Methodes de garantie de débit d'un ¤ot TCP dans un réseau à service assuré

Emmanuel Lochin[*], **Pascal Anelli**[**], **Serge Fdida**[*]

[*] *LIP6 – Université Pierre et Marie Curie*
4, place Jussieu, 75252 Paris Cedex 05 - France
{emmanuel.lochin, serge.fdida}@lip6.fr

[**] *IREMIA - Université de la Réunion*
15 Avenue René Cassin, 97715 Saint Denis Messag 9 - France
{pascal.anelli}@univ-reunion.fr

RÉSUMÉ. *Un réseau à service assuré est dé£ni selon l'architecture à différenciation de services (Diff-Serv). L'architecture Diff-Serv a été conçue comme une solution insensible au facteur d'échelle capable de fournir un service prévisible aux ¤ots sans aucune gestion par ¤ot à l'intérieur du réseau. Le service assuré dé£nit une classe de service pour laquelle le niveau de service est meilleur que le service unique dit "au mieux" de l'Internet. Le meilleur s'exprime pour TCP en termes de débit. Cet article présente une évaluation du service AS pour des ¤ots TCP. L'évaluation du service est faite sur la base d'un ¤ot applicatif et donc sur la perception réelle qu'en aura l'utilisateur. L'objectif consiste à étudier l'assurance de débit pour un ¤ot TCP indépendamment des conditions de tra£c dans la classe de service. L'évaluation est réalisée sur une plate-forme d'expérimentation, dans un environnement IPv6, développée au sein du projet RNRT @IRS.*

ABSTRACT. *Diff-Serv architecture was conceived as a solution insensitive to scalability issues and able to provide a predictable service without per ¤ow management. The assured service de£ned in Diffserv architecture caracterises a class of service better than the "best-effort" service. This results for TCP ¤ows on a guaranteed throughput. This article presents an evaluation of the AS service for TCP ¤ows, the evaluation of the service is made on the basis of application ¤ows and thus on real perception that the user will have. The objective of these measurements consists in studying throughput TCP ¤ow insurance independently of the traf£c presents in the class of service. The evaluation is made on an experimental testbed, in an IPv6 environment, which is developed within project RNRT @IRS.*

MOTS-CLÉS : *Qualité de service, Service assuré, Différenciation de Services, Agrégation de ¤ots, Performance de bout en bout, TCP.*

KEYWORDS: *QoS, Assured Service, Diff-Serv, Behavior aggregate, End to end performance, TCP.*

1. Introduction

EN quittant le monde académique et militaire pour se développer dans la société, les contraintes posées sur le service de transfert offert par l'Internet ont changé. En effet, l'Internet commercial s'accommode mal du service "au mieux" qui a prévalu depuis ses débuts. Les progrès des technologies numériques ont fait émerger des applications qui posent de nouvelles contraintes au service de communication. Les fournisseurs d'accès Internet souhaitent allouer ef£cacement les ressources a£n d'accroître leur activité au prix d'un investissement minimum. La Qualité de Service (QoS) est au cœur de ces nouvelles demandes. La QoS (sous l'angle quantitatif) se conjugue selon deux aspects : temporel et sémantique. Elle s'exprime principalement au travers des paramètres de :

– bande passante,

– latence,

– variation de la latence également appelée gigue,

– taux de perte.

L'architecture générale d'un réseau IP pour le support de la QoS repose sur le principe de l'agrégation de ¤ots en classes et de la gestion des ressources par classe plutôt que par ¤ot individuel [BLA 98]. L'architecture Diff-Serv distingue la frontière de l'intérieur d'un domaine d'administration. Un domaine se dé£nit comme une portion contigüe de l'Internet contrôlée par une même autorité administrative. La frontière d'un domaine d'administration est marquée par un routeur de bordure. Ce routeur joue un rôle différent de celui situé au cœur du domaine. Cette architecture s'inscrit dans le même paradigme que l'Internet qui est :"de reléguer la complexité dans les extrémités du réseau et de laisser le cœur du réseau aussi simple que possible". Cette architecture conduit à procéder à un simple ordonnancement des ¤ots au cœur du réseau et au contrôle du tra£c en bordure.

Le routeur de bordure a en charge la surveillance et le conditionnement du tra£c entrant. Ces tâches sont complexes et mettent en jeu une grande variété de contextes. Elles servent à limiter la quantité de tra£c injecté par chaque utilisateur dans le domaine. Elles sont essentielles et empêchent que les ¤ots du service soient perturbés par la congestion. Les contrôles sur le tra£c entrant s'appliquent au niveau du ¤ot utilisateur. Ce dernier peut être soit le tra£c issu d'un site, soit celui généré par une application. La granularité du ¤ot utilisateur et les paramètres du conditionnement de tra£c sont décrits dans un pro£l (TCA : *Traf£c Conditioning Agreement*). Le résultat du conditionnement se traduit concrètement par un marquage des paquets admis dans le domaine, par la suppression des paquets excédentaires, ou par la remise en forme du ¤ot (assurer un espacement temporel entre les paquets). Le traitement des paquets par les routeurs du domaine est dé£ni par le comportement de relayage (PHB : *Per-Hop Behavior*). La sélection du PHB est fonction de la marque contenue dans l'en-tête du paquet. En plus du comportement standard actuel dit DE (*Default*) utilisé par le service BE, deux PHB sont disponibles EF (*Expedited Forwarding*) [DAV 02] et AS (*Assured Forwarding*) [HEI 99b]. Le PHB EF permet de réaliser un service de trans-

fert à forte contrainte temporelle, tandis que le PHB AS assure à certains paquets une protection contre la perte en cas de congestion.

