Encaminamiento Inter-Dominio con Calidad de Servicio basada en una Arquitectura Overlay y QBGP

Marcelo Yannuzzi[1], Alexandre Fonte[2,3], Xavier Masip-Bruin[1], Edmundo Monteiro[2], Sergi Sànchez-López[1], Marilia Curado[2], Jordi Domingo-Pascual[1], Josep Solé-Pareta[1]

[1] Departament d'Arquitectura de Computadors, Universitat Politècnica de Catalunya (UPC)
Avgda. Víctor Balaguer, s/n – 08800 Vilanova i la Geltrú, Barcelona, Catalunya, España
Teléfono: +34 934011055, Fax: +34 934017055
{yannuzzi, xmasip, sergio, jordid, pareta}@ac.upc.es

[2] Laboratorio de Comunicaciones y Telemática, CISUC-DEI, Universidad de Coimbra,
Pólo II, Pinhal de Marrocos, Dirección Postal 3030-290 Coimbra, Portugal
{afonte, edmundo, marilia}@dei.uc.pt

[3] Instituto Politécnico de Castelo Branco,
Av. Pedro Álvares Cabral, nº12, Dirección Postal 6000-084, Castelo Branco, Portugal

Resumen

Este documento propone un nuevo enfoque al tema de Encaminamiento Inter-Domino con Calidad de Servicio (QoS). Nuestro enfoque consiste en proporcionar una Arquitectura Overlay completamente distribuida así como una nueva capa de encaminamiento para el aprovisionamiento dinámico de QoS, pero haciendo uso de las extensiones de QoS y las capacidades de Ingeniería de Tráfico de la subyacente capa BGP para el aprovisionamiento estático de QoS. Hasta donde conocemos, nadie ha intentado combinar lo mejor de ambos mundos de forma de aportar una solución complementaria al tema de encaminamiento Inter-Domino con QoS. Nuestro objetivo radica principalmente en influenciar como se intercambia el tráfico entre Sistemas Autónomos mutihomed remotos, basados en parámetros específicos y preestablecidos de QoS. Proporcionamos un conjunto de resultados obtenidos por medio de simulaciones los cuales avalan la factibilidad de nuestra propuesta.

1. Introducción

Actualmente, casi el 80% de los más de 15000 Sistemas Autónomos (ASs) que componen Internet son stub [1], donde la mayoría de esta fracción es multi-homed. Para estos ASs, el tema de Encaminamiento con Calidad de Servicio (QoSR) a nivel de Inter-Dominio aparece como una necesidad imperiosa, sobre todo para poder soportar los nuevos servicios y aplicaciones emergentes sobre Internet [2]. Mientras algunos grupos de investigación proponen extensiones de QoS e Ingeniería de Tráfico (TE) sobre BGP [3, 4, 5], otros tienden a evitar nuevas extensiones que puedan sobrecargar el protocolo y proponen abordar el tema usando redes Overlay [6, 7, 8]. Mientras el primer enfoque mencionado proporciona mejoras significativas en redes que presentan escasa dinámica en el encaminamiento, el segundo, resulta más efectivo cuando los cambios en el encaminamiento se producen con una mayor frecuencia. La idea primordial detrás de una propuesta Overlay es desacoplar parte de las políticas de control del propio proceso de encaminamiento en los dispositivos BGP.

En este sentido, ambos enfoques se diferencian en la forma en que se señalizan y controlan las políticas. Las mejoras y extensiones a BGP tienden a proveer señalización en banda, mientras que el enfoque overlay proporciona señalización fuera de banda.

La Arquitectura Overlay resulta particularmente apropiada cuando los dominios involucrados son multi-homed, los cuales necesitan por ende algún mecanismo que le permita rápidamente cambiar el comportamiento de su tráfico dependiendo de las condiciones en la red. De hecho, esta es la tendencia que actualmente exhiben la mayoría de los ASs stub en Internet, ya que a través de la opción del multihoming buscan una conexión a la red que les permita balancear carga además de disponer de un acceso tolerante a fallos [7]. Además, las características actuales del tráfico inter-domino en Internet revelan que a pesar de que un AS intercambiará tráfico con la mayoría de la Internet, sólo un pequeño número de AS resulta responsable de una gran fracción del tráfico existente. Más aún, este tráfico se intercambia mayoritariamente con ASs que no se encuentran directamente conectados, sino que por el contrario se encuentran en general a 2, 3 y 4 hops de distancia [5]. La combinación de todas estas características es lo que nos ha hecho focalizar en QoSR entre ASs multi-homed remotos estratégicamente seleccionados.

Ciertamente la propuesta no se encuentra limitada a este caso y puede ser aplicada aún en el caso en que los ASs compartan enlaces BGP. La propuesta de QoSR que formulamos en este documento consiste en la provisión de una Arquitectura Overlay completamente distribuida, así como una capa de encaminamiento para el aprovisionamiento dinámico de QoS, pero usando las extensiones de QoS y las capacidades de TE de la subyacente capa BGP para el aprovisionamiento estático de QoS. En términos de la estructura de encaminamiento subyacente BGP, pueden operar dos tipos de enrutadores, aquellos que poseen facultades QoS, a los cuales denominaremos enrutadores QBGP y aquellos que no las poseen a los que llamaremos simplemente enrutadores BGP.

Por lo que con el objetivo de desarrollar esquemas de enrutamiento estables y altamente escalables, resulta prioritario que los enrutadores QBGP distribuyan únicamente información no dinámica. Esto se debe a que cambios frecuentes en la red se traducirán en mensajes frecuentes de update en BGP, los cuales pueden generar serias inestabilidades en el encaminamiento de la red. Dentro de la estructura de encaminamiento overlay residen unas entidades especiales denominadas Entidades Overlay (OEs), cuyas principales funcionalidades son las de intercambio de Acuerdos de Nivel de Servicio (SLAs), monitorización extremo a extremo, y análisis del cumplimiento con los SLAs preestablecidos. Estas funcionalidades permiten que la OEs influencien el comportamiento de la subyacente capa de encaminamiento BGP, de forma de poder tomar decisiones rápidas y precisas, y así evitar problemas en la red tales como fallos de enlaces o la degradación del servicio para una Clase de Servicio (CoS) dada. La naturaleza reactiva de esta estructura overlay actúa como una capa complementaria concebida para mejorar el rendimiento de la subyacente capa BGP compuesta por enrutadores BGP y QBGP. Nuestro interés en términos de investigación y por lo tanto el alcance de este documento es en el marco más general, con foco en la capa overlay y su acople con la subyacente capa BGP. Lo que sigue de este documento se encuentra organizado de la siguiente manera. La Sección 2 presenta nuestra propuesta overlay en términos generales. En la Sección 3 se analizan las principales funcionalidades requeridas de las capas overlay y BGP, mientras que la Sección 4 presenta el escenario de simulación así como los resultados de las mismas. Por último, la Sección 5 concluye el documento.

