
De l'utilisation des mesures de trafic pour l'ingénierie des réseaux de l'Internet

N. LARRIEU, P. OWEZARSKI

LAAS-CNRS
7, avenue du Colonel Roche
31077 TOULOUSE Cedex 4
{nlarrieu, owe}@laas.fr

RÉSUMÉ. La métrologie est en train de devenir la pierre angulaire de l'ingénierie et de la recherche en réseau, notamment en ce qui concerne l'Internet. En effet, la complexité grandissante de l'Internet entraîne une méconnaissance du trafic et des conditions d'utilisation du réseau. Dans de telles conditions, il devient de plus en plus difficile d'avoir une réelle maîtrise du réseau, de son comportement et du trafic. Nous proposons dans cet article une nouvelle « méthodologie métrologique » qui intègre des phases de mesure dans toutes ces étapes. La première consiste à caractériser le trafic et sa dynamique. La seconde consiste à utiliser la métrologie pour améliorer le réalisme des simulations de façon à confronter les architectures et protocoles évalués à des conditions réalistes. Au niveau de la caractérisation du trafic, il apparaît que des phénomènes oscillatoires, très dommageables sur la QoS, apparaissent à toutes les échelles d'observation du trafic. Ces oscillations semblent être associées à des mécanismes de dépendance longue mémoire (LRD). Cet article s'attache donc à étudier l'impact de la LRD sur la QoS ainsi que les causes de cette LRD, notamment les mécanismes de contrôle de congestion de TCP. Cet article propose donc pour illustrer cette « méthodologie métrologique » des modifications de TCP pour réduire les phénomènes oscillatoires du trafic et ainsi améliorer la QoS du réseau.

ABSTRACT. Monitoring is becoming the cornerstone of network engineering and research, in particular for the Internet. Because of the increasing complexity of the Internet, the traffic characteristics and network usages are badly known. In such conditions, it become harder and harder to fully control the network and its traffic. We propose in this paper a new “monitoring based methodology” that integrates in all its stages some measurements. First stage consists in characterizing the network traffic and its dynamics. The second consists in using monitoring to improve the realism of simulations, in order to evaluate architectures and protocols in realistic environments. First results on traffic characterization show an oscillatory phenomenon, at all scales, that is a major issue for QoS. These oscillations seem to be associated to some long range dependence feature (LRD) in the traffic. This paper then studies possible causes of LRD and how LRD can impact QoS. In particular TCP congestion control mechanisms are deeply analyzed. This paper then proposes to illustrate our “monitoring based methodology”. For that, it is proposed to modify some TCP features to reduce LRD and then improve network QoS.

MOTS-CLÉS : métrologie Internet, caractérisation du trafic, optimisation, qualité de service, contrôle de congestion, TCP, TFRC.

KEYWORDS: Internet monitoring, traffic characterization, optimization, quality of service, congestion control, TCP, TFRC.

1. Introduction

L'Internet est en train de devenir le réseau universel pour tous les types d'informations, du transfert simple de fichiers binaires jusqu'à la transmission de la voix, de la vidéo ou d'informations interactives en temps-réel. L'Internet se doit donc de fournir de nouveaux services adaptés aux applications Internet et aux données qu'elles transmettent. De plus, l'Internet croît très rapidement, en taille (nombre d'utilisateurs, d'ordinateurs connectés, etc.) et en complexité, en particulier à cause de la nécessité d'offrir de nouveaux services et d'optimiser l'utilisation des ressources de communication pour améliorer la qualité de service offerte aux utilisateurs.

A cause de la complexité grandissante de l'Internet, l'évolution de ce réseau global est indissociable d'une parfaite connaissance et compréhension des caractéristiques du trafic. Par conséquent, le développement d'outils et de techniques de métrologie réseau pour capturer le trafic Internet ainsi que de méthodologies pour analyser ses caractéristiques est aujourd'hui un sujet d'ingénierie et de recherche de premier plan. La métrologie – au sens littéral « la science des mesures » – est en train d'apparaître dans de nombreux domaines du monde des réseaux comme :

- la caractérisation et la modélisation du trafic,
- l'analyse du trafic et du réseau (comportement),
- l'ingénierie des trafics,
- l'optimisation de la QoS¹ et des performances,
- la tarification,
- la sécurité,
- l'administration de réseau,
- etc.

Parmi les domaines d'étude et de recherche concernés par la métrologie, celui de la QoS dans l'Internet n'est certainement pas le moindre. En effet, les propositions faites dans la communauté Internet, ces dix dernières années, pour offrir des QoS différenciées et garanties n'ont pour l'heure pas complètement abouti et continuent à se heurter à de nombreuses difficultés liées à la complexité des interconnexions de réseaux et à leur hétérogénéité de ressources. En particulier, définir et quantifier la QoS dans l'Internet reste un problème non résolu. Les approches issues du monde de la téléphonie et basées sur des métriques simples comme le débit, le délai ou le taux de perte restent insuffisantes pour modéliser entièrement et finement les caractéristiques et performances du trafic Internet. La modélisation du trafic dans son ensemble est une tâche à réinventer. Les travaux les plus récents ont essayé de décrire les phénomènes oscillatoires du trafic Internet, dont les pics sont à la base

¹ QoS : Quality of Service

des difficultés rencontrées par les chercheurs et les ingénieurs réseaux pour mettre au point des techniques de garantie de la QoS. Ces travaux de modélisation sont parvenus à montrer que le trafic Internet est très loin des modèles simples Poissonniens et Markoviens utilisés dans le monde de la téléphonie, et que les modèles qui représentent mieux le trafic Internet sont des modèles ayant des propriétés d'auto-similarité ou de dépendance longue mémoire (LRD²) [Paxson et al., 1995].

La métrologie des réseaux de l'Internet doit permettre d'apporter une réponse à ces questions concernant le (ou les) modèle(s) de trafic de l'Internet qui font aujourd'hui défaut. En particulier, la méconnaissance du trafic Internet est vraisemblablement à la base des difficultés rencontrées pour la mise en œuvre de mécanismes de garantie de QoS, car il était alors impossible de confronter des solutions théoriques à des conditions réalistes de trafic. Ceci est vrai notamment dans le cadre des simulations Internet qui représentent aujourd'hui un des outils les plus utilisés dans le domaine de la conception de ce réseau. Or, ces simulateurs font l'hypothèse de modèles de trafics de fond qui sont loin d'intégrer toutes les caractéristiques et toutes les irrégularités que les premiers travaux de métrologie ont mis en évidence [Floyd et al. 2001b], et de fait les solutions avancées pour améliorer l'Internet se sont avérées mieux fonctionner lors des simulations que dans l'Internet réel [Floyd et al., 2002].

Le travail décrit dans cet article se propose donc de définir une nouvelle méthodologie pour l'ingénierie des réseaux qui intègre une phase de métrologie. Cette phase de métrologie, suivie d'une analyse du trafic capturé, a pour objectif de bien définir, voire même de permettre de trouver un modèle pour le trafic Internet, qui décrira le plus précisément possible les contraintes réelles liées au trafic et à son environnement. La partie 2 de cet article aborde donc les problèmes de caractérisation du trafic Internet. L'objectif est d'en déterminer les caractéristiques intéressantes (à défaut de les déterminer toutes) afin de pouvoir réfléchir à des solutions pour améliorer l'Internet, qui tiennent compte des contraintes liées au trafic et au comportement des utilisateurs, des équipements et des protocoles. Cette partie proposera également quelques premiers éléments pour la définition de paramètres de QoS pour le trafic Internet ainsi que des métriques pour pouvoir la quantifier. Fort de ces métriques, nous pourrons alors mesurer les effets bénéfiques ou non de telle ou telle proposition architecturale ou protocolaire prétendant améliorer la QoS de l'Internet : c'est la première contribution de notre proposition de méthodologie basée sur la métrologie. La seconde contribution de cette méthodologie, décrite dans la partie 3, consiste à utiliser les résultats de métrologie pour obtenir des simulations de l'Internet plus réalistes que ce qui était fait jusqu'à présent. Deux approches existent dans ce domaine : les approches reposant sur des descriptions formelles, intégrant des modèles mathématiques afin d'offrir des preuves et des validations du bon fonctionnement des solutions proposées, et des

² LRD : Long-Range Dependence

approches informelles basées sur des simulateurs ad-hoc, comme NS³ par exemple, très utilisé dans la communauté Internet. Vu les difficultés évoquées précédemment par rapport à la modélisation du trafic, nous nous sommes orientés vers une approche informelle. Notre proposition pour rendre réalistes des simulations consiste à utiliser les traces issues de la métrologie, afin de rejouer dans le simulateur des trafics reprenant le comportement des utilisateurs et des composants matériels et / ou protocolaires du réseau. Enfin, la partie 4 de cet article illustre cette nouvelle méthodologie sur un cas concret qui est celui de l'amélioration de la QoS dans l'Internet. En particulier, à partir des caractéristiques du trafic que nous avons pu observer sur quelques liens de l'Internet, nous évaluons quels pourraient être les effets sur la QoS du réseau si le mécanisme de slow-start de TCP était modifié ou si le mécanisme de contrôle de congestion TFRC⁴ de DCCP⁵ était utilisé à la place de celui de TCP.

A noter que ce travail est réalisé dans le cadre du projet METROPOLIS qui est un projet national soutenu par le RNRT⁶ sur le thème de la métrologie des réseaux.

2. Caractérisation et modélisation du trafic et de la QoS

2.1. Les paramètres caractéristiques du trafic et de la QoS

2.1.1. Besoins et contexte

Caractériser le trafic Internet est une activité extrêmement difficile de nos jours. Le modéliser l'est encore plus. En effet, l'Internet est aujourd'hui un réseau multi-services dont l'objectif est de transporter un grand nombre de types de trafic possédant des caractéristiques différentes. De façon grossière, on dénombre deux grandes classes de trafic :

– Le trafic de type « **stream** », dont la caractéristique principale – le débit – doit être maintenu assez régulier. Ceci implique donc de préserver une certaine intégrité sur les profils temporels de ces trafics souvent « orientés connexion », et donc de maintenir à un niveau régulier les délais (et ses variations : la gigue), les taux de pertes (qui doivent rester en général relativement bas), etc. Ces trafics sont caractéristiques d'applications comme la téléphonie IP, la visioconférence ou le téléchargement de séquences audio ou vidéo en ligne.