1.1. *Experimentations*

1.1.1. *Présentation du routeur de cœur*

Concernant les ¤ots AS, le pro£l d'un ¤ot est dé£ni par un débit moyen et par une sporadicité. Sachant que le débit moyen correspond à l'assurance soit au débit minimum. Le contrôle de conformité du ¤ot par rapport au pro£l se fait donc naturellement par un *token bucket* caractérisé par deux paramètres (r, b), à savoir le taux de fuite et la taille du seau. Les paquets sont marqués AS et une priorité spatiale leur est également attribuée en fonction de leur conformité au pro£l. Les paquets détectés conformes reçoivent une priorité à la perte faible (marqué *IN*) et les paquets non conformes ont une indication de forte priorité à la perte (marqué *OUT*). Ces derniers deviennent des paquets opportunistes. La priorité à la perte est utilisée quand le paquet passe par des routeurs congestionnés. En l'absence de congestion, cette priorité ne sert pas.

Concernant les ¤ots BE, aucun contrôle particulier ne doit être effectué : tous les paquets sont marqués DE. Le métreur n'a qu'un rôle d'information pour l'administration de réseau.

L'interface de sortie d'un routeur @IRS[1] comprend l'ensemble des mécanismes de relayage. Ces mécanismes ont en charge l'ordonnancement des paquets et la gestion des pertes en cas de congestion. La £gure 1 décrit la solution retenue et développée concernant le conditionnement du service AS. A l'entrée de l'interface, les paquets sont classés dans l'une des £les d'attentes en fonction de la marque qui leur a été attribuée par le conditionnement lors de leur entrée dans le domaine. Une telle classi£cation est dite BA (*Behavior Aggregate*), ou autrement dit selon la classe de service. La notion de ¤ot individuel n'existe plus.

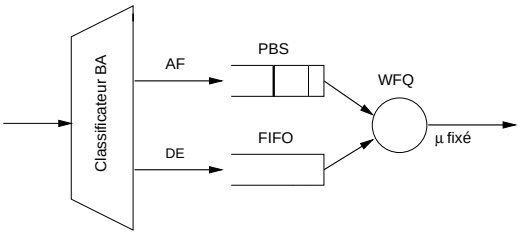


Figure 1. *Éléments fonctionnels d'une interface de sortie*

Le taux de service μ représenté sur la £gure 1 au niveau du WFQ (*Weighted Fair Queuing* [BEN 96]) sert à régler le débit d'émission. Le débit d'émission de l'interface

1. @IRS (Architecture Intégrée de Réseaux et Services - Décembre 1998 - Avril 2001) est un projet français £nancé par le Réseau National de la Recherche en Télécommunications (RNRT).

est ainsi con£gurable et indépendant du débit du support. Cette fonction trouve son utilité dans les expérimentations pour faire apparaître un goulot d'étranglement par rapport à la charge de tra£c offerte. Le taux de service μ est mis en œuvre au moyen d'un *leaky bucket*.

Une différenciation est faite entre paquets conformes et paquets non conformes. La discrimination de la perte selon une politique probabiliste tel que RED (*Random Early Discard*) est l'objet de débats. Des études ont montré que la politique de gestion de l'attente RED introduit de l'instabilité et des oscillations [BON 00a] [ZIE 99]. Notre solution repose sur une politique déterministe misee en œuvre par un mécanisme à simple seuil de type PBS (*Partial Buffer Sharing*). Les paquets opportunistes sont systématiquement rejetés dès que le nombre total de paquets en attente dans la £le excède un certain seuil.

2. Mesures et analyses des performances du système de communication

Les évaluations présenentées plus bas s'intéressent à la QoS reçue par un ¤ot élastique lorsqu'il utilise le service AS tel que celui développé dans @IRS. Ces évaluations sont faites dans le pire des cas. Le réseau est à chaque fois saturé par du tra£c BE (le réseau est donc en état de congestion). La mesure de l'impact dans le premier cas d'étude se fera par le délai de transit de bout en bout et le taux de perte. Plus précisément, les valeurs minimales, moyennes et maximales du délai de transit sont déterminées pour des sessions expérimentales d'environ 300 secondes chacune. On utilisera des sources de tra£c TCP et retiendra le débit utile comme paramètre de QoS.

2.1. *La plate-forme de tests*

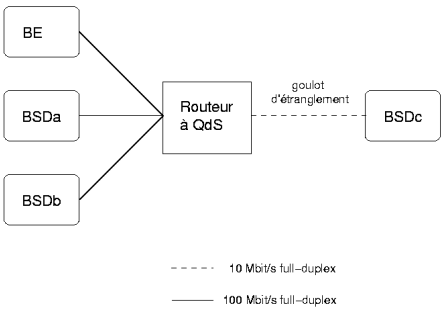


Figure 2. *Représentation fonctionnelle de la plate-forme de tests*

Comme le montre la £gure 2, la plate-forme de tests est composée de 5 nœuds. Le routeur sous FreeBSB comporte les mécanismes d'ordonnancement décrits dans la première partie. Les deux hôtes FreeBSD (BSD_A et BSD_B) émettent le tra£c à QoS

AS TCP. Le rôle du puit de tra£c qui a en charge certaines mesures est joué par l'hôte (BSD_C) sous système FreeBSD. Une machine est source de tra£c BE en émettant à un débit équivalent à la capacité du goulot d'étranglement.

Le débit de l'interface de sortie du routeur à 10Mbits/s est con£guré à l'aide d'un *leaky bucket*. Les ¤ux sont générés au moyen d'un générateur de tra£c : BENCH [ROC] que nous avons porté en IPv6 a£n d'effectuer nos mesures et modi£é pour qu'il puisse marquer le ¤*ow label* de chaque paquet. BENCH offre la possibilité de générer plusieurs connections simultanées depuis un même émetteur. Cette capacité a été utilisée par la machine BSD_B.