2. Análisis general de la propuesta Overlay

Como indicamos en la Sección anterior, en este documento proponemos una Arquitectura Overlay combinada con QBGP para poder brindar QoSR inter-dominio. Las ideas centrales detrás de la Arquitectura Overlay son:

- Las OEs deben responder aproximadamente dos órdenes de magnitud más rápido que la capa BGP en caso de fallos en la red.
- Las OEs deben reaccionar e intentar reacomodar el tráfico cuando se detecten situaciones de violación de alguno de los parámetros de QoS enmarcados dentro de los acuerdos previamente pactados en los SLAs para cualquier CoS.
- La estructura subyacente BGP no necesita modificación alguna en esta etapa de nuestra investigación, y permanece por lo tanto sin conocimiento de la existencia de la capa superior overlay.

La siguiente figura esquematiza un escenario posible en el cual nuestra propuesta podría ser aplicada. En este modelo, dos OEs pertenecientes a ASs diferentes separados por varios hops de ASs son capaces de intercambiar SLAs y pactar acerca de ciertos

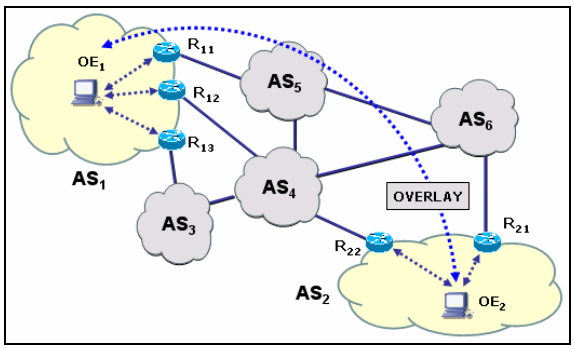

Fig. 1. Escenario de Encaminamiento Inter-Dominio con QoS donde las OEs son utilizadas para el aprovisionamiento dinámico de QoS entre ASs multi-homed remotos.

parámetros de QoS concerniente al tráfico entre ambos ASs. Los ASs intermedios no necesitan participar en absoluto de la Arquitectura Overlay ni del acuerdo y por lo tanto no resultan necesarias OEs en los ASs de tránsito. Desde nuestra perspectiva el verdadero desafío es poder desarrollar un sistema overlay completamente distribuido, donde cada OE se comporte de modo reflectivo. En este sentido nuestra propuesta es como mirar en un espejo. En lugar de proponer esquemas complejos para intentar manejar en forma precisa y dinámica como el tráfico debería de entrar al AS de destino, nos centramos en el problema de cómo debería de haber salido el tráfico desde el AS de origen. Por ende lo que buscamos es que la OE dentro del AS de origen se comporte como la imagen en un espejo de la OE dentro del AS de destino. Este esquema espejado permite que la OE dentro del AS de origen maneje dinámicamente el tráfico saliente del AS y dirigido al AS de destino, dependiendo del cumplimiento o no con los SLAs preestablecidos para las distintas CoSs que operen entre sí. Luego, bajo condiciones normales de red, y basadas en el aprovisionamiento estático de QoS, cada OE medirá el estado extremo a extremo de ciertos parámetros de QoS y será capaz de detectar violaciones en los acuerdos para cada CoS. Dentro de cada AS, la OE provee herramientas para realizar estas medidas a lo largo de cada enlace que conecte el AS multi-homed con Internet. De aquí en adelante asumiremos que la topología es tal que existen al menos dos paths (caminos) distintos extremo a extremo que permiten conectar a un par de ASs cualesquiera participando de nuestro modelo de QoSR inter-dominio. Cuando una violación es detectada, la OE dentro del AS de origen es capaz de reconfigurar rápidamente su patrón de tráfico hacia el AS de destino para la CoS afectada. En este caso la escala de tiempo necesaria para detectar y reaccionar a un cierto problema es muy pequeña cuando comparamos con las escalas de tiempo que necesita BGP [9].

Las medidas extremo a extremo están basadas en el sondeo activo (active probing) entre pares de OEs, a través de los distintos paths que conectan los ASs de interés. Por lo tanto, cada OE generará paquetes de prueba, a los que denominaremos probes, destinados al AS remoto a través de cada enlace disponi-

ble que conecte el AS de origen con Internet. Sostenemos que esta práctica de AS-AS probing no resulta demandante ni en términos de tráfico, ni en términos de procesamiento, siempre que el número de ASs remotos en el esquema overlay así como el número de CoS permanezcan limitadas. De hecho, el tráfico de probes generado entre las OEs es despreciablemente pequeño y una provisión acertada de recursos a través de QBGP implica que las OEs prácticamente no realizarán trabajo alguno. Vale la pena destacar que una violación en un SLA podría ocurrir únicamente en una dirección del tráfico entre los ASs, lo cual significa que el cuello de botella se encuentra simplemente en el flujo de datos de subida ó en el de bajada. Por ejemplo, en la Fig. 1 las OEs en AS_1 y AS_2 miden los mismos parámetros, como ser el Retardo en un Sentido (One-Way Delay = OWD) [10] o las Pérdidas en un Sentido (One-Way Loss = OWL) [11], y son capaces de reaccionar de la misma manera debido a su comportamiento reflectivo (espejo). De esta forma, cualquiera de las dos OEs es capaz de reaccionar independientemente y decidir si debe reacomodar o no el tráfico saliente desde su AS. Esta es la premisa fundamental de la Arquitectura Overlay distribuida, es decir lograr el objetivo de que cada OE actúe independientemente de acuerdo a su percepción del estado de la red.