– Le trafic « **élastique** » dont le débit peut s'adapter à des contraintes extérieures (bande passante disponible par exemple) sans pour autant remettre en cause la

³ NS : Network Simulator

⁴ TFRC : Tcp-Friendly Rate Control

⁵ DCCP : Datagram Congestion Control Protocol

⁶ RNRT : Réseau National de Recherche en Télécommunications

viabilité du service. Cette classe de trafic est essentiellement engendrée par le transfert d'objets numériques tels que des pages Web, des messages électroniques ou des transferts de fichiers, etc.

La caractérisation du trafic « stream » est assez aisée car directement inspirée des travaux effectués dans le cadre de la téléphonie. Par contre, caractériser le trafic élastique, et donc le trafic Internet, qui est composé en large majorité de trafic élastique (aux alentours de 90 à 95% actuellement et même parfois plus [Thompson et al., 1997]), n'est pas aussi aisée. En effet, même si les modèles issus du monde des télécommunications pour caractériser le trafic Internet sont encore utilisés, il est clair aujourd'hui que les débits, délais ou taux de pertes constants ne suffisent pas à caractériser et/ou modéliser le trafic élastique – et donc le trafic Internet – et tous ses aspects dynamiques.

D'autre part, il existe plusieurs niveaux de modélisation du trafic qui correspondent à des entités réelles et différentes du trafic : les *paquets* qui correspondent aux unités élémentaires de traitement par les couches réseau, les *flux* qui sont des séries consécutives de paquets ayant des caractéristiques communes, et souvent dépendantes des applications ou des protocoles de transport (comme TCP) et les *sessions* définies plus ou moins aléatoirement de façon à se rapprocher au plus près des comportements des utilisateurs ou de leurs applications (par exemple).

La modélisation du trafic est également complexe de par les mécanismes de superposition des services de natures très différentes, de dispositifs de contrôle (notamment ceux de TCP) aux comportements chaotiques [Veres, 2000], engendrant des processus d'apparition très complexes, complexité encore accrue par le nombre d'utilisateurs de tous les services disponibles.

Etant donnée la difficulté de travailler simultanément sur les 3 niveaux (paquets, flux, sessions), la suite de ce travail ne considère que le niveau paquet, et se concentrera principalement sur le trafic TCP, dominant dans l'Internet.

2.1.2. Quelques éléments sur la modélisation du trafic de paquets dans l'Internet

Les travaux de ces dernières années sur la modélisation du trafic des paquets dans l'Internet ont montré qu'un trait important concernant la nature du trafic Internet est son aspect auto-similaire (ou possédant des dépendances à long terme, ou encore de caractère fractal, sans trop entrer dans les subtilités mathématiques qui différencient ces trois notions voisines). L'auto-similarité du trafic de données a été mise en évidence dans [Leland et al., 1994] et souvent confirmée depuis : la structure des variations d'amplitude du signal analysé (par exemple le nombre d'octets ou de paquets transférés par unité de temps ou la série des durées inter-paquets) se reproduit de manière similaire quelle que soit la finesse temporelle avec laquelle il est représenté. Le comportement d'un trafic auto-similaire est à l'opposé de celui d'un trafic poissonnien, pour lequel les variations d'amplitude sont filtrées au fur et à mesure que l'on augmente la taille de la fenêtre d'intégration. Cette auto-similarité, voire cette multi-fractalité, et son extrême variabilité à toutes les échelles

de temps est caractéristique du principe même de la commutation de paquets qui induit des transmissions en rafales [Roberts, 2000] [Tran Gia et al., 2000]. Ce comportement se caractérise notamment par une décroissance lente⁷, par exemple sous forme de loi de puissance, sous exponentielle ou à queue lourde [Heyman, 1998], de la fonction d'autocorrélation du nombre de paquets transférés par unité de temps (typiquement 100 ms) : les processus auto-similaires sont des cas particuliers des processus à dépendance à long terme (LRD).

Même pour les applications « stream », la caractéristique de LRD se retrouve dans les transferts de séquences vidéo à débit variable (VBR selon la terminologie ATM) [Beran et al., 1995], probablement due à la variabilité des paramètres de transmission liés au codage des trames (MPEG, par exemple), à la dynamique des images, etc.

Concernant le trafic de type élastique, l'identification du processus des paquets tel qu'il est offert au réseau est particulièrement délicate. En effet, les dispositifs de correction d'erreur et de perte génèrent la retransmission de paquets supplémentaires et les mécanismes de contrôle de flux (TCP notamment) régulent les débits de transmission [Black, 1992]. Les analyses de trafic doivent donc se contenter des données de trafic effectivement mesurées sur des liens, compte tenu de ces retransmissions et régulations.

Le caractère auto-similaire du trafic TCP a été largement étudié [Leland et al., 1993] [Leland et al., 1994] [Crovella et al., 1997]. En complément de ce qui a été dit au paragraphe précédent, notons les tentatives d'explication avancées : aux échelles de temps supérieures à un délai de transmission typique (RTT, de l'ordre de 100 ms), le comportement auto-similaire serait dû à l'extrême variabilité de la taille des documents transférés (la loi de distribution est de type « heavy-tailed », telle la loi de Pareto) ; tandis que les caractéristiques multi-fractales aux échelles de temps inférieures seraient provoquées par les mécanismes de contrôle de congestion du protocole TCP [Feldmann et al., 1998 a] [Feldmann et al., 1998 b]. C'est aussi la conclusion à laquelle [Crovella et al., 1997] est arrivé lorsqu'il a analysé le trafic HTTP (dominant à cette époque dans le trafic Internet). De la même manière, [Leland et al., 1994] a montré que le trafic Internet peut être représenté par un processus ON/OFF dont la distribution des durées des périodes ON est à queue lourde.

2.1.3. Illustration de la problématique

Pour illustrer la notion d'auto-similarité, la figure 1 représente une courbe journalière de trafic sur laquelle sont mis en avant les phénomènes oscillatoires sur des périodes courtes et sur des périodes longues (la journée), et si des échelles différentes étaient choisies, il apparaîtrait le même phénomène oscillatoire.

⁷ Ce phénomène de décroissance lente signifie que la probabilité d'obtenir des très grandes valeurs de la variable aléatoire est asymptotiquement beaucoup moins faible que pour une loi de type exponentiel.

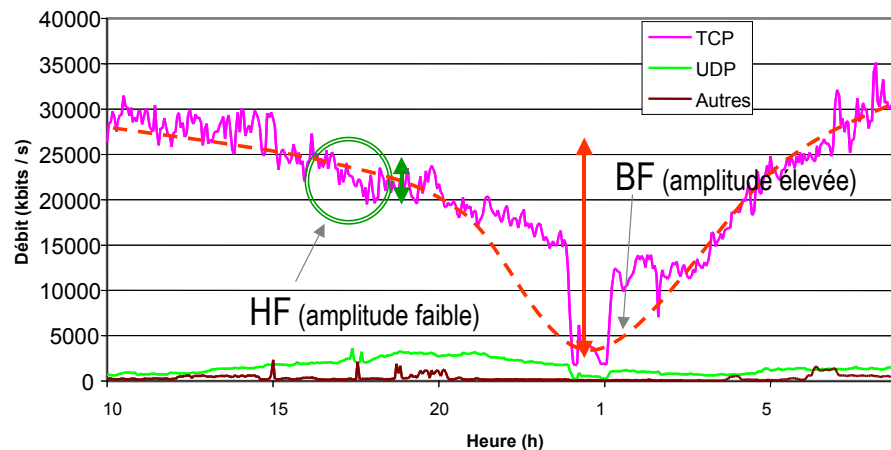


Figure 1 : Evolution du trafic Internet au cours du temps (24 heures)

Clairement, ces phénomènes oscillatoires sont une source de baisse de la QoS sur les flux du trafic [Erramilli et al., 1996]. En effet, la charge et donc les ressources disponibles étant très variables, les flux transitent dans des conditions très changeantes, et de brutales variations sur certains flux peuvent perturber d'autres flux : des congestions peuvent apparaître, provoquer des pertes, et ainsi contraindre les applications à réduire leur débit d'émission, conduisant à une baisse de QoS. Il semble donc impossible dans ces conditions, sans un mécanisme d'isolation de certaines parties du trafic de garantir une QoS de bout en bout et pendant toute la durée de la communication (l'isolation de certaines parties du trafic est, par exemple, un des principes mis en place avec la technologie ATM mais qui est difficilement applicable à l'Internet). En allant plus loin, c'est même le niveau de performance du réseau qui est impacté par les oscillations des flux du trafic qui influent sur les mécanismes de reprise d'erreur et de contrôle de congestion qui en dépendent.

Par conséquent, ce phénomène oscillatoire apparaît comme un paramètre important de QoS et de performance du réseau. Une des contributions de ce travail consiste donc à évaluer l'impact des phénomènes d'auto-similarité et/ou de LRD sur la QoS, à analyser comment ces perturbations interviennent et se traduisent, et à essayer de les quantifier.

2.2 Etude métrologique des caractéristiques du trafic

2.2.1. Détails de la méthodologie

Pour illustrer ce point de vue sur l'impact de l'auto-similarité et/ou LRD sur la QoS, nous avons mené une étude métrologique sur plusieurs liens de types différents.

Les deux types de trafics analysés sont les suivants :

- des traces capturées en bordure d'un réseau universitaire sur un lien d'accès à 10 Mbps.
- des traces capturées sur le réseau de cœur d'un opérateur, sur des liens OC-3 (155 Mbps) et OC-12 (622 Mbps).