Le choix de cette plate-forme peut paraître simple au premier abord, mais elle caractérise en fait n'importe quel routeur localisé dans un domaine. Que ce soit un routeur de bordure ou un routeur de cœur, au niveau microscopique, le comportement de relayage de chacun est identique. Au niveau macroscopique, la QoS résultante dépendra du routeur qui accusera les conditions de tra£c les plus mauvaises en entrée par rapport à la capacité d'écoulement de son support en sortie. Pour le service AS, les autres routeurs suivant un tel routeur recevront un ¤ot déjà lissé. Construire une plate-forme de tests à un seul routeur repose sur l'hypothèse que le ¤ux subira les perturbations les plus signi£catives au niveau de ce routeur. Une telle con£guration peut être désignée par le terme de goulot d'étranglement. Formellement un lien est dé£ni comme goulot d'étranglement pour un ¤ot si le ¤ot possède le débit plus important par rapport aux autres ¤ots de ce lien et que le lien est saturé. L'objectif de cette plate-forme consiste à mesurer les écarts de QoS les plus signi£catifs. Par exemple, la perte de temps au niveau d'un paquet ou la perte d'un paquet ne se corrige pas sur la suite de la route. Les détériorations d'un ¤ot de paquets ne peuvent qu'empirer ou au mieux rester identiques. Donc une plate-forme de tests composée d'un seul routeur pour former un goulot d'étranglement est suf£sante pour effectuer l'évaluation du service proposé.

2.2. *Scénarios d'études*

2.2.1. *Étude du service AS pour les ¤ots élastiques*

Le service AS vise à donner une garantie de QoS à un ¤ot en termes de débit et cela quelque soit la composition de l'agrégat. L'objectif de cette étude consiste à évaluer l'adéquation du service AS pour les ¤ots élastiques. Dorénavant, la source de tra£c doit pouvoir changer son débit en fonction de l'état de charge de la route. Des paquets avec des priorités spatiales différentes vont constituer le ¤ot applicatif. Ainsi, c'est le marquage de la priorité spatiale qui est contrôlé et non plus le débit de génération des paquets. Les mesures sont effectuées en utilisant TCP. Les ¤ux de données reposant sur le protocole de transport TCP présente un caractère élastique de par le mécanisme de contrôle de congestion de TCP. Le débit est dé£ni pour TCP comme la quantité de bits reçus par le récepteur (à l'exclusion des retransmissions) pendant la durée d'un transfert. Dans le cas présent, ceci correspond au débit utile (*goodput*).

La méthode retenue s'appuie sur un ¤ot AS nommé "¤ot de référence" généré depuis BSD_A et son comportement est mesuré selon le nombre de ¤ots supplémentaires générés par la machine BSD_B composant l'agrégat. Les tests se déroulent de la façon suivante :

– La pondération du WFQ utilisée pour AS est équivalente à celle de BE, à savoir 0.5. Le service AS a donc une garantie de 50 % de la bande passante du goulot d'étranglement. Notons R_{AS}, cette garantie de débit.

– aucun biais n'est introduit entre les services par la taille des paquets. Tous les paquets ont une taille de 1024 octets,

– Les delais de propagation entre émetteurs et récepteur sont équivalents.

– un ¤ot BE en UDP est émis en permanence depuis la machine BE (voir £gure 2) à un débit équivalent au goulot d'étranglement a£n de maintenir l'interface de sortie du routeur dans un état de congestion,

– Chaque ¤ot AS est émis en TCP et l'ensemble des ¤ots de l'agrégat commence en même temps pour une durée de 120 secondes. Un ¤ot représente le transfert d'un £chier (*bulk transfer*).

– chaque seconde, le générateur de tra£c donne une évaluation du débit utile du ¤ot de référence. En £n de test, celui-ci fait la moyenne des résultats obtenus et retourne en plus la valeur minimale et maximale. Cette mesure est faite par la machine source BSD_A,

– Le débit du goulot d'étranglement est £xé à 10 Mbits/s. La taille de la £le d'attente PBS (voir Interface de sortie) est £xée à $64Ko$ correspondant à la taille maximum par défaut de la fenêtre TCP et pour seuil, suivant [ZIE 01], la moitié : $32Ko$.

Lorsque des évaluations sont faites avec TCP, il faut prendre garde à ce que la voie de retour des acquittements ne soit pas congestionnée [PAP 01]. Les mesures faites ici ne concernent que l'augmentation du nombre de ¤ot dans un agrégat composé uniquement de segments de données. La voie retour prise par les acquittements est isolée de la voie aller de par la con£guration des expérimentations et de l'utilisation de liens *full-duplex* de la plate-forme de tests.

L'étude du comportement d'un agrégat de ¤ots TCP dans la classe de service AS a déjà suscité de nombreuses études. Dans [SED 99], cinq paramètres (RTT (*Round Trip Time*), nombre de ¤ots, débit voulu, taille des paquets, ¤ots non-réactifs) sont étudiés a£n de déterminer leur rôle respectif dans le service rendu à des ¤ots TCP. Ces cinq paramètres affectent le débit des ¤ots TCP. Cependant, dans un réseau surdimensionné, une assurance de débit peut être donnée indépendamment de ces cinq paramètres. Ces résultats sont corroborés par les travaux de [GOY 00] [REZ 99]. Mais la distribution de la bande passante en excès dépend de ces cinq paramètres La présente étude se situe dans le cas limite d'un réseau convenablement dimensionné et sans bande passante en excès.

2.3. *Résultats des mesures et analyses*

2.4. *Hypothèses de départ*

On pose r_i, le paramètre d'un *token buket* d'un ¤ot i, correspondant au débit des paquets marqués IN. ϕ_{AS} le poids normalisé provisionnant le service AS. n le nombre de ¤ots AS TCP dans l'agrégat au niveau du goulot d'étranglement et C la capacité du lien, plus exactement, cette capacité est représentée par le goulot d'étranglement du réseau. Si on considère un nombre de ¤ots i traversant ce goulot d'étranglement, alors la capacité allouée pour le service assuré R_{AS} correspond à :

$$\sum_{i=1}^{n} r_i \leq R_{AS} \quad \text{avec} \quad R_{AS} = \phi_{AS} * C \tag{1}$$

Dans le cas où :

$$\sum r_i << R_{AS} \tag{2}$$

la bande passante en excès e vaut :

$$e = C - \sum r_i - min(R_{BE}, \phi_{BE} * C) \tag{3}$$

Avec R_{BE}, la bande passante utilisée par les ¤ots BE et ϕ_{BE} le poids normalisé provisionnant le service BE. Cette bande passante se trouve équitablement distribuée entre les ¤ots TCP à partir du moment où les RTT de chaque ¤ots sont identiques. En effet, comme nous le montre la £gure 3 extraite de [MED 01], à partir du moment où un routeur est traversé par des ¤ots TCP ayant des RTT différents, la bande passante n'est plus équitablement distribuée. Dans les tests suivants, a£n de se placer dans le cas général et éviter d'introduire une complexité supplémentaire, le RTT de chaque ¤ot TCP est identique.