A modo de ejemplo y como para mostrar la flexibilidad de nuestra propuesta, asumamos que C_1 representa una CoS anunciada por los enrutadores R_{11}, R_{12} y R_{13} en el AS_1 y que por razones de políticas locales AS_1 prefiere recibir tráfico que cumpla con el SLA de C_1 desde el AS_2 a través de R_{11}. Sin embargo, en caso que el path seleccionado no sea capaz de proveer las exigencias previamente negociadas en el SLA entre ambos ASs, AS_1 permite la recepción de este tráfico a través de R_{12} ó R_{13} siempre que el acuerdo pueda ser respetado. En este caso, en lugar de utilizar QBGP, el objetivo principal podría satisfacerse si el aprovisionamiento estático se realiza por medio de TE-BGP, y la capa overlay provee del QoSR dinámico. Por ejemplo, basados en los atributos de comunidades de BGP [5], AS_1 podría requerir a AS_4 que éste realice un prepend de su propio AS tres veces antes de anunciar C_1 a AS_2, que realice un prepend de su AS dos veces antes de anunciar C_1 a AS_6, y que no realice ninguna operación de prepend al anunciar esta clase a cualquier otro AS vecino. Por ende, los anuncios que AS_2 recibe bajo este escenario son: {AS_4, AS_4, AS_4, AS_1}; {AS_6, AS_5, AS_1}. Esto determina que AS_2 elija enviar tráfico para C_1 a través de AS_6. No obstante, una vez que se ha seleccionado el mejor path, TE-BGP desconoce completamente cualquier restricción de QoS pactada entre AS_1 y AS_2. Asumamos ahora que el enlace entre AS_2 y AS_6 pasa a estar cargado, mientras que el path {AS_4, AS_1} a través de R_{22} no lo está. A pesar de estas condiciones desiguales, BGP seguirá prefiriendo el path a través de R_{21}. Nuestra propuesta permite que la OE en AS_2 se percate de estas condiciones y dinámicamente reencamine el tráfico saliente de AS_2 para C_1 a través de R_{22}. Una ventaja sustancial es que este enfoque permite evitar completamente los updates de BGP ya que, como lo único que debe reacomodarse es tráfico saliente en AS_2, resulta posible emplear por ejemplo la LOCAL PREFERENCE de BGP.

Un punto importante de nuestra propuesta es que intentamos reducir al máximo posible la complejidad adicional que introduce la capa overlay. Agarwal y otros proponen un mecanismo overlay interesante para reducir los tiempos de fallos en la red y para conseguir balanceo de carga en el tráfico entrante a un AS [7]. Sin embargo, la propuesta no reutiliza ningún esquema de QoS o de TE de la capa BGP. Más aún, introduce un complejo servidor centralizado que permite que un AS en la estructura overlay infiera, por medio de heurísticas, la topología así como las relaciones cliente/vecino entre los múltiples ASs que conforman todos los paths tentativos entre los AS afectados en caso de un fallo. La propuesta requiere que todos los ASs intermedios participen de la estructura overlay. La complejidad introducida se debe principalmente al hecho de que controlar en forma precisa como el tráfico ingresa a un AS es una tarea intrincada, particularmente cuando esto debe hacerse bajo condiciones dinámicas. Como alternativa, nuestra propuesta trata acerca de cómo acomodar el tráfico en el AS de origen, ya que estamos absolutamente convencidos de que propuestas efectivas pero más sencillas como esta resultarán más atractivas para ser desplegadas y adoptadas en redes reales.

3. Principales Funcionalidades de las distintas capas de Enrutamiento

En esta Sección describimos en detalle las funcionalidades de las dos capas de enrutamiento en nuestra propuesta. A saber la capa superior, la cual trata las funcionalidades de encaminamiento de la capa de la overlay, así como las funcionalidades de encaminamiento relativas a la capa inferior BGP.

3.1 Capa Superior: Funcionalidades del Encaminamiento Overlay

Esta capa está compuesta por un conjunto de OEs:

3.1.1 Conjunto básico de componentes:

- Al menos una OE existe por dominio QoS.
- Una OE contará con total acceso a los enrutadores de borde dentro de un AS.
- En el escenario más simple, una OE podrá actuar en forma completamente independiente y sin acuerdo alguno con OE remotas. En este caso, la OE analizará únicamente el cumplimiento o no de políticas locales de QoS y en caso de detectar violaciones, sintonizará BGP dinámicamente para lograr una mejor distribución de tráfico de forma de poder cumplir con las exigencias locales de QoS.
- Una OE cuenta tanto con algoritmos para detectar condiciones de incumplimiento con el SLA para una CoS dada, así como para decidir cuando y de

que forma reacomodar el tráfico de manera de cumplir con dicho SLA.

3.1.2 Componentes Principales:

Un Protocolo Overlay: Se requiere un protocolo de comunicación entre OEs remotas. Este protocolo permite que las OEs intercambien SLAs entre sí, así como información sustancial para la Arquitectura Overlay. Una OE en el AS destino podría reacomodar parte del tráfico entrante al AS, solicitando remotamente a la OE del AS origen que modifique el tráfico saliente desde ese AS. Este comportamiento proactivo resulta necesario cuando por ejemplo, la OE dentro del AS destino recibe información de que ciertas tareas de mantenimiento deberán ser llevadas a cabo sobre uno de sus enlaces con el exterior, lo cual afectará una porción del tráfico intercambiado entre ambos ASs. En este caso, en lugar de esperar a que la OE en el AS de origen detecte y reaccione frente al problema, se provee dentro del protocolo de un mecanismo de aviso de forma tal de poder evadir completamente el fallo por adelantado.

Selección de una Métrica: Para poder validar nuestra propuesta elegimos un parámetro simple de QoS para la porción dinámica de nuestro modelo de QoSR. El parámetro que hemos seleccionado es un filtrado o alisado del OWD, al que denominaremos SOWD, el cual define la siguiente métrica:

$$\overline{OWD}(m,n) = \frac{1}{N} \sum_{k=n}^{k=n+N-1} OWD(m,k) \qquad (1)$$

Este SOWD esencialmente corresponde al promedio de un conjunto de muestras de OWD a través de una ventana deslizante de tamaño N. En lugar de utilizar valores instantáneos de OWD, proponemos usar este filtro pasabajos que alisa el OWD y evita que se produzcan cambios frecuentes en nuestra métrica. Resulta claro que existe un compromiso en términos del tamaño de la ventana N. Un valor grande de N implica que la reacción será lenta cuando las condiciones de la red cambien y tal vez un reacomodo del tráfico resulte necesario. Por otra parte, valores pequeños de N pueden generar reacomodos frecuentes de tráfico ya que las situaciones de incumplimiento con los SLAs se producirán probablemente con más frecuencia. En este escenario, el SLA intercambiado entre OEs será simplemente el máximo SOWD D_j tolerado para cada CoS diferente C_j.