Les outils d'analyse statistique qui ont été utilisés sont les suivants :

- la représentation quantile-quantile (Q-Q Plot) de la loi d'arrivée des paquets TCP : il s'agit d'une méthode de représentation graphique qui permet de déterminer visuellement si la loi qui régit la série de données que l'on analyse peut s'approcher par une loi connue (dans notre cas, la loi exponentielle).
- La fonction d'autocorrélation – ou autocovariance – de la loi d'arrivée des paquets TCP : cet outil statistique permet de mesurer la relation de corrélation (et donc de dépendance) qui existe entre différents paquets plus ou moins « éloignés » dans la trace que l'on analyse.
- Le coefficient de LRD de la loi d'arrivée des paquets : il permet de mesurer le phénomène de dépendance inter-paquets que l'on peut observer dans le trafic à des échelles de temps importantes. Pour calculer ce coefficient, appelé facteur de Hurst, on s'appuie sur une analyse par « ondelettes » et un outil développé pour Matlab par Abry et Veitch [Abry et al., 1998].

2.2.2. Réseaux de bordure

Dans un premier temps, cet article se focalise sur l'analyse statistique réalisée sur les traces du réseau d'accès. La figure 2 représente le diagramme Q-Q Plot de la loi d'arrivée des paquets TCP pour la trace Ethernet à 10 Mbps et la figure 3 la fonction d'autocorrélation de la même trace.

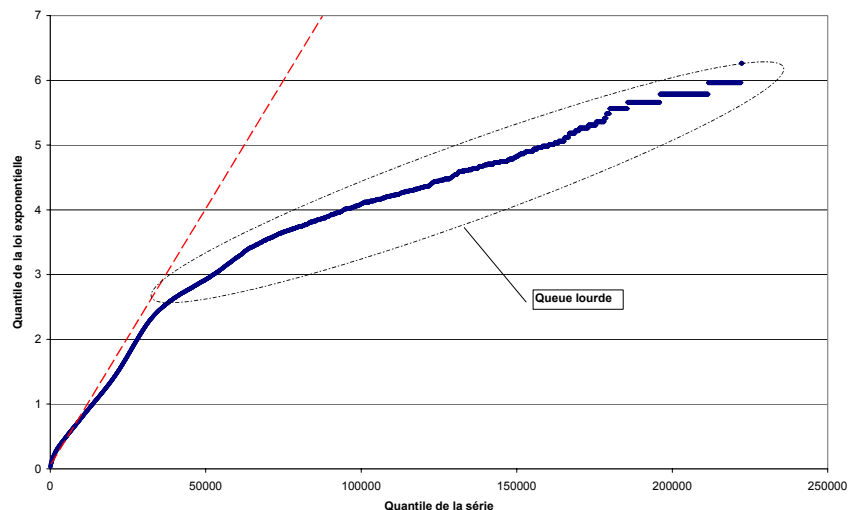


Figure 2 : Représentation Q-Q Plot de la loi d'arrivée des paquets TCP (réseau d'accès)

On constate que la série présente d'une forte queue lourde dans la représentation en Q-Q plot qui ne permet pas de se rapprocher des lois poissonniennes ou markoviennes classiques des réseaux de télécommunications (cf. Figure 2). Pour compléter notre étude métrologique, nous avons effectué un calcul du paramètre de Hurst (H) caractéristique de l'auto-similarité et de la LRD. Cela va nous permettre d'analyser de manière plus quantitative la présence d'auto-similarité et de longue mémoire dans le trafic. Ainsi, après évaluation par la méthode des « ondelettes », le facteur de Hurst vaut **0,806**. Cette valeur, bien supérieure à 0,5, indique que la série de données possède de la dépendance longue mémoire.⁸

D'autre part, on observe que la série est très fortement corrélée (70% des points sont en dehors des bornes de confiance d'un comportement Gaussien, cf. Figure 3) et que sa fonction d'autocorrélation présente un comportement complexe (chute lente, oscillations basses fréquences). Le processus des arrivées de paquets apparaît donc complexe et ne peut s'assimiler à une loi indépendante.

⁸ Le parameter de Hurst ne peut dépasser 1.

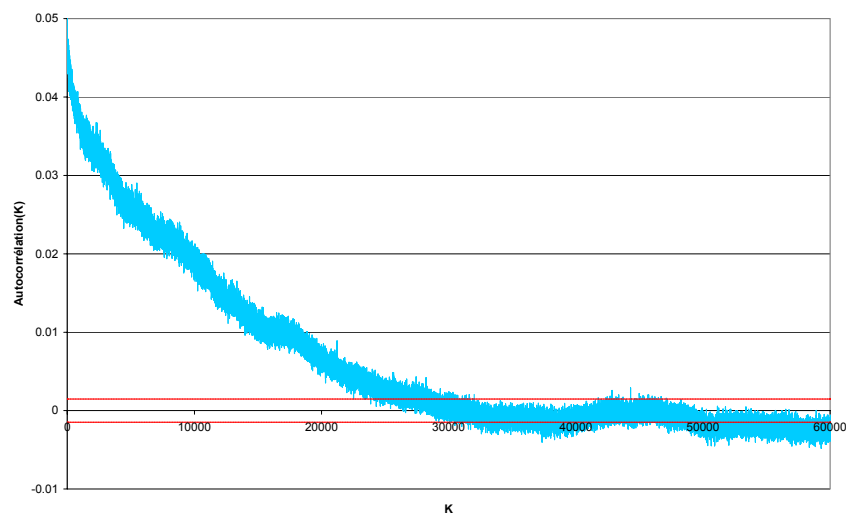


Figure 3 : Fonction d'autocorrélation de la loi d'arrivée des paquets TCP (réseau d'accès)

2.2.3. Réseaux de cœur

Dans un deuxième temps, nous présentons la même analyse mais appliquée à la trace du réseau de cœur.

En ce qui concerne la loi d'arrivée des paquets TCP, on observe que sa représentation en Q-Q plot est quasi linéaire (cf. Figure 4) : dans la mesure où ce Q-Q plot est basé sur la loi exponentielle, ceci indique que la loi d'arrivée des paquets correspond bien à une exponentielle. Le fait que les arrivées des paquets suivent une loi « simple » est confirmé lorsqu'on trace la fonction d'autocorrélation de la série de données (cf. Figure 5). En effet, on constate que la loi étudiée est très faiblement corrélée : seulement 7% des points sont en dehors des bornes de confiance (on considère qu'en dessous de 5%, la corrélation est nulle).

Le résultat du calcul de Hurst obtenu pour la loi d'arrivée des paquets TCP est le suivant : $H = 0,501$. Cette valeur, très proche de 0,5, indique que la série de données est très peu dépendante à long terme. Ajouté à la faible corrélation sur le court terme, cela confirme que la série correspond à une loi « simple », en l'occurrence dans le cas présent à une loi poissonnienne.

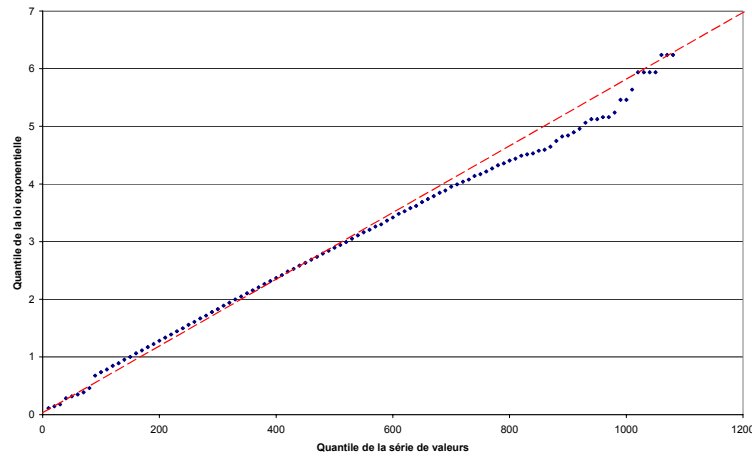


Figure 4 : Représentation en Q-Q plot de la loi d'arrivée des paquets TCP (réseau de cœur)

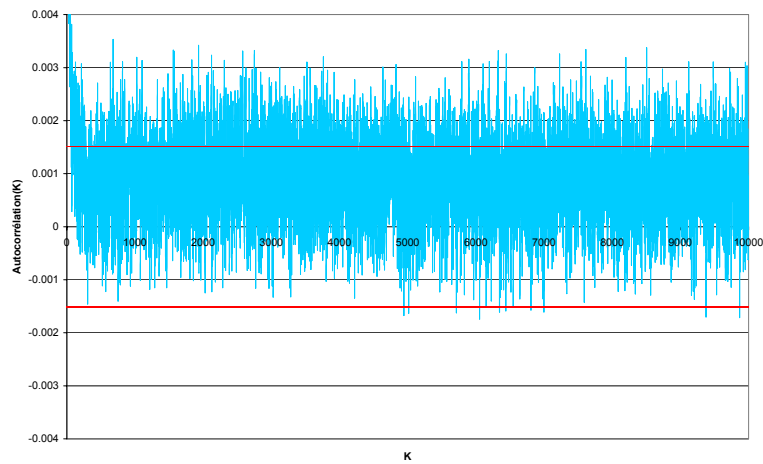


Figure 5 : Fonction d'autocorrélation de la loi d'arrivée des paquets TCP (réseau de cœur)

Ces deux analyses ont été reproduites sur un petit nombre de traces et le résultat obtenu a été sensiblement le même : le trafic de cœur est proche d'un trafic poissonnien, alors que le trafic de bordure est beaucoup plus irrégulier, corrélé et dépendant à long terme. Si on compare ce résultat sur le facteur de Hurst à des

paramètres de QoS classiques, on s'aperçoit que là où H est grand, le réseau présente une courbe de trafic très irrégulière avec des variations de forte amplitude (surtout sur des échelles de temps longues) et des taux de pertes plus élevés que sur les liens où le trafic présente un facteur de Hurst bas.