Donc, un ¤ot ayant une valeur de marquage de ces paquets IN à r_i devrait obtenir un débit théorique b_i de :

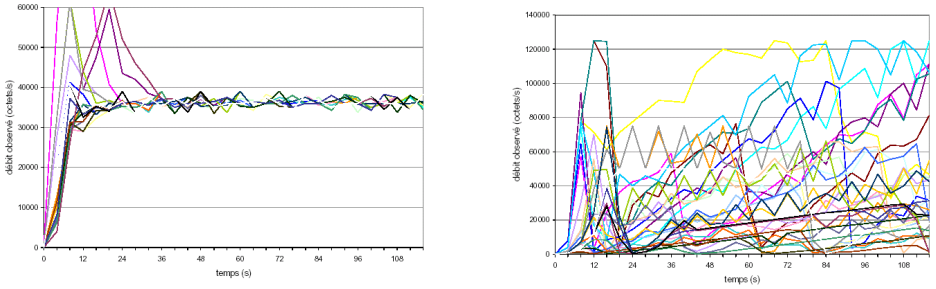
$$b_i = r_i + \frac{e}{n} \tag{4}$$

avec n le nombre de ¤ots TCP traversant le lien. Cette évaluation est bien évidemment valide dans le cas où le lien ne brasse que des ¤ots TCP et devient inutilisable dans le cas d'un brassage avec des ¤ots UDP.

L'état saturation correspond à :

$$\sum_{i=1}^{n} r_i = R_{AS} \tag{5}$$

Dans ce cas, il n'y a pas de bande passante en excès. Lors des tests, il a été remarqué que plus il y a de ¤ots TCP dans l'agrégat, plus le contrat de tra£c est dif£cile à maintenir. Et plus le contrat de tra£c est proche du partage équitable de TCP, plus il est facile à un ¤ot de maintenir son contrat de tra£c.



(a) ¤ux TCP à RTT identiques (b) ¤ux TCP à RTT différents

Figure 3. *Comportement de débit TCP observé par des ¤ux à RTT identiques (a) et différents (b)*

2.4.1. *Étude du service AS pour les ¤ots élastiques*

Dans le cas le plus favorable, un ¤ot de référence en-pro£l est agrégé avec plusieurs ¤ots hors-pro£l. Dans ces conditions, les paquets du ¤ot de référence sont tous marqués IN et sont en concurrence avec uniquement des paquets marqués OUT. Le résultat est illustré par le tableau 1 et par la £gure 4. Le tableau 1 indique qu'à partir de 10 ¤ots hors-pro£l, la £le PBS est pleine et accuse quelques pertes de paquets opportunistes. Notre ¤ot de référence reste quant à lui constant au niveau du délai et de son débit utile moyen, quelque soit le nombre de ¤ots hors-pro£l arrivant au routeur. Ce test démontre qu'un contrôle de la bande passante peut être obtenu par une priorité spatiale [CLA 98]. Ce test dé£nit la borne supérieure de la QoS AS.

Nb. de ¤ots hors-pro£l	0	1	2	3	5	10	20	30	50	100
Débit utile Mbits/s	4.930	4.824	4.823	4.728	4.718	4.697	4.703	4.650	4.598	4.583
RTT en ms	1.662	1.698	1.699	1.733	1.736	1.744	1.742	1.762	1.782	1.787
Perte de paquets IN	0	0	0	0	0	0	0	0	0	0
Perte de paquets OUT	0	0	0	0	0	0.27%	2.36%	4.27%	7.58%	13.4%

Tableau 1. *Débit utile du ¤ot de référence en-pro£l en fonction du nombre de ¤ots hors-pro£l*

Inversement, le pire des cas correspond au ¤ot de référence hors-pro£l et tous les autres ¤ots en-pro£l. Les résultats du tableau 2 montrent que le délai devient in£ni, entraînant un débit nul du ¤ot de référence à l'arrivée du deuxième ¤ot hors-pro£l. Ce test dé£nit la borne inférieure de la QoS AS.

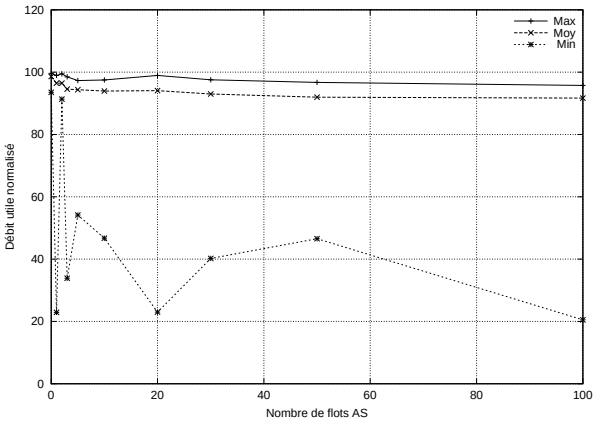


Figure 4. *Débit utile normalisé min, moyen, max du ¤ot de référence en fonction du nombre de ¤ots hors-pro£l*

Nb. de ¤ots en-pro£l	0	1	2 ... 100
Débit utile en Mbits/s	4.962	0.087	0
RTT en ms	1.651	94.573	∞
Perte de paquets IN	0	0	
Perte de paquets OUT	0	14.8%	

Tableau 2. *Débit utile du ¤ot de référence hors-pro£l en fonction du nombre de ¤ots en-pro£l*

Le cas du ¤ot de référence en pro£l en concurrence avec des ¤ots en-pro£l, revient au fonctionnement classique du service BE. En l'absence de biais (comme le RTT), les ¤ots TCP se partagent équitablement la bande passante du goulot d'étranglement [CHI 89]. Le débit du ¤ot de référence tend vers 0 quand le nombre de ¤ots tend vers l'in£ni. Cette caractéristique a été véri£ée sur la plate-forme de test. Le débit est quasiment nul au-delà de 30 ¤ots.