Asumimos que cada OE usa una dirección lógica diferente para cada CoS, y que además se han aplicado políticas locales específicas en IBGP (Internal BGP). Bajo estas hipótesis, una OE puede enviar sus probes hacia una OE remota para cualquier CoS, en un esquema round-robin, a través de todos los enlaces externos disponibles en el AS local. De este modo, m y n corresponden al $n_{ésimo}$ probe generado por la OE del AS origen y enviado a lo largo del $m_{ésimo}$ enlace externo del AS. Entonces, las OEs computan para cada clase de servicio el costo de alcanzar el AS remoto a través de cada uno de sus enlaces externos m basados en la métrica anterior. Además, los paquetes de probe para una CoS dada, pertenecen a esa CoS. Por ejemplo, en un marco QBGP basado en Servicios Diferenciados (Diff-Serv), cuando se prueba una CoS en particular, la cual se encuentra localmente asignada a una clase Assured Forwarding (AF) en cada AS intermedio, los probes serán marcados bajo la misma clase AF [12]. Asumimos que las OEs se encuentran adecuadamente sincronizadas (por ejemplo mediante GPS) y los detalles concernientes al proceso de sincronización están fuera del alcance del presente trabajo.

Un mecanismo de Piggy-Backing: Un punto importante es que una técnica de probing activa desarrollada para medir el OWD requiere de realimentación de la OE remota. Sin embargo, el esquema espejado implica que la OE remota se encuentra generando probes hacia la OE local y espera a su vez el mismo tipo de realimentación. Entonces, para evitar cargar innecesariamente la red con estos mensajes proponemos dotar al protocolo de comunicaciones entre las OEs de un mecanismo de piggy-backing. De esta forma, la realimentación del OWD es acarreado dentro de los propios probes.

Estabilidad: Otro punto central en nuestra propuesta es que el proceso de reubicación de tráfico nunca debe introducir inestabilidad en la red. Podría ocurrir que un grupo de OEs comparta un conjunto de paths, donde cada una opera desconociendo acerca de la existencia de las demás. Éstas OEs podrían manejar CoSs con requerimientos muy dispares sobre esos paths, y bajo este esquema no existe ninguna entidad centralizada capaz de resolver la contienda por los paths. Estos conflictos podrían disparar varias acciones de reubicación de tráfico por parte de las distintas OEs, lo que podría derivar en oscilaciones del tráfico en la red. A modo de evitar que esto ocurra, pero manteniendo la premisa de que buscamos diseñar una arquitectura completamente distribuida donde cada OE debe contar únicamente consigo misma para manejar estos problemas, imponemos la siguiente restricción:

> "Tráfico destinado a cierta CoS C_j nunca debe ser reubicado sobre un enlace s, si y solo si el enlace primario para alcanzar C_j era s en $[t - T_h, t]$ o C_j ha excedido su máximo número de reubicaciones posibles $\Rightarrow R_j(t) \geq R_j^{MAX}$ "

De esta forma el parámetro T_h evita las oscilaciones de alta frecuencia, mientras que el parámetro R_j^{MAX} evita aquellas que se producen a más bajas frecuencias. De este modo, cada vez que un proceso de reubicación de tráfico tiene lugar para una CoS C_j, la variable $R_j(t)$ se ve incrementada. Nuestra propuesta consiste en proveer de una clase de penalización similar al damping en BGP [13], donde la penalidad se incrementa por un valor fijo P con cada nueva reubicación, pero decae exponencialmente con el tiempo cuando no existe ninguna reubicación de tráfico de acuerdo a:

$$R_j(t) = R_j(T)e^{-\left(\frac{t-T}{\tau}\right)} \quad (2)$$

Donde T_h, R_j^{MAX}, P y τ son parámetros configurables, cuyos valores dependen de los grados de libertad en el número de reubicaciones a corto y largo plazo que permitimos para una CoS dada C_j.

Un desafío adicional en términos de estabilidad surge cuando un path pasa a estar altamente ocupado, ya que varias CoSs dentro del path pueden experimentar violaciones con sus respectivos SLAs. Para prevenir la reubicación simultánea de todas las CoSs afectadas, dotamos a las OEs con un mecanismo de contención, el cual prioriza la relevancia de las distintas CoSs. De este modo, las CoSs de mayor prioridad serán reubicadas antes que aquellas de menor prioridad. El algoritmo de contención opera de la siguiente forma:

$$\begin{cases} \text{Sea } C_j \text{ una de las } q \text{ CoS afectadas dentro del enlace } m, \\ \text{donde } j = 1,..,q \\ C_j \text{ será reubicada en } T_j, \text{ donde } T_j \in [K_{j-1}, K_j) \\ y\ T_j \text{ se selecciona aleatoriamente} \\ \text{Definimos: } K_0 = 0 \end{cases}$$

$\Rightarrow$ *Entonces, las clases* C_1 *de mayor prioridad dentro del enlace m serán reubicadas en un tiempo aleatorio* $T_1 \in [0, K_1)$, *las clases* C_2 *serán reubicadas en un tiempo aleatorio* $T_2 \in [K_1, K_2)$, *y así sucesivamente.*

Claramente, nuestro mecanismo de contención permite que una OE reubique iterativamente tráfico del path cargado, mientras analiza dinámicamente si las clases que aún permanecen sin ser movidas continúan bajo condiciones de incumplimiento con sus respectivos SLAs. A medida que comencemos a extraer tráfico del path, es de esperar que las clases que aún permanecen ahí, comiencen a experimentar una mejora en sus parámetros de QoS extremo a extremo. Sin embargo, una situación muy distinta se produce cuando un enlace falla. En este caso, una OE debe reaccionar lo más rápido posible para reubicar todo el tráfico afectado. Entonces, existe un compromiso entre el mecanismo de contención y la habilidad de poder redistribuir rápidamente todo el tráfico en un enlace dado. En lugar de sintonizar eficientemente el algoritmo de contención de forma de abordar ambos problemas en simultáneo, nos apoyamos en el mecanismo de probing ya que el fallo de un enlace causará la pérdida de todos los probes para todas las CoS que utilizaban dicho enlace. Nuestra propuesta consiste en incrementar la frecuencia de los probes para cada CoS ni bien se comienzan a detectar las pérdidas. Sostenemos que este incremento en la frecuencia no exacerba la carga en la red, ya que en primer lugar la fracción de tráfico que genera la OE que detecta el problema es despreciable al comparar con el tráfico total que intercambian ambos ASs. En segundo lugar, esto se hace por un período muy breve de tiempo y con el único objetivo de acelerar el proceso de reubicación de tráfico. Una vez que una CoS es trasladada, la frecuencia de los probes retorna a su valor original.

