 74

[29] R. A. Guérin and A. Orda, "Qos routing in networks with inaccurate information: theory and algorithms," *IEEE/ACM Trans. Netw.*, vol. 7, pp. 350–364, June 1999. 74

[30] S. Chen and K. Nahrstedt, "Distributed qos routing with imprecise state information," pp. 614–621, 1998. 74

[31] H. F. Salama, D. S. Reeves, and Y. Viniotis, "A distributed algorithm for delay-constrained unicast routing," in *IEEE INFOCOM'97*, pp. 239–250, 1997. 74

[32] L. Benmohamed, C.-J. M. Liang, E. Naber, and A. Terzis, "Qos enhancements to bgp in support of multiple classes of service," in *Proceedings of the 2006 IEEE conference on Military communications*, MILCOM'06, (Piscataway, NJ, USA), pp. 395–401, IEEE Press, 2006. 75, 80, 86, 104

[33] B. Benmohamed, L.; Doshi, "Qos routing in multi-domain networks," in *Communications, Computers and signal Processing, 2005.*, PACRIM. 2005, (Piscataway, NJ, USA), pp. 137–140, IEEE Press, 2005. 75, 80, 86, 104

[34] B. D. T. C. R. Benmohamed, L.; Doshi, "Inter-domain routing with multi-dimensional qos requirements," in *Military Communications Conference, 2005.*, MILCOM'05, (Piscataway, NJ, USA), pp. 265–271, IEEE Press, 2005. 75, 80, 86, 104, 105, 106, 113, 135

[35] E. Mingozzi, G. Stea, M. A. Callejo-Rodríguez, J. Enríquez-Gabeiras, G. García-de Blas, F. J. Ramón-Salquero, W. Burakowski, A. Beben, J. Sliwinski, H. Tarasiuk, O. Dugeon, M. Diaz, L. Baresse, and E. Monteiro, "Euqos: End-to-end quality of service over heterogeneous networks," *Comput. Commun.*, vol. 32, pp. 1355–1370, July 2009. 81

[36] C. Cristallo, G.; Jaquenet, "Providing Quality of Service Indication by the BGP-4 Protocol: the QOS_NLRI attribute." draft-jacquenet-qos-nlri-00.txt, June 2003. 82, 86, 113

[37] L. Xiao, J. Wang, K.-S. Lui, and K. Nahrstedt, "Advertising interdomain qos routing information," *IEEE J.Sel. A. Commun.*, vol. 22, pp. 1949–1964, Sept. 2006. 86, 92, 99, 104, 105, 106, 107, 109, 111, 112, 113, 122, 123, 124, 125, 131, 135

[38] W. Jiang and H. Schulzrinne, "Modeling of packet loss and delay and their effect on real-time multimedia service quality," in *PROCEEDINGS OF NOSSDAV '2000*, 2000. 109

[39] J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach*. USA: Addison-Wesley Publishing Company, 5th ed., 2009. 109

[40] A. Konrad, B. Y. Zhao, A. D. Joseph, and R. Ludwig, "A markov-based channel model algorithm for wireless networks," *Wirel. Netw.*, vol. 9, pp. 189–199, May 2003. 109

[41] J. pierre Ebert, A. Willig, D. ing Adam Wolisz, and T. Berlin, "A gilbert-elliot bit error model and the efficient use in packet level simulation," 1999. 109

[42] I. A. Almerhag and M. E. Woodward, "Security as a quality of service routing problem," in *Proceedings of the 2005 ACM conference on Emerging network experiment and technology*, CoNEXT '05, (New York, NY, USA), pp. 222–223, ACM, 2005. 109

[43] E. Lindskog, S.; Jonsson, "Adding security to quality of service architectures," 2002. 109

[44] J. J. L. A. Kenneth C. Mansfield, *Computer Networking for LANs to WANs: Hardware, Software and Security*. Cengage Learning, 1st ed., 2010. 118

[45] M. Jain and C. Dovrolis, "End-to-end available bandwidth: measurement methodology, dynamics, and relation with tcp throughput," *IEEE/ACM Trans. Netw.*, vol. 11, pp. 537–549, Aug. 2003. 123

[46] C.-K. Tham, Y. Jiang, and C.-C. Ko, "Monitoring qos distribution in multimedia networks," *Int. J. Netw. Manag.*, vol. 10, pp. 75–90, Mar. 2000. 123

[47] Y. Gordon, A. Litvak, C. SchÃ$\frac{1}{4}$tt, and E. Werner, "Minima of sequences of gaussian random variables," *Comptes Rendus Mathematique*, vol. 340, no. 6, pp. 445 – 448, 2005. 132

[48] J. E. Hill, "The minimum of n independent normal distributions," 2011. 132