Dans quelle mesure l'assurance de débit peut-elle être contrôlée ?

Pour répondre à cette question, le ¤ot de référence va être conditionné selon des pro£ls différents au moyen d'un *token bucket* de paramètres (r, b). De manière évidente, l'assurance de débit du service AS n'est valable que si le service est convenablement dimensionné. A savoir, tant que la condition suivante est vraie :

$$\sum_{i=1}^{n} r_i \leq R_{AS} \tag{6}$$

avec n le nombre de ¤ots de l'agrégat dans le service au niveau du goulot d'étranglement, r_i le débit assuré du ¤ot i (le débit des paquets IN), R_{AS} le débit garanti au service AS. Dans [REZ 99], il est noté que la taille du seau a une in¤uence sur l'as-

surance de débit pour les ¤ots qui ont une assurance de débit importante dans le cas d'un réseau sur-dimensionné. Le sur-dimensionnement est dé£ni comme :

$$\sum_{i=1}^{n} r_i \leq 0.4 * R_{AS} \tag{7}$$

Cependant, l'augmentation de la taille du seau combinée avec la sporadicité inhérente de TCP accroît les rafales de paquets IN. Dans le cas du dimensionnement utilisé dans la plate-forme de test, des congestions conjoncturelles peuvent se produire pouvant conduire à des pertes de paquet IN. Pour éviter ce genre de situation, la taille du seau b est invariable et est équivalente à la taille d'un paquet, soit 1024 octets.

Selon l'équation 6, le débit assuré d'un ¤ot i de l'agrégat vaut :

$$r_i = \frac{R_{AS} - r_{ref}}{n - 1} \tag{8}$$

avec n nombre de ¤ots au total et r_{ref} le débit assuré du ¤ot de référence. Pour $R_{AS} = 5Mbits/s$, le débit assuré du ¤ot de référence varie de 1 à 4 Mbits/s avec un pas de 1 Mbits/s. La £gure 5 présente les résultats obtenus. Elle montre une forte variation du débit utile moyen du ¤ot de référence lorsque le nombre de ¤ots AS augmente. Le ¤ot de référence n'est pas isolé des conditions de tra£c. De plus, on a observé une augmentation du taux de perte des paquets hors-pro£l du ¤ot de référence quand le nombre de ¤ots AS augmente. En£n, l'augmentation du taux r du *token bucket* du ¤ot de référence ne se traduit pas par une augmentation signi£cative du débit utile pour ce dernier. Quand n tend vers l'in£ni, le débit du ¤ot de référence converge quelque soit son pro£l. En£n, très logiquement, un ¤ot avec un débit minimum faible atteint plus facilement son objectif qu'un ¤ot avec une forte assurance de débit. L'explication provient de TCP qui réagit par un facteur multiplicatif à la perte et par un facteur additif aux transmissions réussies. Il en résulte un temps différent pour retrouver la taille de la fenêtre de contrôle de congestion permettant d'émettre au débit minimum [YEO 99].

Ce test montre également que le *token bucket* est un très mauvais marqueur pour les ¤ots TCP car il ne prend pas en compte la sporadicité de TCP [LIN 99]. Des rafales de paquets hors-pro£l sont émis et peuvent entraîner des pertes en séquence. Il est connu que TCP a des problèmes de performance lorsque la connexion souffre de pertes en rafale [FAL 95].

A noter que ces résultats sont indépendants de la gestion de la £le d'attente du routeur. En effet, il a été observé par des simulations étendues sur les agrégats de ¤ots TCP que la gestion de la £le d'attente du routeur a un faible impact sur le débit utile et le taux de perte [QIU 01]. Cependant, une gestion probabiliste de la £le d'attente tend à rendre le partage de la bande passante plus équitable. Il y a tout de même une diminution linéaire du débit utile de TCP au fur et à mesure que le nombre de connections augmente.

La principale conclusion de cette expérience est qu'il n'est pas possible d'effectuer une différenciation de service entre les ¤ots TCP par un marquage selon un

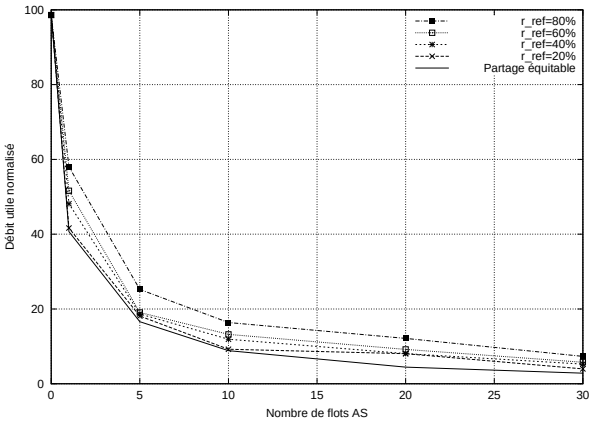


Figure 5. *Débit utile normalisé du ¤ot de référence en fonction du nombre de ¤ots AS*

simple *token bucket*. Cette analyse rejoint celle faite par plusieurs études dont celle de [SAH 00]. En£n, la perte d'un paquet pour TCP, qu'il soit en pro£l ou hors pro£l, est préjudiciable au débit utile moyen du ¤ot. TCP réagit à la perte sans distinction de la priorité associée au paquet. Dans [FEN 97], il est proposé un contrôle de congestion à deux fenêtres : une pour chaque partie du service AS.