2.3. Conclusions

Les deux exemples précédents ne peuvent naturellement pas se généraliser. Toutefois, ils illustrent l'importance du paramètre de LRD dans la définition et la quantification de la QoS du trafic Internet. Le paramètre de Hurst semble donc représenter une métrique intéressante de la QoS du trafic, même si le travail qui a été présenté dans cette partie est loin d'être abouti et si les effets de la LRD sur la QoS doivent encore être approfondis. Sur un plan qualitatif, par contre, ce premier résultat démontre, s'il en était besoin, que le trafic Internet reste extrêmement complexe, et nécessitera encore beaucoup de travail avant d'en comprendre toutes les caractéristiques et de pouvoir en donner un modèle proche de la réalité.

Par rapport, à l'analyse des deux exemples de cette partie, même s'il est impossible de généraliser les observations, il est à noter que ces résultats rejoignent les résultats de [Cao et al., 2001], à l'origine de nombreux débats dans la communauté Internet, qui défend une théorie opposée à celle de l'auto-similarité pour le trafic de cœur. En effet, cette étude montre que plus la quantité de trafic augmente, plus le trafic devient régulier, avec des distributions d'arrivées exponentielles (et plus sous-exponentielles ou à queue lourde) et une dépendance qui diminue (évaluée à l'aide de méthodes entropiques). Le modèle de trafic ne serait ainsi plus aussi complexe à partir du moment où une certaine quantité importante de paquets, issus d'un grand nombre de flux, s'entrelacent sur un lien haut débit. Ce résultat semble donc éliminer l'hypothèse de l'auto-similarité du trafic Internet, car si le trafic de bordure était auto-similaire, alors le trafic de cœur le serait aussi, ce qui n'est pas le cas ici. Seule la notion de dépendance longue mémoire est donc présente dans les caractéristiques du trafic.

3. Vers des simulations de réseaux plus réalistes avec la métrologie

3.1. Problématique de la simulation de réseaux de l'Internet

3.1.1. Pourquoi est-il si difficile de simuler l'Internet ?

De telles caractéristiques des trafics observés sur les quelques liens que nous avons étudiés montrent bien que le problème de la modélisation des trafics de l'Internet sera une tâche ardue. En effet, de nombreuses approches reposant sur des

modèles connus comme les modèles poissonniens, Markoviens, des modèles ON/OFF [Leland et al., 1993], des modèles de files d'attente, $M/G/1/N$, $M_k/G_k/1/N$ [Garcia et al., 2001], des mouvements fractionnels Browniens caractéristiques des comportements auto-similaires, ou même des modèles à base de fractales [Feldmann et al., 1998 a] ne parviennent pas à représenter toutes les caractéristiques du trafic Internet sur un lien. Cette difficulté dans la modélisation des trafics engendre également des difficultés pour réaliser des simulations réalistes de l'Internet. C'est ce qui est mis en évidence dans [Floyd et al., 2001 b] qui montre qu'il est extrêmement complexe de simuler de tels trafics, en particulier à cause des caractéristiques d'auto-similarité, LRD ou multi-fractalité qui ont pu être mises en évidence lors des premières analyses de trafic de liens de l'Internet. Les résultats du paragraphe 2 ont bien montré que cette difficulté est une réalité, déjà sur un seul lien. Or, il apparaît également que les caractéristiques, et donc le modèle pour les représenter, sont différents d'un lien à un autre, sans qu'il soit aujourd'hui possible de connaître les règles de dépendance d'un lien à l'autre. L'ingénierie des réseaux de l'Internet à partir de modèles formels du trafic est donc une activité à développer, et qui prendra certainement de nombreuses années. Pour pouvoir continuer à développer et améliorer l'Internet actuel, il est donc essentiel de mettre en place des techniques de simulations réalistes. Cette notion de réalisme est un problème majeur des moyens de simulations qui existent aujourd'hui. Par exemple, dans les simulateurs actuels, les sources de trafic sont généralement des sources régulières comme des générateurs constants ou respectant des processus d'émission markoviens, soit dans tous les cas des sources de trafic plus régulières que le trafic de l'Internet. Cette régularité est assez dommageable pour le réalisme des simulations actuelles car les trafics à étudier ne sont pas confrontés aux contraintes réelles du trafic, mais à des contraintes moins dures. Souvent, les protocoles ou nouvelles architectures issues de ces simulations, et qui donnaient satisfaction lors des simulations ne fournissaient pas les mêmes résultats lors des déploiements en environnement réel [Floyd et al., 2002]. Nous nous sommes donc attachés à utiliser les résultats de mesures de trafic pour améliorer le réalisme des environnements dans lesquels sont faits les simulations Internet.

3.1.2. Les deux approches de simulation

De façon évidente, la métrologie modifie le processus d'ingénierie réseau en ajoutant en amont une phase de caractérisation et d'analyse du trafic, comme cela a été fait dans la partie 2 de cet article. D'autre part, vu les difficultés à modéliser le trafic, les approches entièrement formelles qui pouvaient être utilisées jusqu'à présent dans l'ingénierie des réseaux et des protocoles, sont aujourd'hui difficilement applicables et le resteront tant qu'un modèle formel du trafic ne sera pas trouvé. De fait, nous sommes forcés de nous rabattre sur une approche de simulation informelle – dite événementielle – avec un simulateur comme NS-2 par exemple, qui est le simulateur officiel de l'IETF. Avec ce type de simulateur, comme avec des simulateurs utilisant des approches formelles, il est difficile d'obtenir des résultats réalistes pour les mêmes raisons que précédemment : les

sources de trafic ne prennent pas en compte toutes les caractéristiques d'irrégularité du trafic Internet, ni qualitativement, ni quantitativement. Pour cela, l'approche que nous proposons consiste à utiliser la métrologie en rejouant les traces capturées par les équipements de métrologie dans le simulateur, de façon à avoir des sources de trafic réalistes et reproduisant les comportements des utilisateurs et de leurs applications. Il est à noter que d'autres projets de recherche en rapport avec la simulation sont en train de voir le jour et devraient permettre une plus grande interaction entre la simulation et les traces réelles de trafic comme par exemple dans le projet Entrapid [Huang et al., 1999]. D'autre part, il existe plusieurs travaux basés sur l'émulation de réseau comme les projets PlanetLab [PlanetLab] ou Emulab [White et al., 2002]. La problématique du réalisme des sources de trafic est identique en émulation et en simulation, et l'approche décrite dans la suite est adaptable aux deux cas. La suite va donc présenter comment fonctionne le module de jeu de traces de métrologie que nous avons développé.

3.2. Le jeu de traces de métrologie

3.2.1. Principes du jeu de traces

Cette approche, même si elle nous fait perdre tout l'aspect formel de la simulation, présente l'avantage de n'introduire aucune modification sur les caractéristiques du trafic ce qui constitue un atout fondamental pour le réalisme de cette simulation. De plus, l'aspect formel est ici inutile car le simulateur recommandé par l'IETF et utilisé le plus souvent pour les simulations de l'Internet (NS-2) n'utilise pas de modèle formel et n'offre pas de moyens de vérification, d'évaluation des performances ou de validation formelles. Nous avons donc travaillé à l'intégration de nouvelles fonctionnalités permettant le jeu de traces réseaux réelles dans le simulateur NS-2.

Pour pouvoir rejouer des traces, en fait, il faut pouvoir reproduire le comportement des utilisateurs et des applications qu'ils utilisent. Autrement dit, il faut pouvoir recréer la dynamique d'apparition des flux, et rejouer ces flux avec des caractéristiques réalistes, comme la taille de chaque flux, le nombre de paquets composant chacun d'eux, etc. Pour cela, nous avons isolé les deux paramètres nous apparaissant comme les plus importants pour conserver le profil du trafic initial dans le simulateur :

- les instants de début de chaque flux dans le profil général qui représentent le comportement des utilisateurs et plus généralement de la couche application [Leland et al., 1994].

- la taille de chacun des paquets émis à l'intérieur d'un flux qui conditionne les caractéristiques de queue lourde des distributions que l'on peut observer en analysant par exemple la taille des fichiers échangés dans l'Internet [Crovella et al., 1998].

En aucun cas, la dynamique interne du flux, c'est à dire la dynamique temporelle d'arrivée des paquets dans ce flux n'est à reproduire par rapport à la trace originelle. En effet, ce sont les sources de trafic du simulateur utilisé (implémentant les protocoles classiques ou éventuellement de nouveaux protocoles en cours de discussion dans les groupes de standardisation comme l'IETF), qui vont servir à générer ces flux, en fonction de la topologie utilisée dans la simulation et de ses caractéristiques (débits, délais, etc.). A la charge de l'utilisateur du simulateur ensuite de recréer une topologie, des matrices de trafic, etc. pour recréer un environnement de simulation restituant au mieux les RTT pour chaque flux, les goulots d'étranglements, etc. En effet, le simple fait de rejouer des traces réelles de trafic ne garantit pas que le trafic mesuré en simulation sera réaliste. [Joo et al, 2001] a déjà utilisé des techniques de rejeu sans pour autant que le résultat soit positif. En effet, dans [Joo et al., 2001], l'environnement de simulation tel qu'il a été mis en place ne permettait pas de recréer toutes les caractéristiques du trafic et notamment les caractéristiques dynamiques.

La suite présente les deux étapes pour réaliser des simulations à partir de traces de trafic issues de la métrologie. La première consiste à extraire les informations sur la dynamique des flux, la seconde sur la création d'une topologie qui permette de rejouer le trafic dans des conditions imitant la topologie proche du point où la trace de trafic a été capturée.

3.2.2. *Extraction des informations de rejeu des traces et rejeu dans NS-2*

Un ensemble d'outil a été développé pour réaliser toutes les opérations nécessaires, à savoir :

- mettre en forme la trace pour pouvoir la rejouer dans le simulateur. Pour cela, il est nécessaire d'extraire l'ensemble des flux contenus dans la trace initiale et de ne garder que les instants de départ des flux et les séries respectives des tailles de paquets pour chacun des flux, à enregistrer dans un fichier.