3.2 Capa Inferior: Funcionalidades del Encaminamiento BGP

Las rutas que serán analizadas por las OEs usando la técnica de probing descrita en la sub-sección anterior, fueron predeterminadas por la subyacente capa BGP. En esta capa dos tipos de dispositivos pueden operar; enrutadores BGP tradicionales y enrutadores QBGP. Por un lado, un enrutador BGP tradicional distribuye Network Layer Reachability Information (NLRI) a sus vecinos de acuerdo a políticas de enrutamiento locales. Por otra parte, un enrutador QBGP puede distribuir información de QoS y por ende tomar decisiones de encaminamiento basado en esta información.

La escalabilidad es uno de los requerimientos más importantes de QoSR inter-dominio. Por lo tanto, desarrollar internets escalables, pero manteniendo el overhead en niveles aceptables, implica un compromiso entre la frecuencia de los anuncios y la imprecisión en la información de encaminamiento. Consecuentemente, en nuestro enfoque los dispositivos QBGP sólo deben manejar información de QoS no dinámica [14], y deben tomar decisiones de encaminamiento por CoS restringida al SLA preestablecido. En nuestro modelo los enrutadores QBGP pueden ser vistos como la herramienta práctica para establecer la infraestructura global de QoSR inter-dominio para cada CoS, la cual puede a su vez ser influenciada dinámicamente por la capa overlay. Propuestas interesantes de QBGP, así como información adicional en el área puede encontrase en [3, 4, 14].

3.3 Algoritmo Combinado de QoSR

El siguiente esquema (Fig.2) esquematiza nuestro algoritmo combinado de QoSR. Sea *m* el enlace externo que de momento aloja tráfico de clase C_j. Resulta importante remarcar que en nuestra propuesta el proceso de reubicación de tráfico solo puede ocurrir cuando existe un incumplimiento con un SLA. En este sentido, a pesar de que un path alternativo pueda mostrar mejor costo en términos de SOWD, evitamos reubicar el tráfico de clase C_j desde el enlace *m* hasta que una violación con el SLA sea detectada. Dos líneas separadas de eventos ocurren cuando se recibe un probe para una clase C_j. Inicialmente, el probe *(k,l)* es separado de la realimentación del *OWD(m,n)*, la cual llega piggy-backed en el propio mensaje. Lo primero que debe ser procesado es el *OWD(k,l)* de forma de responder en forma precisa a la OE que ha enviado el mensaje, lo cual se indica como (I) en la Fig. 2. Por otro lado, se procesa el *OWD(m,n)* piggy-backed, indicado como (II) en la figura. Una vez que se computa el SOWD, el algoritmo analiza si existe una violación con respecto al máximo SOWD tolerable D_j. Si no ha ocurrido una violación, el algoritmo simplemente queda a la espera de la llegada del próximo probe.

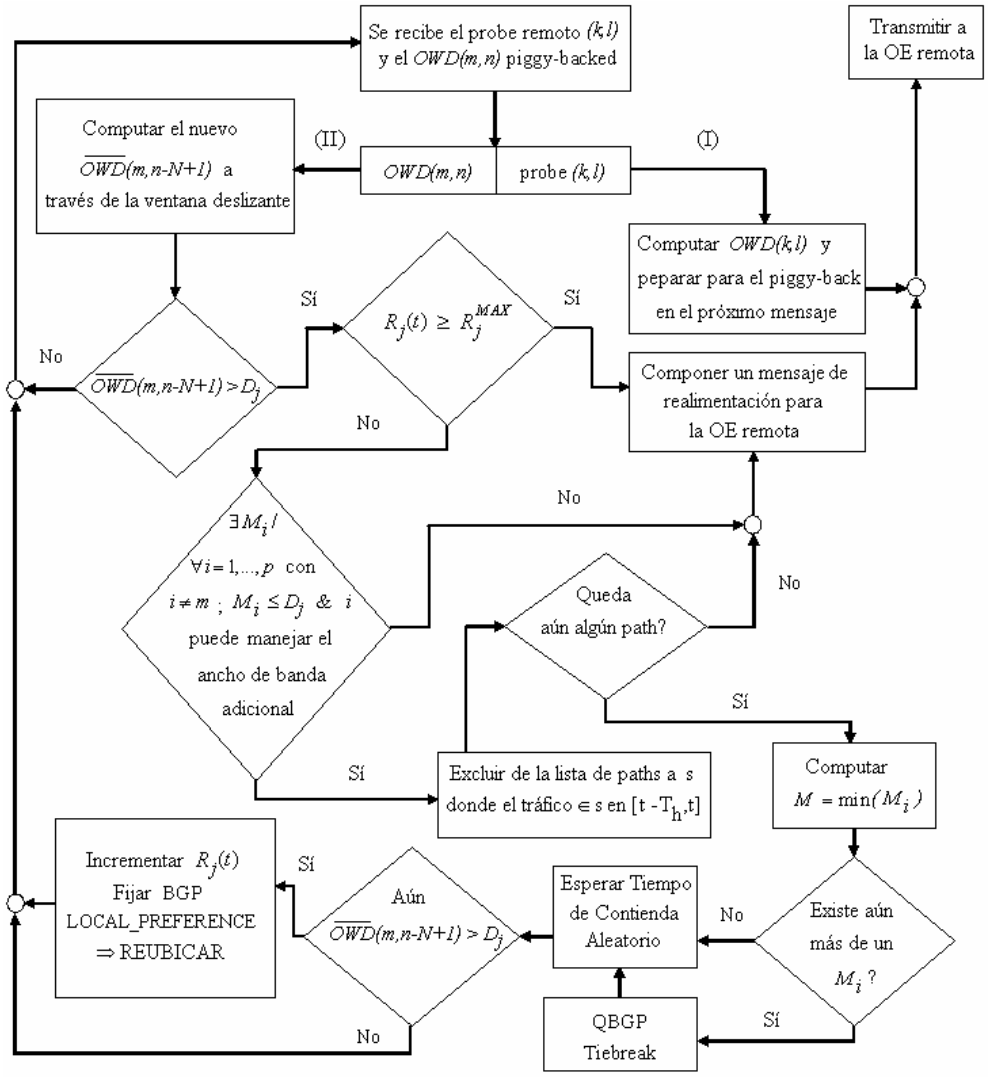

Fig. 2. Algoritmo Combinado de QoSR.

Sin embargo, si una violación es detectada en el enlace *m* el algoritmo examina si el máximo número de reubicaciones R_j^{MAX} ha sido alcanzado. Si este es el caso, la OE local compone un mensaje de realimentación para advertir a la OE remota acerca de esta situación. La idea central detrás de este procedimiento de aviso es que la realimentación provea de información a la OE remota, de forma tal que esta última pueda intentar manejar el problema reacomodando su aprovisionamiento estático de QoS usando ya sea QBGP o TE-BGP.