3. Quelles sont les solutions envisagées pour obtenir le débit recherché ?

3.1. *Limiter les paquets OUT*

La £gure 6 reprend l'expérience précédente mais une remise en forme du ¤ot de référence est effectuée selon son pro£l (tous les paquets sont en pro£l). Le ¤ot de référence obtient un débit constant quelque soit le nombre de ¤ots dans l'agrégat. Ceci démontre bien que la perte d'un paquet hors-pro£l est préjudiciable au débit assuré et que la politique de marquage a donc un rôle prédominant dans la différenciation de services. Cependant, le lissage du tra£c ne peut être une solution pour obtenir une différenciation de services. Car en émettant aucun paquet hors-pro£l, le débit de la source est limité à son débit minimum (indiqué par le pro£l) et elle ne peut concourir à obtenir la bande passante en excès si il y a. Dans ce cas, le service AS peut devenir un service "moins bien que BE".

3.2. *Rapport de proportionnalité débit/perte*

Cette idée a été introduite par [DOV 00], inspira beaucoup de propositions dans le domaine de la garantie de débit TCP pour le service assuré. Chaque paquet arrivant dans le réseau est marqué soit IN soit OUT en fonction d'un *token bucket* marqueur.

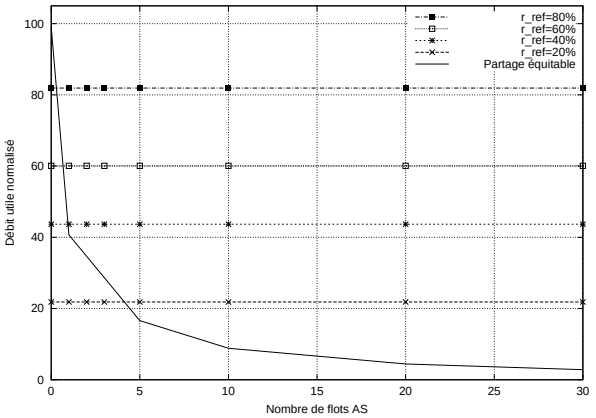


Figure 6. *Débit utile normalisé du ¤ot de référence lissé en fonction du nombre de ¤ots AS*

C'est au niveau de l'ordonnanceur, au sein du routeur, que le traitement s'effectue. En partant de l'équation de modélisation d'un ¤ot TCP donné dans [FLO 99] :

$$t_i \leq \frac{1.5\sqrt{\frac{1}{3}} * B_i}{RTT * \sqrt{p_i}} \tag{9}$$

Avec :

– t_i : le débit visé par le ¤ot i,

– B_i : taille des paquets du ¤ot i,

– p_i : probabilité de perte de paquets du ¤ot i.

Soit deux ¤ots avec deux débits recherchés différents t_1 et t_2 ayant un RTT et une taille de paquet identique, alors suivant l'équation ci-dessus on a :

$$\frac{t_1}{t_2} = \sqrt{\frac{p_2}{p_1}} \tag{10}$$

Si l'on compare le nombre de paquets jetés par unité de temps noté $d1$ et $d2$, on obtient :

$$\frac{d_1}{d_2} = \frac{t_1 * p_1}{t_2 * p_2} = \sqrt{\frac{p_1}{p_2}} = \frac{t_2}{t_1} \tag{11}$$

Ce qui nous indique que le nombre de paquets jetés par unité de temps doit être inversement proportionnel au débit recherché d'un ¤ot. Le résultat de cette technique est identique à celle présentée en section 3.1 dans le sens où il est necessaire d'être prêt du niveau de service pour que celle-ci soit ef£cace.

3.3. *Vers un token bucket marqueur dynamique*

Les propositions de marquage pour TCP se divisent en deux familles : celles qui traitent du marquage d'un ¤ot TCP avec un pro£l sur un agrégat, et celles qui traitent du marquage du ¤ot TCP par rapport à un pro£l individuel. La première vise la propriété d'équité en plus de l'assurance de débit recherchée par la seconde. Plusieurs algorithmes de ce type ont été proposés pour travailler avec le service AS. Le *Two Rate Three Color Marker (TRTCM)* dans [HEI 99a] qui est basé sur un *token bucket* metreur et le *Time Sliding Window Three Color Marker (TSWTCM)* dans [FAN 00], basé sur un estimateur de débit moyen présenté dans [CLA 98]. Dans ces *token bucket* marqueurs, deux débits sont dé£nis : un débit assuré appelé *Commited Information Rate (CIR)* et un débit maximal le *Peak Information Rate (PIR)* utilisé lors d'un surplus de bande passante. Une version enrichie prenant en compte la dynamique de TCP a été proposée dans [LIN 99].

3.3.1. *Marquage adaptatif pour les agrégats de ¤ots*

Dans [YEO 01a], il est décrit un marquage équitable d'un ¤ot TCP dans un agrégat en utilisant un modèle de TCP qui repose sur le RTT et la taille du paquet. En partant du modèle mathématique du comportement d'un ¤ot TCP dé£nit dans [YEO 01b] :

$$b_i = \frac{3}{4} r_i + \frac{3k_i}{4RTT} \sqrt{\frac{2}{p_i}} \tag{12}$$

on pose :

$$b_i = \frac{3}{4} r_i + \epsilon_i \tag{13}$$

Avec b_i, taux d'émission du ¤ot i. On obtient donc le débit d'un paquet IN voulu en fonction du paramètre du *tocken bucket*. Partant de l'équation 13, [YEO 01a] identi£e les cas suivants (t_i correspondant au débit visé par le ¤ot i) :

1) Si $b_i \leq \frac{3}{4} r_i + \epsilon_i < t_i$ alors : nous sommes dans un réseau chargé et donc on accuse des pertes IN. Solution : diminuer le paramètre r_i.