- Ensuite, associer à chaque flux, une entité du simulateur (cf. l'objet « traffic/trace » de NS), qui va permettre de faire le lien entre la définition du flux sous forme de fichiers de données et l'objet qui va permettre son rejeu dans le simulateur.

- Enfin, « connecter » de façon logique l'objet représentant le flux à un agent de haut niveau (par exemple un agent TCP) pour permettre l'émission dans le réseau de l'ensemble des données constituant le flux.

3.3. *Principes d'utilisation du module*

Dans la seconde étape pour reproduire au mieux la dynamique de rejeu des paquets au sein des différents flux, il est important de générer des topologies qui soient capables de reproduire les paramètres ayant une influence importante sur le fonctionnement des protocoles. Pour illustrer la mise en place de cette topologie

dans le simulateur, la suite prend l'exemple du taux de perte – qui est un paramètre essentiel influant sur le comportement de TCP et sur le profil du trafic généré – et de la matrice de trafic originelle – de façon à ce que les mêmes contentions entre les mêmes flux que dans la trace originelle apparaissent dans la simulation. Il faut alors s'inspirer de la phase de caractérisation réalisée sur les traces métrologiques, qui permet notamment d'évaluer le taux de perte sur chaque flux de la trace utilisée et d'obtenir la matrice de trafic.

Ensuite, on procède en deux étapes. Tout d'abord, on définit des classes de flux en fonction de leurs taux de perte, une classe étant composée de flux ayant des taux de perte proches (la largeur d'une classe de perte peut être de 5 ou 10 % par exemple). Cette décomposition en classe permet ensuite de déterminer les caractéristiques des routeurs du réseau simulé, notamment par rapport à la taille de leurs files d'attente et de la bande passante du lien que les flux de chaque classe vont emprunter. Ainsi, avec la connaissance du débit moyen des flux des différentes classes et de leur taux de perte moyen, on peut déterminer la capacité du lien pour chaque classe de flux, ainsi que la taille de la file d'attente associée sur le routeur dans la simulation. En effet, la qualité des composants électroniques, des fibres optiques et autres câbles étant telle, les erreurs sont très rares, et les pertes sont presque exclusivement liées à des congestions des files d'attente. D'autre part, les délais qui seront simulés sur ces liens sont déterminés en fonction du RTT moyen mesuré sur l'ensemble des flux d'une classe. Dans un second temps, on s'intéresse aux adresses source et destination de chacun des flux. Ainsi, on peut « reconstruire » la topologie de simulation de façon à ce que tous les flux qui ont été émis ou reçus à partir du même domaine réseau (en fait qui sont arrivés ou partis par un même lien) traversent lors de la simulation un même lien. On obtient ainsi des topologies de simulation qui s'approchent de la topologie initiale sur laquelle la trace métrologique a été prélevée.

Cette méthode de rejeu implémentée dans le simulateur NS-2 a été testée et validée sur un grand nombre de traces réelles. Les tests ont permis de mettre en évidence que les comportements initiaux des flux TCP contenus dans les traces rejouées étaient correctement reproduits en simulation. Ainsi, la figure 6 montre une comparaison des distributions des lois d'arrivées de paquets dans le cas d'une trace réelle et de cette même trace rejouée en respectant la méthode présentée ci-dessus. Ces deux distributions sont très proches et ce résultat montre que la mise en forme des paquets dans le simulateur se fait de façon analogue à ce qui se passe dans le réseau réel. La différence qui apparaît est relative aux paquets qui se suivent de très près. Ces paquets proches temporellement sont plus nombreux dans la trace réelle que dans la trace simulée. Ce résultat est lié au fait que nous avons considéré la moyenne des RTT sur les flux des différentes classes, ce qui élimine donc de la simulation les connexions ayant de faibles RTT et qui sont celles qui peuvent engendrer des émissions de paquets très rapprochées.

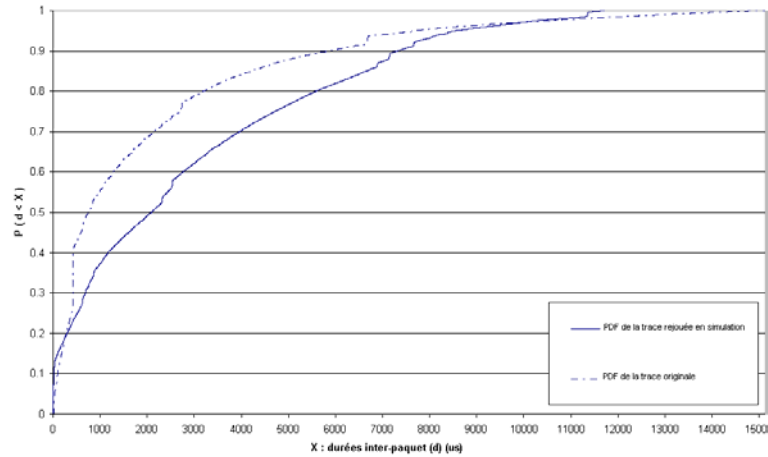


Figure 6 : Comparaison des distributions des durées inter-paquets pour les traces réelle et simulée

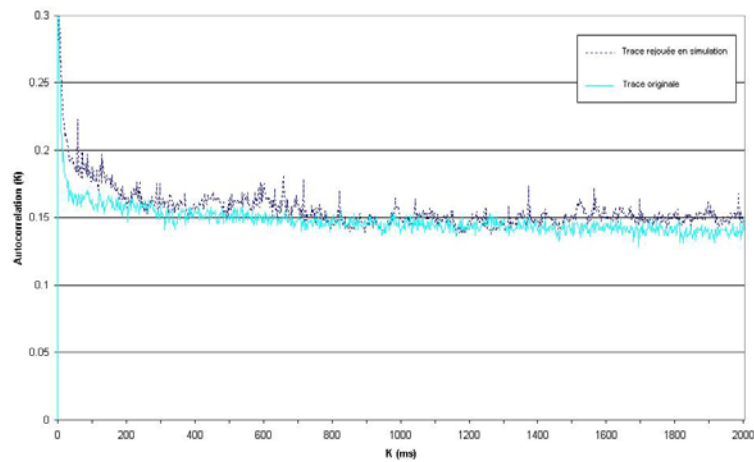


Figure 7 : Comparaisons des fonctions d'autocorrélation pour les traces réelle et simulée

Au premier ordre, les résultats de notre méthode sont donc très satisfaisants. Il en est de même à l'ordre deux, où notre méthode de rejeu des traces permet de très bien reproduire la corrélation qui existe entre les paquets (Figure 7). Cette figure

montre que les courbes sont quasiment superposées, et donc que la dynamique du trafic est respectée.

Au final, notre méthode de rejeu s'avère donner des résultats particulièrement satisfaisants, puisque les traces rejouées en simulation sont extrêmement réalistes en présentant les mêmes caractéristiques statistiques que les traces réelles.

3.4. Conclusion

Dans cette partie, nous avons présenté une nouvelle approche pour la simulation réseau qui permet de s'affranchir des difficultés actuelles pour trouver un modèle formel de trafic. Celle-ci permet, en s'appuyant sur la méthode de caractérisation présentée dans la partie 2, de rejouer des traces réseaux et ainsi de disposer d'un environnement de simulation réaliste et proche, par ces caractéristiques, de celui observable dans l'Internet.

Une illustration de l'utilisation de la méthodologie de caractérisation du trafic ainsi que du module de rejeu va être présentée dans la partie suivante. Ainsi, dans ce qui va suivre, nous allons illustrer l'utilisation de notre méthodologie sur des propositions protocolaires pour optimiser la QoS de l'Internet⁹. Parmi ces propositions, nous évaluerons l'apport de TFRC¹⁰ qui est en cours d'étude à l'IETF.

4. Illustration : propositions théoriques pour améliorer la QoS dans l'Internet

Au cours de la partie 2, nous avons présenté une nouvelle méthode de caractérisation et d'analyse du trafic reposant sur des informations apportées par la métrologie. Ainsi, à l'issue de la phase de caractérisation du trafic, nous disposons de nouveaux paramètres (la corrélation et la dépendance longue mémoire) nous permettant de quantifier la QoS de façon plus précise qu'auparavant. En effet, ces paramètres vont amener des informations statistiques complémentaires à celles apportées par des paramètres plus traditionnels de QoS tels que le délai, le débit ou encore le taux de perte : en effet, les paramètres d'autocorrélation et de LRD sont des quantités qui vont permettre de mieux quantifier les aspects dynamiques du trafic.

D'autre part, l'utilisation des traces issues de la métrologie pour disposer d'un environnement de simulation réaliste (cf. le module de rejeu présenté dans la partie

⁹ Il faut noter, toutefois, que l'objectif de la partie 4 sera d'illustrer notre méthodologie mais que les propositions que nous allons faire restent à l'état d'étude. Des questions importantes, notamment par rapport au déploiement de ces solutions ne seront pas abordées.

¹⁰ TFRC : TCP-Friendly Rate Control

précédente), va nous permettre de pouvoir tester et valider expérimentalement – mais dans un environnement réaliste – des propositions de nouveaux mécanismes.

4.1. Une nouvelle expression de la problématique de la QoS dans l'Internet

Comme cela a été montré, le trafic réseau possède des caractéristiques complexes et fortement oscillatoires. En effet, sur la Figure 1, on observe la présence d'une loi d'échelle dans le trafic qui entraîne la répétition d'un phénomène oscillatoire dans celui-ci. Il apparaît également que l'amplitude des oscillations hautes fréquences (HF) est bien moindre que les oscillations basses fréquences (BF)¹¹.

A partir de cette observation, il apparaît que le problème le plus urgent à régler concerne la réduction des oscillations BF et donc une régulation sur du long terme, même s'il faut aussi être attentif aux oscillations HF, toutes les oscillations ayant un effet dégradant sur la QoS du trafic.