Si R_j^{MAX} no ha sido excedido, la OE debe examinar dentro de todos los enlaces externos *p* disponibles, exceptuando a *m*, si existe al menos uno cuyo costo M_i satisfaga las restricciones que impone la clase C_j. Además, se debe analizar si el enlace dispone de suficiente espacio como para aceptar la reubicación del tráfico. Luego, y con el objetivo de evitar cualquier oscilación a corto plazo, la OE excluye del grupo de enlaces capaces de aceptar la reubicación a aquellos que hayan ubicado tráfico para la clase C_j en [t - T_h, t]. Una vez realizado esto, nos apoyamos en el tiebreak (desempate) de QBGP si se da el caso de que dos o más enlaces muestren el mismo costo en términos de SOWD. En este punto ha quedado un único enlace como destino de la reubicación del tráfico de la clase C_j. Luego, el algoritmo de contención es ejecutado y T_j segundos después la OE examina si la clase aún continúa en una condición de incumplimiento. Si este es el caso, la OE incrementa $R_j(t)$ en *P* y reubica el tráfico de la clase C_j.

4. Resultados de las Simulaciones

La Arquitectura Overlay propuesta en este documento está siendo evaluada y validada mediante simulación. En esta sección presentamos algunos resultados preliminares a modo de realizar una primera evaluación de la arquitectura y de su capacidad para soportar distintas clases de tráfico con QoS en forma dinámica. Estamos utilizando el simulador J-Sim [15] con el paquete BGP Infonet [16] el cual cumple con las especificaciones de la RFC 1771 de BGP [17]. Se desarrollaron un conjunto de componentes Java con las funcionalidades de la capa overlay. A modo de permitir que las OEs tuvieran total acceso a las bases Adj-RIBs-In y Loc-RIB de un dispositivo BGP, y para tener control sobre el proceso de decisión de BGP, fue necesario agregar algunas extensiones al paquete Infonet. Una definición precisa de los términos Adj-RIBs-In y Loc-RIB

CoS	CBR (Mbps)	Tamaño de paquete (KB)	PHB	Máx. SOWD (ms)	Frecuencia de Probing	Hold (Contención) &T_h (s)	Mov. Average
CoS1	0,4	1	EF	85	1 s, 1KB	3 & 8	3 s
CoS2	0,8	1	AF11	100	1 s, 1KB	6 & 12	3 s
CoS3	1,0	1	AF21	120	1 s, 1KB	9 & 20	3 s
CoS4	1,6	N.A	BE	N.A	N.A	N.A	N.A

Tabla 1. Condiciones de las Simulaciones

puede ser encontrada en [17]. Además, otras extensiones fueron necesarias para permitir que la Loc-RIB fuera actualizada con los cambios en la Adj-RIB-In. Finalmente, y debido a que no existe ningún protocolo QBGP estandarizado, hemos incluido dentro del paquete Infonet las siguientes extensiones de QoS en BGP:

- Un atributo opcional y transitivo para distribuir las identificaciones (IDs) de las distintas CoSs, así como un conjunto de modificaciones a las tablas de BGP para permitir el almacenamiento de esta información adicional, siguiendo una línea similar a la descrita [3].
- Un conjunto de mecanismos para: i) permitir que los dispositivos BGP puedan cargar las CoS localmente soportadas; ii) permitir que cada prefijo IP local sea anunciado dentro de una CoS dada; iii) permitir que dispositivos BGP puedan fijar la permisión de rutas mediante el uso de filtros para permitir/denegar una ruta dada basado en políticas locales y las capacidades soportadas de QoS.

Para nuestras simulaciones, hemos utilizado la topología presentada en la Fig. 3. La topología está basada en el Backbone de la red Académica Europea GÈANT [18] con algunas simplificaciones para reducir la complejidad del modelo a simular.

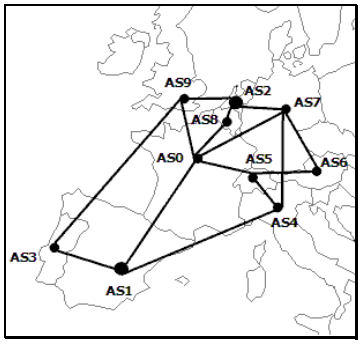

Fig. 3. Topología basada en la red GÉANT

En esta topología hemos considerado como ASs remotos multi-homed, a AS_1 y AS_2. Asumimos todos los enlaces como bidireccionales, con la misma capacidad C (C = 2Mbps) y con retardo de propagación P_d (P_d = 10ms), a excepción de los enlaces de AS_2, donde con el objetivo de contar con algunos cuellos de botella, la capacidad elegida fue C/2. De acuerdo con esto y solo por razones de complejidad, optamos por modelar cada AS como un único enrutador QBGP con capacidades DiffServ, configurado para soportar cuatro clases distintas de tratamiento de paquetes IP (EF, AF11, AF21 y Best-Effort) permitiendo entonces cuatro clases diferentes de tráfico, denominadas CoS1, CoS2, CoS3 y CoS4 respectivamente. Para completar el escenario, en el dominio extremo en el que inyectamos el tráfico hemos utilizado la capacidad de DiffServ de marcar paquetes con un DSCP (DiffServ Code Point) específico dependiendo de su correspondiente CoS. Este marcado fue aplicado tanto a los paquetes IP del tráfico de las fuentes así como a los probes generados por las OE. Las condiciones de las simulaciones están resumidas en la Tabla 1. Los resultados obtenidos son presentados en las Fig. 4 a Fig. 7. El máximo SOWD tolerado por CoS (D_j) fue heurísticamente elegido de forma que las OEs pudieran aprovechar los paths alternativos. El SWOD computado cuando se producían pérdidas de probes fue también elegido heurísticamente. El criterio seleccionado fue que 3 pérdidas consecutivas implicaran un aumento de un 25% aproximadamente en el SWOD. Para los experimentos presentados en esta etapa hemos fijado $R_j^{MAX} = \infty \ \forall j$. Además, no fueron generados probes para el tráfico Best-Effort, el cual figura como NA (Not Available) en la tabla, y una ventana deslizante de 3 segundos fue utilizada en todos los casos, lo cual aparece como Mov.Average en la Tabla 1. El primer objetivo de la simulación consistió en validar la premisa inicial de que nuestra propuesta, basada en una capa de encaminamiento complementaria, mejora la reacción de la infraestructura global de encaminamiento. De este modo, hemos elegido como indicador de performance, comparar el tiempo de respuesta frente a un fallo en un enlace. La Fig. 4 esquematiza un conjunto de gráficas para tráfico de CoS1 mostrando el throughput medido en el destino, el SOWD experimentado por los probes a través de todos los paths disponibles, y los cambios de path determinados por los cambios en el next-hop para el AS de origen AS_1. De estas gráficas podemos observar que en un marco puramente QBGP (sin OEs operando en AS_1 y AS_2) se necesitan unos 80 segundos para recuperar una falla en un enlace, pero tan solo 5 segundos son necesarios cuando las OEs están funcionando. Este resultado valida nuestra presunción inicial. Cabe destacar que este último valor incluye no solamente el tiempo para la detección implícita del fallo, basada en la violación del máximo SOWD tolerado, sino que también incluye un rango de contención aleatorio de 3 segundos antes de proceder a reubicar el tráfico. En segundo término examinamos el comportamiento de nuestra infraestructura global QoSR inter-dominio. Utiliza-