2) Si $\frac{3}{4} r_i + \epsilon_i < b_i < t_i$ alors : le ¤ot n'atteind pas son but. Solution : augmenter le paramètre r_i

3) Si $t_i \leq b_i$ alors : le ¤ot atteind son but. Pour éviter de prendre des ressources pour rien, on réduit le taux de marquage r_i.

Plus récemment, un *token bucket* à con£guration dynamique a été proposé [CHA 02] pour le marquage d'un ¤ot TCP selon un pro£l par ¤ot. D'autres propositions de marquages similaires, basées sur les équations de modélisation de ¤ot TCP [PAD 98], ont été proposées dans [ELG 02].

La solution consisterait donc à faire un *shaping* adaptatif.
Le *shaping* se ferait avant le marquage. Dans [BON 00b], le débit du *shaping* dépend du niveau de remplissage de la £le du *shaper* et du débit mesuré. Plus la £le se remplit, moins il y a de perte dans le réseau (le débit de TCP augmente ou tout du moins sa fenêtre), le réseau est peu chargé. Le rythme d'émission du *shaper* augmente et des

paquets OUT sont générés au-delà du debit assuré. Lorsque la £le diminue, le rythme d'émission diminue également. Dans cette technique, le marquage ne tient pas compte du RTT et utilise uniquement des informations locales tirées du ¤ux de données.

4. Conclusion et travaux futurs

Les travaux présentés dans cet article posent le problème de la garantie de QoS dans l'Internet pour les ¤ots TCP en service AS. Nous avons étudié sur une plate-forme réelle l'in¤uence de plusieurs ¤ots TCP sur un ¤ot TCP de référence. Notre architecture nous permet de garantir un débit d'émission pour un ¤ot totalement marqué IN après contrôle d'admission, au travers du réseau. Et cela, quelquesoit le nombre de ¤ots composant l'agrégat arrivant à un nœud. Ceci est assez peu utile en pratique mais con£rme qu'une isolation entre ¤ots peut être atteinte par une priorité à la perte. En revanche, il est dif£cile dans l'état actuel des choses de dé£nir les caractéristiques de notre classe AS lorsque nous utilisons le *token bucket* à deux priorités à la perte pour le marquage du ¤ot TCP. Le marquage d'un ¤ot TCP dans un agrégat AS ne permettant pas de différencier le débit, il n'est donc pas possible de l'assurer à un ¤ot TCP. Toutefois, on peut identi£er trois approches pour solutionner le problème de la différenciation du débit de ¤ots TCP :

– au niveau de TCP : les solutions posent certains problèmes dans la mise en œuvre. Tout d'abord, il demande que le code de TCP soit modi£é. D'un point de vue de l'architecture DiffServ, le marquage est effectué par la source exclusivement. Dans ce cas, le marquage n'est plus sous la responsabilité du prestataire de réseau. La véri£cation du marquage du client par le prestataire n'est pas non plus sans poser des dif£cultés de réalisation. En£n dans les situations où le marquage est effectué au niveau d'un agrégat, cette solution n'est pas envisageable. [FEN 97] présente une évolution de TCP pour intégrer un marquage en fonction d'un pro£l en tenant compte de l'état du réseau.

– au niveau du conditionnement, il est évident que le *token-bucket* est un très mauvais marqueur de ¤ots TCP, l'objectif serait de calquer un marquage propre à la dynamique de ces ¤ots. Des algorithmes de marquages tentant de solutionner ce problème sont proposés dans [KUM 01] et [YEO 01a] et [BON 00b].

– au niveau de l'AQM[2], de nouvelles techniques d'ordonnancement telles que JoBS [CHR 02] permettent d'imposer des garanties pour les ¤ots de la classe assurée. Ces techniques sont dérivées des mécanismes de proportionnalité introduits par [DOV 00]. Une autre solution serait dans l'inter-agissement que peut avoir l'AQM avec la source TCP. Le drapeau *Ef£cient Congestion Noti£cation* de TCP pourrait être utilisé a£n de contrôler le débit de la source en vue de limiter les paquets marqués OUT entrant dans le réseau.

Dans l'état actuel, la classe AS semble donc mal adaptée aux ¤ots élastiques. La solution passe par le développement d'un mécanisme de marquage adapté à TCP a£n

2. *Active Queue Management*

de contrôler la différenciation de débit entre les ¤ots quelque soit la composition de l'agrégat. Les enseignements tirés au terme de l'évaluation du service AS pour les ¤ots élastiques amènent à penser qu'il est possible d'effectuer une différenciation entre les ¤ots par une priorité spatiale mais que cette différenciation dépend en grande partie du conditionnement.

5. Bibliographie

[BEN 96] BENNETT J. C. R., ZHANG H., « WF2Q : Worst-Case Fair Weighted Fair Queueing », *Proc. of IEEE INFOCOM*, vol. 1, 1996, p. 120-128.

[BLA 98] BLAKE S. et al., « An Architecture for Differentiated Services », RFC n° 2475, décembre 1998, IETF.

[BON 00a] BONALD T., MAY M., BOLOT J.-C., « Analytic Evaluation of RED Performance », *Proc. of IEEE INFOCOM*, vol. 3, Tel-Aviv - Israël, mars 2000, p. 1415-1424.

[BON 00b] BONAVENTURE O., DE CNODDER S., « A Rate Adaptive Shaper for Differentiated Services », RFC n° 2963, octobre 2000, IETF.

[CHA 02] CHAIT Y., HOLLOT C., MISRA V., TOWSLEY D., ZHANG H., « Providing Throughput Differentiation for TCP Flows using Adaptive Two Color Marking and Multi-level AQM », *Proc. of IEEE INFOCOM*, New York, juin 2002.

[CHI 89] CHIU D., JAIN R., « Analysis of the Increase and Decrease Algorithms for Congestion Avoidance in Computer Networks », *Computer Networks and ISDN Systems*, vol. 17, 1989, p. 1-14.