Nous nous sommes donc posé la question de savoir quelles étaient les causes de ces oscillations ? S'il apparaît évident que les mécanismes de contrôle de congestion de TCP ont une influence sur les oscillations HF, il est moins aisé de conclure sur les causes des oscillations BF. Toutefois, si on en croit [Crovella et al., 1997] ces oscillations BF pourraient être liées à la taille des flux transportés. Au cours de la caractérisation du trafic, la taille des flux a été étudiée. Il apparaît donc sur la figure 8 qu'un grand nombre de petits flux (ou « souris ») représente une très faible proportion de la bande passante totale. Ce résultat s'inverse lorsqu'on considère les grands flux (ou « éléphants »). En effet, seuls quelques très longs flux (plus de cent¹² paquets) véhiculent plus de la moitié de la bande passante du réseau.

¹¹ Il est impossible de représenter tous les phénomènes oscillatoires se produisant à toutes les échelles sur une seule courbe de trafic. La Figure 1 présente une courbe de trafic vue avec une granularité d'une seconde. On voit donc apparaître des oscillations HF à l'échelle 1 seconde, et des oscillations BF qui sont ici des variations journalières. Si on étudie ce trafic à d'autres échelles, on retrouve le même phénomène oscillatoire.

¹² Le choix des bornes pour les « souris » (10 paquets) et pour les « éléphants » (100 paquets) est un choix arbitraire qui correspond à la valeur des bornes traditionnellement observée dans les travaux de recherche connexes.

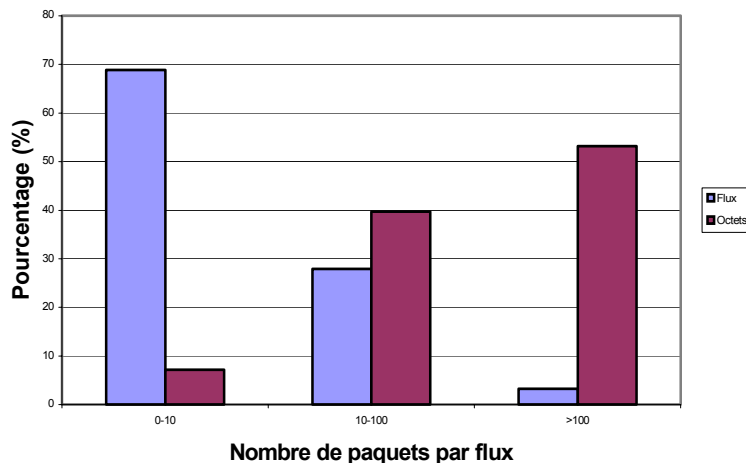


Figure 8 : Taille des flux TCP vis à vis de la bande passante totale

En s’inspirant des résultats de cette caractérisation, il semble donc important d’apporter plus de stabilité aux flux « éléphants ». En effet, ils vont engendrer des phénomènes de dépendance très longue mémoire entre tous les paquets d’un même flux, à cause de leur taille. Nous proposons donc pour optimiser la QoS perçue par les utilisateurs d’utiliser une typologie différente de ce qui est fait habituellement, et de proposer une différenciation des traitements pour les flux courts et les flux longs, comme décrit ci-après :

- **un service « souris »** : il s’agit de trouver un mécanisme permettant d’augmenter les performances des flux « souris ». En effet, ces flux sont si courts qu’ils ne peuvent pas quitter la phase d’émission en mode SS¹³ avant leur terminaison. Ainsi, ils restent avec une fenêtre d’émission CWND¹⁴ réduite pendant toute la durée du flux. Pour se ramener à une terminologie de classification plus classique, ces flux correspondent à du trafic généré par des applications comme le mail ou le web, c’est à dire qu’ils correspondent dans la grande majorité des cas à du trafic « best effort » et élastique.

Comme ces « souris » ne représentent qu’une faible proportion du trafic, il nous semble que le SS est ici un mécanisme inutile. Nous pensons aussi que la transmission de ces flux « souris » pourrait être optimisée – sans que cela ne dégrade la QoS du réseau – si ces flux étaient transmis sans mécanisme de contrôle

¹³ SS : Slow-Start

¹⁴ CWND : Congestion WiNdoW ou fenêtre de congestion

de congestion, i.e. si tous les paquets étaient émis dans la même fenêtre¹⁵. C'est cette solution que nous allons mettre en œuvre dans notre simulateur et l'évaluer.

– **un service « éléphant »** : il s'agit donc pour ces flux de lisser le débit au cours du temps (c'est à dire supprimer le comportement oscillatoire observable sur de grandes échelles de temps). En effet, ce type de flux correspond de plus en plus à des flux ayant des contraintes temporelles générés par des applications comme des visioconférences, de la Vidéo à la demande, de la téléphonie sur IP, etc., c'est à dire des applications qui ont besoin de services dits garantis, comme le service EF de DiffServ par exemple¹⁶.

Pour accroître la régularité des flux « éléphants », nous allons tester l'apport du nouveau mécanisme de contrôle de congestion TFRC qui a été conçu pour fournir un service adapté aux applications multimédias, en essayant d'éviter les variations brutales de débit qui surviennent avec TCP lors des reprises de pertes.

A noter que dans les deux cas, nous estimerons l'évolution du comportement oscillatoire du trafic par l'évolution du facteur de Hurst sur les séries d'arrivées de paquets.

D'autre part, cette proposition d'utiliser des protocoles de transport différents en fonction de la taille des flux à transmettre reste dans cet article à l'étude théorique, servant à illustrer les bienfaits de notre méthodologie d'ingénierie réseaux basée sur des mesures de trafic. Mais les questions relatives à la mise en œuvre de notre proposition ne sont pas abordées. En effet, notre approche aurait besoin pour être déployée de connaître a priori la taille des flux qui vont être transmis de façon à déterminer sa classe (souris ou éléphant) et donc le protocole à utiliser. Une autre approche possible pourrait être de demander ce paramètre à l'utilisateur au moment de l'ouverture de la connexion, et de contrôler ensuite que le paramètre transmis correspond bien à la réalité (contrôle permanent du contrat à l'ouverture). Toutefois, ces questions de mise en œuvre et de déploiement sont en dehors des objectifs de cet article. Ils seront étudiés dans des travaux futurs.

¹⁵ Des travaux ont déjà été menés par [Allman et al., 2002] pour déterminer la taille optimale pour la CWND initiale dans le protocole TCP. Les résultats montrent une amélioration des performances de TCP si on augmente la CWND initiale de 1 paquet (ce qui est le standard) jusqu'à entre 4 et 8 paquets, mais qu'ensuite des phénomènes de congestion peuvent apparaître plus fréquemment. Si on s'en tient à ces résultats, et comme nous ne proposons cette modification de TCP que pour les flux souris (de moins de 10 paquets en tout, paquets de contrôle compris), nous devrions augmenter le débit efficace généré sur le réseau.

¹⁶ Les flux éléphants ne comportent naturellement pas que des flux temporellement contraints, mais aussi des flux de type FTP (généralement transmis en utilisant des applications comme le web ou les applications de « peer to peer »). Toutefois, par rapport aux oscillations BF qui apparaissent sur le trafic, il apparaît utile de réguler aussi le débit généré par ces flux de type FTP, qui dans tous les cas tireront profit de la régulation du trafic de chaque flux éléphant et donc du trafic global sur le lien.

4.2. Suppression du Slow-Start pour les « souris »

Le premier mécanisme propose donc une modification du protocole TCP New Reno pour les « souris ». Il s'agit donc de supprimer toute forme de contrôle de congestion, ce qui rend le protocole plus « agressif » (i.e. capable de consommer plus de ressources rapidement). Nous allons montrer que ce choix (augmentation de l'agressivité) se justifie par la faible quantité d'informations que représentent les « souris » (cf. Figure 8) et par la faible surcharge sur le réseau qu'ils occasionnent.

En simulant sous NS augmenté de notre module de rejeu cette version modifiée de TCP on observe que la durée d'émission des flux « souris » est fortement diminuée (cf. les deux courbes de la partie gauche de la Figure 9) avec l'utilisation du nouveau mécanisme tandis que la durée d'émission des autres flux n'est pas modifiée (cf. les deux courbes confondues de la partie droite de la Figure 9).

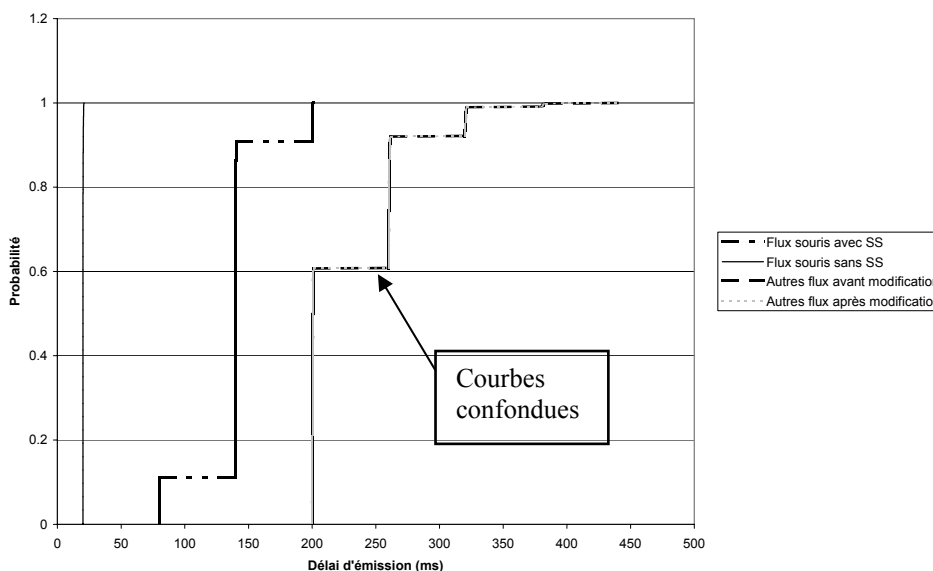


Figure 9 : Distribution cumulée du délai d'émission

De plus, le débit des flux « souris » émis soit avec TCP (dans sa version New Reno), soit avec TCP sans SS reste quasiment inchangé (cf. les deux courbes dans la partie inférieure de la Figure 10). De même pour le débit des autres flux qui reste inchangé lorsque les flux « souris » sont émis sans SS ou avec TCP New Reno (cf. les deux courbes dans la partie supérieure de la Figure 10).