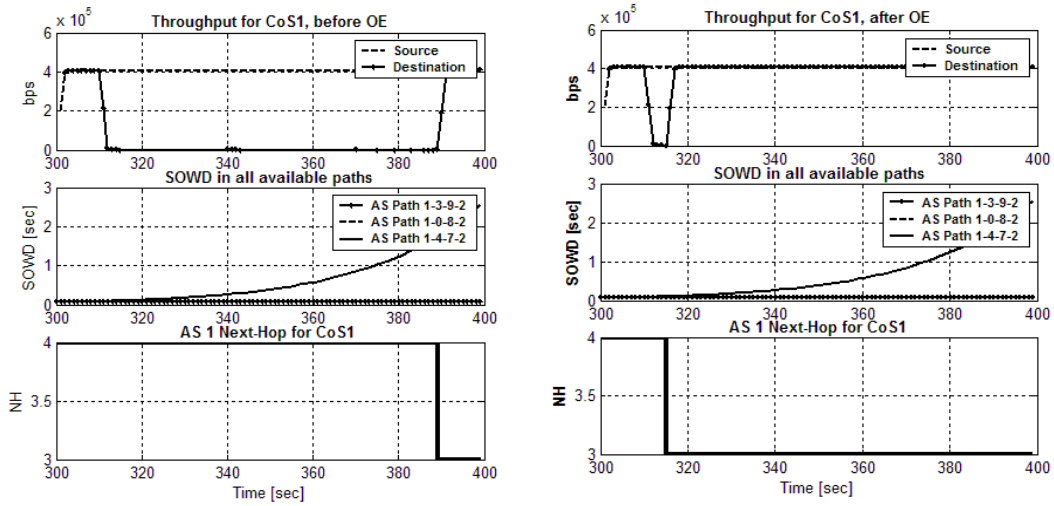

Fig. 4. Reacción frente al fallo de un enlace con y sin OE.

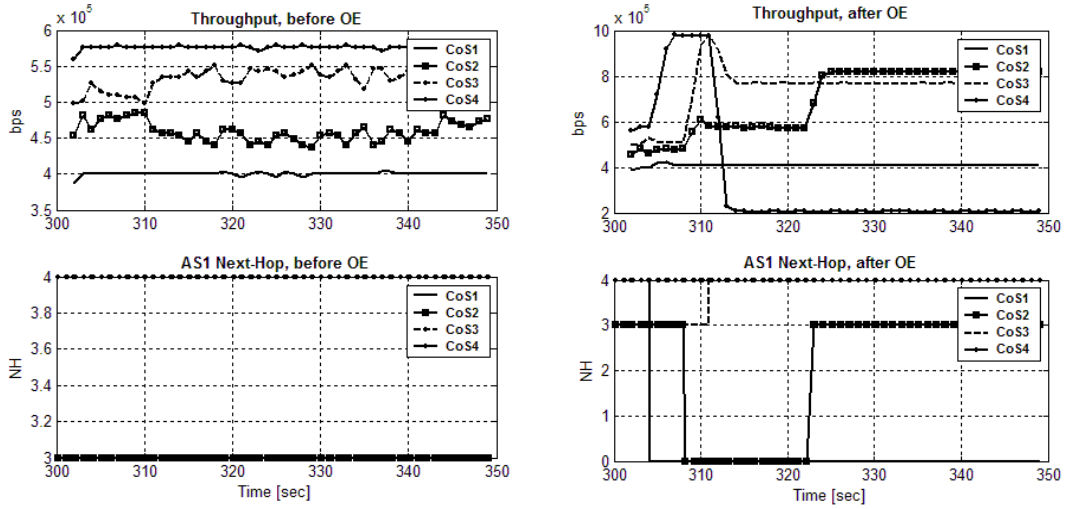

Fig. 5. Throughput para el tráfico de CoS1-CoS4, con y sin OE.

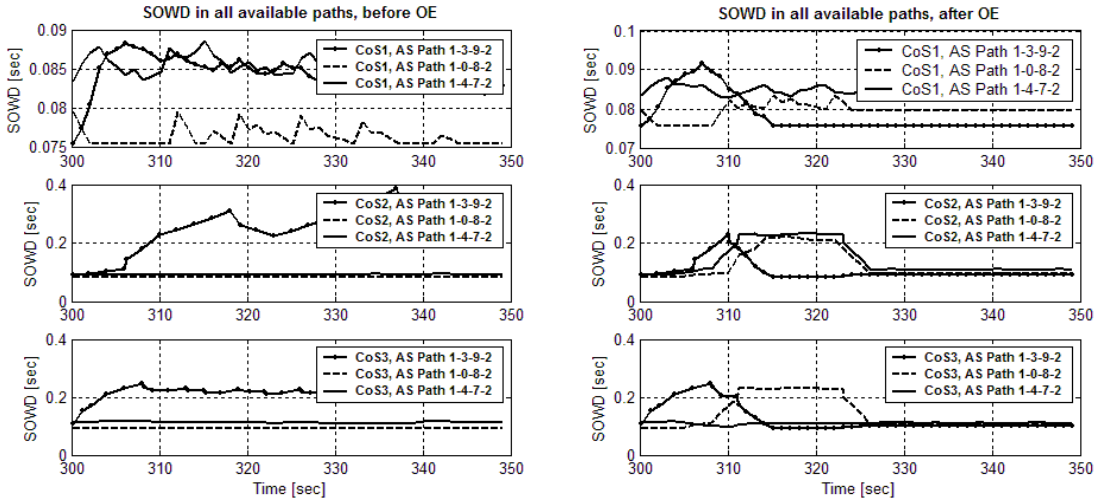

Fig. 6. OWD en todos los paths disponibles para CoS1-CoS4, con y sin OE.