[CHR 02] CHRISTIN N., LIEBEHERR J., ABDELZAHER T., « A quantitative assured forwarding service », *Proc. of IEEE INFOCOM*, vol. 2, New York, NY, juin 2002, p. 864-873.

[CLA 98] CLARK D., FANG W., « Explicit Allocation of Best Effort Packet Delivery Service », *IEEE/ACM Transactions on Networking*, vol. 6, n° 4, 1998, p. 362-373.

[DAV 02] DAVIE B., CHARNY A., BENNETT J., BENSON K., AL., « An Expedited Forwarding PHB (Per-Hop Behavior) », RFC n° 3246, mars 2002, IETF.

[DOV 00] DOVROLIS C., RAMANATHAN P., « Proportional differentiated services, part II : Loss rate differentiation and packet dropping », *International Workshop on Quality of Service*, Pittsburgh, PA, juin 2000.

[ELG 02] EL-GENDY M., SHIN K., « Assured Forwarding Fairness Using Equation-Based Packet Marking and Packet Separation », *Computer Networks*, vol. 41, n° 4, 2002, p. 435-450.

[FAL 95] FALL K., FLOYD S., « Comparison of Tahoe, Reno and SACK TCP », *ACM Computer Communication Review*, vol. 26, n° 3, 1995.

[FAN 00] FANG W., SEDDIGH N., AL., « A Time Sliding Window Three Colour Marker », draft, mars 2000, IETF, Internet Draft, draft-fang-diffserv-tc-tswtcm-01.txt.

[FEN 97] FENG W., KANDLUR D., SAHA D., SHIN K. S., « Adaptive Packet Marking for Providing Differentiated Services in the Internet », rapport n° CSE-TR-347-97, octobre 1997, IBM.

[FLO 99] FLOYD S., FALL K., « Promoting the use of end-to-end congestion control in the Internet », *IEEE/ACM Transactions on Networking*, vol. 7, n° 4, 1999, p. 458–472.

[GOY 00] GOYAL M., DURRESI A., JAIN R., LIU C., « Performance Analysis of Assured Forwarding », draft, février 2000, IETF, Internet Draft, draft-goyal-diffserv-afstdy-00.txt.

[HEI 99a] HEINANEN J., GUERIN R., « A Two Rate Three Color Marker », RFC n° 2698, septembre 1999, IETF.

[HEI 99b] HEINANEN J., BAKER F., WEISS W., WROCLAWSKI J., « Assured Forwarding PHB Group », RFC n° 2597, juin 1999, IETF.

[KUM 01] KUMAR K., ANANDA A., JACOB L., « A Memory based Approach for a TCP friendly Traf£c Conditioner in DiffServ Networks », *Proc. of the IEEE International Conference on Network Protocols - ICNP*, Riverside, California, USA, novembre 2001.

[LIN 99] LIN W., ZHENG R., HOU J. C., « How to Make Assured Service More Assured », *Proc. of the IEEE International Conference on Network Protocols - ICNP*, Toronto, Canada, novembre 1999, p. 182+.

[MED 01] MEDINA O., « Etude Des Algorithmes D'attribution De Priorités Dans Un Internet à Différenciation De Services », These de doctorat, ENST Bretagne, Rennes, France, mars 2001.

[PAD 98] PADHYE J., FIROIU V., TOWSLEY D., KRUSOE J., « Modeling TCP Throughput : A Simple Model and its Empirical Validation », *Proc. of ACM SIGCOMM*, Vancouver, CA, 1998, p. 303–314.

[PAP 01] PAPAGIANNAKI K., THIRAN P., CROWCROFT J., DIOT C., « Preferential Treatment of Acknowledgment Packets in a Differentiated Services Network », *Proc. of IEEE/IFIP International Workshop on Quality of Service - IWQoS*, 2001, p. 187-201.

[QIU 01] QIU L., ZHANG Y., KESHAV S., « Understanding the Performance of Many TCP Flows », *Computer Networks*, vol. 37, n° 3-4, 2001, p. 277-306.

[REZ 99] DE REZENDE J. F., « Assured Service Evaluation », *Proc. of IEEE GLOBECOM*, Rio de Janeiro, Brasil, décembre 1999, p. 100-104.

[ROC] ROCA V., « BENCH : a Network Performance Evaluation Environment », http ://www.inrialpes.fr/planete/people/roca/bench/bench.html.

[SAH 00] SAHU S., NAIN P., DIOT C., FIROIU V., TOWSLEY D. F., « On Achievable Service Differentiation with Token Bucket Marking for TCP », *Measurement and Modeling of Computer Systems*, 2000, p. 23-33.

[SED 99] SEDDIGH N., NANDY B., PIEDA P., « Bandwidth Assurance Issues for TCP Flows in a Differentiated Services Network », *Proc. of IEEE GLOBECOM*, Rio De Janeiro, Brazil, décembre 1999, page 6.

[YEO 99] YEOM I., REDDY N., « Realizing Throughput Guarantees in a Differentiated Services Network », *Proc. of IEEE International Conference on Multimedia Computing and Systems- ICMCS*, vol. 2, Florence, Italy, juin 1999, p. 372-376.

[YEO 01a] YEOM I., REDDY N., « Adaptive marking for aggregated ¤ows », *Proc. of IEEE GLOBECOM*, San Antonio, Texas, USA, novembre 2001.

[YEO 01b] YEOM I., REDDY N., « Modeling TCP behavior in a differentiated services network », *IEEE/ACM Transactions on Networking*, vol. 9, n° 1, 2001, p. 31-46.

[ZIE 99] ZIEGLER T., FDIDA S., HOFMANN U., « RED+ Gateways for Identi£cation and Discrimination of Unfriendly Best-Effort Flows in the Internet », *IFIP Broadband Communications*, 1999, p. 27-38.

[ZIE 01] ZIEGLER T., FDIDA S., BRANDAUER C., « Stability Criteria of RED with TCP Traf£c », *IFIP ATM&IP Working Conference*, Budapest, juin 2001.