Ces deux mesures ont donc mis en évidence le comportement positif de ce nouveau mécanisme, en terme de diminution du temps d'émission des flux

« souris » par exemple. Elles montrent aussi que la modification de TCP pour les « souris » n'agit pas de façon néfaste sur l'état du réseau – le débit reste quasiment inchangé – et cette modification n'a donc aucune influence sur le niveau de congestion du réseau.

Ces premières observations sont pour le moins positives, au moins sur les paramètres classiques de QoS. On peut maintenant s'intéresser aux paramètres statistiques de QoS (corrélation, dépendance longue mémoire) que nous avons mis en évidence lors de la caractérisation du trafic.

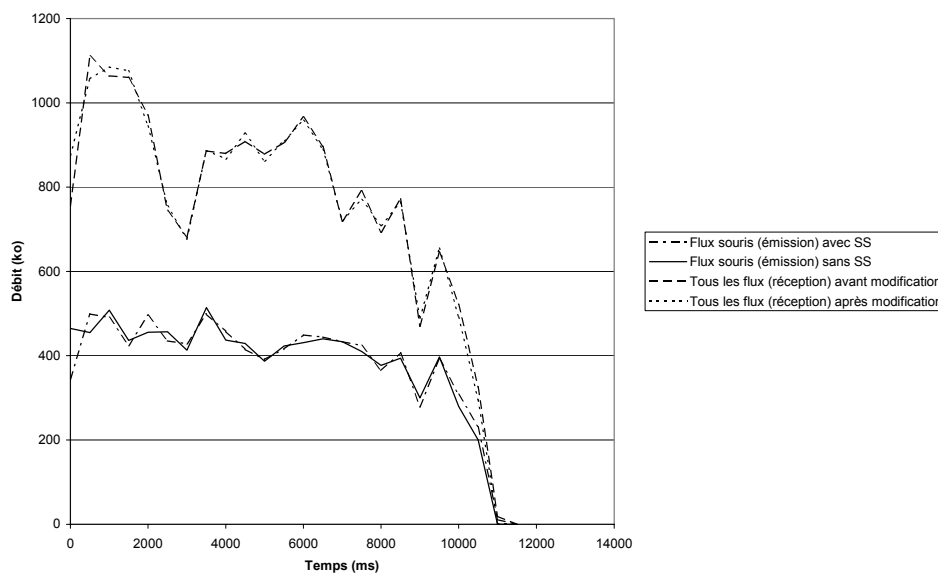


Figure 10 : Variation du débit avant et après la suppression du SS pour les flux "souris"

Ainsi, le calcul de la fonction d'autocorrélation sur le trafic avec (cf. Figure 12) et sans (cf. Figure 11) le nouveau mécanisme d'émission des flux « souris » montre que ce dernier permet une diminution des composantes à hautes fréquences de la fonction d'autocorrélation traditionnellement présentes dans un trafic où seul TCP New Reno est utilisé. Cette diminution va se traduire par un accroissement de la stabilité du comportement réseau. Ce résultat illustre le comportement positif de ce mécanisme pour l'état du réseau qui permet de diminuer la corrélation présente entre les paquets TCP des flux « souris » en supprimant le mécanisme d'émission par fenêtre de données.

De plus, une évaluation de la dépendance longue mémoire du trafic montre que ce nouveau mécanisme permet de diminuer de façon conséquente le facteur de Hurst

qui passe à **0,515**. Ce résultat est cohérent avec le principe qui consiste en la suppression du mécanisme d'émission par fenêtre qui ne va plus introduire une dépendance arbitraire entre les paquets des différentes fenêtres d'émission (les paquets d'une même fenêtre peuvent en effet être considérés comme indépendants car l'émission de l'un ne dépend pas de l'arrivée de l'acquittement de l'autre). Par contre, ce résultat montre que la corrélation et la dépendance ne sont pas complètement supprimées. Nous pensons que la corrélation et la dépendance résiduelle sont dues à l'utilisation du protocole TCP New Reno pour les flux non « souris » qui représentent la grande partie du trafic en terme de bande passante (même si cette corrélation est faible lorsque TCP New Reno fonctionne en régime stationnaire avec de grandes fenêtres CWND, atteignables avec les flux éléphants).

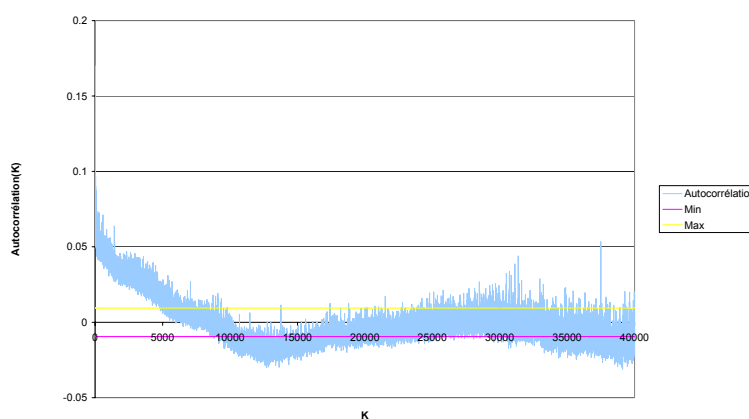


Figure 11 : Fonction d'autocorrélation des durées inter-paquets lorsque les flux « souris » sont émis avec TCP New Reno

Dès lors, on peut affirmer que la suppression du mécanisme de SS pour les flux « souris » est bénéfique à la fois pour les utilisateurs et pour le réseau. En effet, elle apporte un gain direct en terme de QoS pour les utilisateurs et contribue à stabiliser l'état du réseau (diminution de la corrélation et de la LRD) en diminuant les phénomènes oscillatoires observables dans les HF.

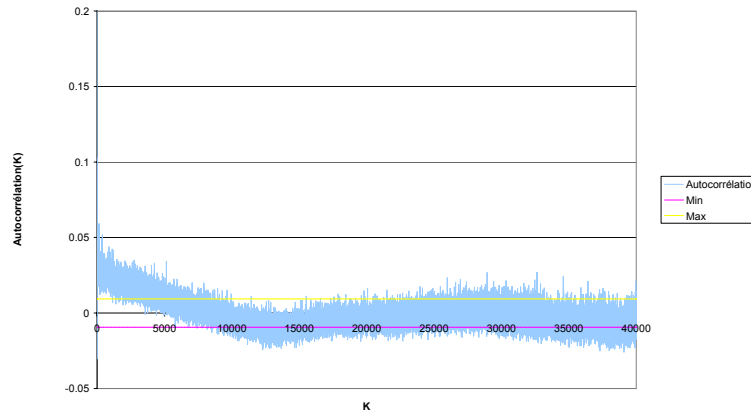


Figure 12 : Fonction d'autocorrélation des durées inter-paquets lorsque les flux « souris » sont émis sans SS

4.3. Analyse comparative de TCP et TFRC

Le second point des améliorations de la QoS dans l'Internet proposées porte sur la diminution du caractère oscillatoire du trafic observable sur de longue échelle de temps. Ce paramètre apparaît comme le phénomène le plus dangereux pour la stabilité du réseau à long terme. C'est pourquoi, dans l'objectif de réguler le trafic du réseau nous proposons d'utiliser le nouveau mécanisme de contrôle de congestion TFRC sur les flux « éléphants ».

4.3.1. TFRC

L'intérêt de TFRC est de proposer aux applications qui l'emploient des montées en puissance plus douces et un débit beaucoup plus régulier au cours du temps que le protocole TCP New Reno. Ainsi, en associant ce mécanisme à l'émission des flux « éléphants », nous avons la possibilité de réguler le trafic Internet de façon efficace étant donné que ces flux représentent la majorité de la bande passante utilisée sur le réseau (cf. Figure 8). Les performances de ce mécanisme sont dues à un fonctionnement orienté récepteur, qui calcule, une fois par RTT, le débit à l'émission en fonction de l'évolution du taux d'événement pertes¹⁷ collecté par le récepteur ([Floyd et al., 1998], [Floyd et al., 2000], [Floyd et al., 2001 a]).

¹⁷ Un événement perte est constitué de l'ensemble des pertes qui peuvent apparaître dans un même RTT. En procédant ainsi, le modèle de dépendance des pertes est brisé puisque toutes les pertes qui sont dépendantes sont regroupées dans un même événement de perte. La récupération en sera ainsi facilitée. A noter que cette

4.3.2. Impact de TFRC sur la QoS

De part sa conception, TFRC permet d'offrir aux flux de données qui l'utilisent un débit très régulier. Pour illustrer les avantages de l'utilisation de TFRC nous allons nous intéresser aux paramètres « statistiques » de QoS : variabilité, fonction d'autocorrélation et dépendance longue mémoire.

La première remarque concernant les améliorations apportées par TFRC a trait à la variabilité du trafic. En effet, en observant le rapport :

$$\text{Coefficient de stabilité} = \text{Moyenne du trafic échangé} / \text{Ecart-type du trafic échangé}$$

On remarque (cf. Tableau 1) que ce coefficient est bien plus grand dans le cas de l'utilisation de TFRC (coefficient = **9**) que dans le cas de l'utilisation de TCP New Reno (coefficient = **2**). Ainsi, on peut affirmer que l'utilisation de TFRC à la place de TCP New Reno entraîne une amélioration significative dans la stabilité du trafic échangé.

	TCP New Reno	TFRC
Moyenne (ko)	73,171	4,739
Ecart-type (ko)	36,391	0,523
Coefficient de stabilité	2,011	9,061

Tableau 1 : Analyse de la variabilité du trafic sur la totalité des flux échangés

La deuxième remarque concerne la corrélation que l'on peut observer dans le trafic généré (cf Figure 13). Cette dernière est très fortement diminuée par l'utilisation du mécanisme TFRC. Cette dernière remarque constitue un nouvel argument positif pour l'utilisation du mécanisme TFRC dans l'échange des gros flux de l'Internet étant donné sa contribution à l'obtention d'un trafic plus régulier et beaucoup moins variable à long terme.