mos como indicadores de performance para cada clase de tráfico tanto el throughput en el destino, así como el SOWD experimentado por los probes a lo largo de todos los paths disponibles. De las figuras 5 y 6, podemos observar que sin OEs existen claras violaciones a los SLAs establecidos entre ambos dominios. Sin embargo, con la presencia de las OEs, se ve claramente que la red es capaz de reaccionar a estas violaciones en los SLAs, y encontrar los mejores paths para reubicar el tráfico de las clases afectadas. Consecuentemente, luego de un transitorio de aproximadamente 13 segundos, necesarios para acomodar el tráfico para cada CoS, resulta visible que se alcanza un estado estable donde los SLAs son

respetados para todas las clases de tráfico.

Por último, y con el objetivo de evaluar la utilización general de enlaces, hemos medido el throughput en todos los enlaces disponibles en el AS de destino (AS_2). La Fig. 7 muestra que con las OEs, además de cumplir con los SLAs preestablecidos, se consigue una mejor distribución del tráfico inter-dominio, y por tanto, los recursos son utilizados más eficientemente. En contraste, sin las OEs, la distribución de tráfico depende de la exactitud y eficiencia del aprovisionamiento estático, el cual no tiene en cuenta la dinámica real de la red. El costo extra en todos estos casos fue apenas un incremento de 8 Kbps, por CoS, sobre cada enlace en el tráfico entre ambos ASs, cuando exigimos al sistema con probes sobredimensionados de 1 KB generados cada 1 segundo.

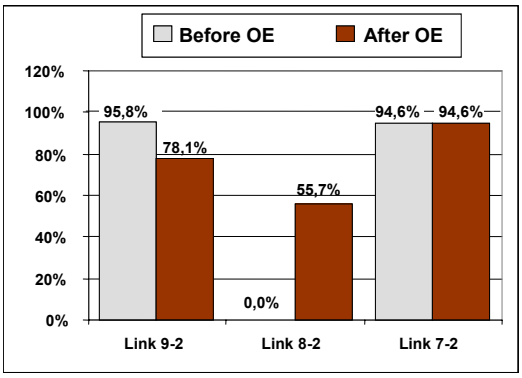

Fig. 7. ASs: utilización de enlaces remotos

5. Conclusiones

Este documento presenta el marco general de un paradigma combinado de encaminamiento inter-dominio con calidad de servicio, basado en una Arquitectura Overlay completamente distribuida y acoplada con una capa de encaminamiento con funcionalidades QBGP o TE-BGP. Como primer paso en nuestra investigación, y con el objetivo de validar nuestro enfoque nos hemos centrado en el acoplamiento de la capa overlay con una capa subyacente DiffServ QBGP.

Los resultados obtenidos muestran que nuestra Arquitectura Overlay distribuida mejora sustancialmente la calidad de servicio extremo a extremo cuando comparamos con un modelo QBGP puro. Creemos que si bien varias mejoras, así como significativas extensiones se seguirán planteando para BGP, la estructura overlay se posiciona como un candidato firme para poder proporcionar un modelo de encaminamiento con QoS fuera de banda, flexible y de valor agregado. Este modelo resulta particularmente aplicable cuando el tráfico inter-dominio debe adaptarse dinámicamente como para poder reaccionar rápidamente a condiciones de red cambiantes con frecuencias medias y altas, para las cuales las soluciones anteriores resultan impracticables en el presente.

Referencias

1. Olivier Bonaventure, Bruno Quotin, Steve Uhlig, “Beyond Interdomain Reachability”, Workshop on Internet Routing Evolution and Design (WIRED), October 2003.
2. E. Crawley, R. Nair, B. Rajagopalan, H. Sandick, “A Framework for QoS-based Routing in the Internet”, Internet Engineering Task Force, Request for Comments 2386, August 1998.
3. Cristallo, G., C. Jacquenet, “An Approach to Inter-domain Traffic Engineering”, Proceedings of XVIII World Telecommunications Congress (WTC2002), France, September 2002.
4. Li Xiao, King-Shan Lui, Jun Wang, Klara Nahrstedt, “QoS Extension to BGP”, IEEE ICNP, November 2002.
5. B. Quoitin, S. Uhlig, C. Pelsser, L. Swinnen and O. Bonaventure, “Interdomain Traffic Engineering with BGP”, IEEE Communications Magazine, May 2003.
6. L. Subramanian, Ion Stoica, Hari Balakrishnan, R. Katz, “OverQoS: Offering Internet QoS Using Overlays”, ACM SIGCOMM, Computer Communications Review, January 2003
7. S. Agarwal, C. Chuah, R. Katz “OPCA: Robust Inter-domain Policy Routing and Traffic Control”, IEEE Openarch, April 2003.
8. Zhi Li, Prasant Mohapatra, “QRON: QoS-aware Routing in Overlay Networks”, IEEE Journal on Selected Areas in Communications, June, 2003
9. C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, "Delayed Internet routing convergence," in Proc. ACM SIGCOMM, 2000.
10. G. Almes, S. Kalidindi, M. Zekauskas, “A One-way Delay Metric for IPPM”, Internet Engineering Task Force, Request for Comments 2679, September 1999.
11. G. Almes, S. Kalidindi, M. Zekauskas, “A One-way Packet Loss Metric for IPPM”, Internet Engineering Task Force, Request for Comments 2680, September 1999.
12. J. Heinanen, F. Baker, W. Weiss, J. Wroclawski, “Assured Forwarding PHB Group”, Internet Engineering Task Force, Request for Comments 2597, June 1999.
13. C. Villamizar, R. Chandra, R. Govindan, “BGP Route Flap Damping”, Internet Engineering Task Force, Request for Comments 2439, November1998.
14. IST MESCAL project, “Specification of Business Models and a Functional Architecture for Inter-domain QoS Delivery”, Deliverable D1.1, June 2003.
15. J-Sim Homepage, http://www.j-sim.org.
16. Infonet Suite Homepage, http://www.info.ucl.ac.be/~bqu/jsim/
17. Y. Rekhter, T. Li , “A Border Gateway Protocol 4 (BGP-4)”, Internet Engineering Task Force, Request for Comments 1771, March 1995.
18. GÉANT Website, http://www.dante.net/server/show/nav.007