La troisième remarque porte sur la diminution du caractère de dépendance longue mémoire du trafic généré. En effet, en comparaison d'un trafic échangé uniquement avec le protocole TCP New Reno, le facteur de Hurst, avec l'utilisation de TFRC, est très fortement diminué pour atteindre une valeur de **0,352**.

Ainsi, il apparaît bien que TFRC permet d'accroître de façon substantielle la stabilité du trafic pour lequel il est employé. Son utilisation pour le transfert des flux « éléphants » permet donc une amélioration importante à la fois pour les utilisateurs (en terme de stabilité du service fourni) et pour le réseau (en terme de stabilité du trafic généré) de la régularité du service offert.

approche suit les résultats de [Zhang et al., 2001] sur l'étude des pertes dans l'Internet.

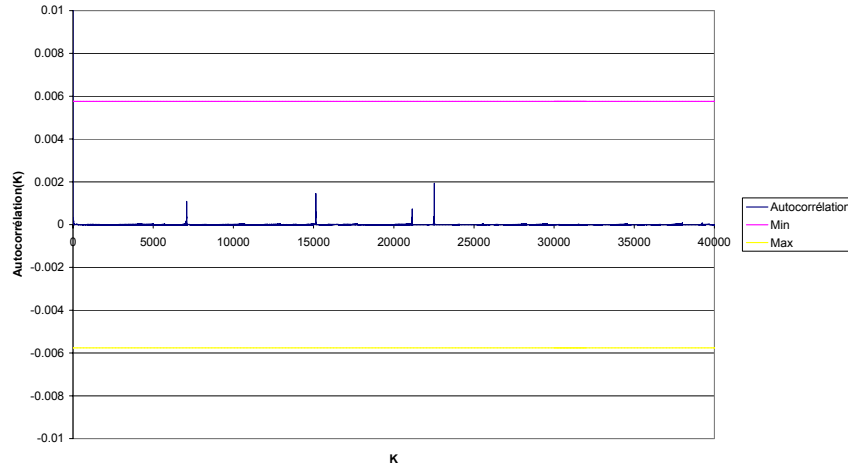


Figure 13 : Fonction d'autocorrélation des durées inter-paquets du trafic TFRC

4.3.3. Impact global des modifications proposées

Les améliorations de QoS obtenues de façon indépendante pour l'échange des flux « souris » et des flux « éléphants » sont des résultats intéressants. Néanmoins, un résultat plus fort serait de pouvoir tester la complémentarité de ces deux mécanismes lorsqu'on les place en concurrence, même si leur complémentarité semble intuitive ; en effet, ces deux mécanismes agissent sur des parties du trafic bien distinctes (flux courts et flux longs), il est nécessaire de vérifier cette propriété en simulation et de quantifier précisément les avantages ou les inconvénients éventuels d'une utilisation simultanée des deux mécanismes.

C'est ce que nous avons réalisé expérimentalement. Pour cela, nous avons décidé de rejouer une trace dans laquelle les flux « souris » sont échangés avec notre version de TCP sans, les flux « éléphants » avec le mécanisme TFRC et les flux intermédiaires (ni « souris » ni « éléphants ») utilisent TCP New Reno. Au préalable, nous aurons rejoué la trace en n'utilisant que TCP New Reno pour les différents flux (étape 1). Cette étape initiale nous servira de cadre de référence pour mesurer les avantages de la mise en concurrence (étape 2) des différents mécanismes présentés individuellement dans les paragraphes précédents.

En comparant l'évolution du débit au cours du temps (cf. Figure 14) à l'issue de la première étape et à l'issue de la deuxième on peut faire la remarque suivante : TFRC régule très fortement le trafic en comparaison de TCP New Reno. En effet, malgré un débit moyen dans les deux situations proche ($\text{Débit}(\text{trafic TCP New Reno} / \text{Etape 1}) = 82.335 \text{ ko}$ contre $\text{Débit}(\text{trafic sans SS, TCP New Reno et TFRC} / \text{Etape 1}) = 82.335 \text{ ko}$).

2) = **77.707 ko**) la variabilité du trafic dans le cas de l'utilisation unique de TCP est bien plus importante que lorsque les trois mécanismes sont utilisés en simultanément ($\sigma(\text{trafic TCP New Reno / Etape 1}) = \mathbf{157.959 ko}$ contre $\sigma(\text{trafic sans SS, TCP New Reno et TFRC / Etape 2}) = \mathbf{102.176 ko}$).

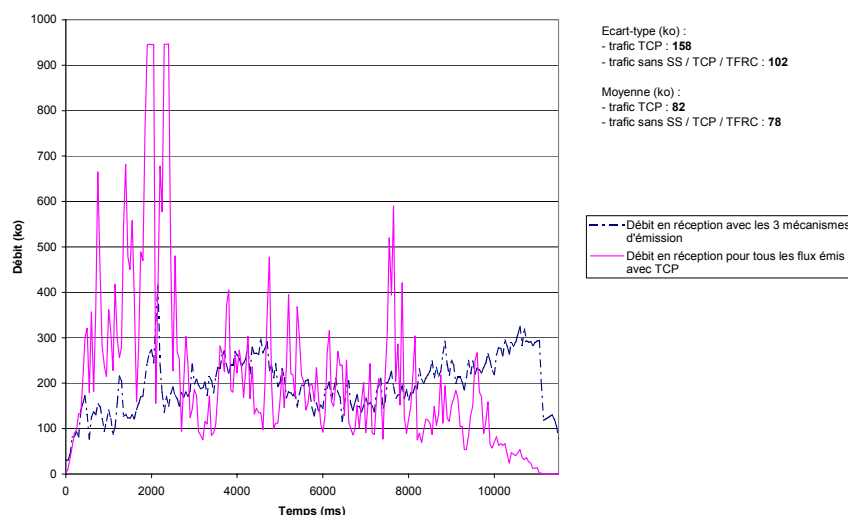


Figure 14 : Evolution du débit au cours du temps

Il est à noter que le débit moyen lors de l'étape 2 étant plus faible que lors de l'étape 1, ceci constitue une autre preuve que l'utilisation conjointe des mécanismes proposés optimise la QoS du réseau.

Ensuite, si l'on s'intéresse à la corrélation du trafic présent dans la situation 2 on observe une disparition quasi-complète de l'autocorrélation présente dans le trafic (cf. Figure 15 et Figure 16 respectivement pour les étapes 1 et 2).

Il en est de même pour la présence de dépendance longue mémoire ($H(\text{Situation 1}) = \mathbf{0,641}$ et $H(\text{Situation 2}) = \mathbf{0,194}$). Ainsi, le caractère complémentaire de ces deux mécanismes pour l'état du réseau est démontré. En effet, la corrélation et la dépendance sont plus faibles lorsque les deux mécanismes sont utilisés en parallèle que lorsqu'on utilise l'un ou l'autre des mécanismes de façon isolée.

De plus, l'utilisation conjointe des deux mécanismes permet de diminuer de façon conséquente les oscillations HF et BF présentes initialement dans le trafic (i.e. cas de l'utilisation de TCP New Reno de façon unique). Ce phénomène oscillatoire qui est identifié comme un danger réel pour la stabilité et la régularité du réseau sur de grandes échelles de temps, a pu être jugulé en proposant de nouveaux

mécanismes qui ont la particularité d'influer sur les paramètres les plus dégradants de la QoS : la corrélation inter-paquets et la LRD¹⁸.

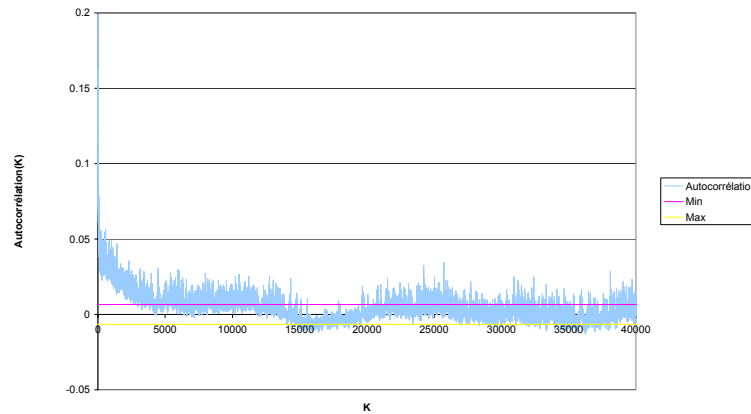


Figure 15 : Fonction d'autocorrélation des durées inter-paquets (étape 1)

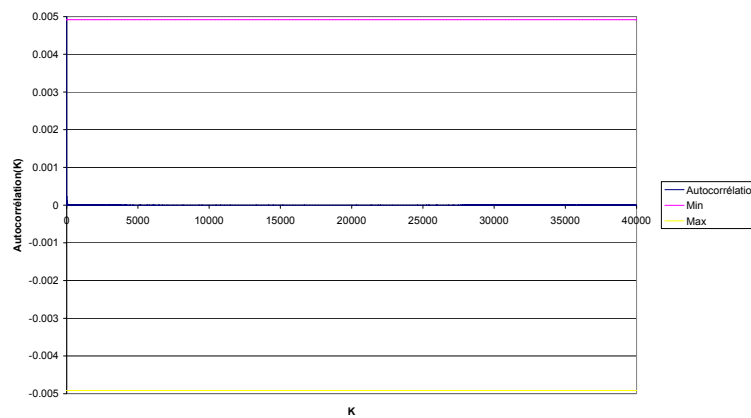


Figure 16 : Fonction d'autocorrélation des durées inter-paquets (étape 2)

¹⁸ A noter toutefois que les outils actuels d'évaluation de la LRD peuvent prendre pour de la LRD ce qui peut n'être en fait que de la non stationnarité. En ce qui nous concerne, une telle erreur ne pose aucun problème puisque dans les deux cas, LRD ou non stationnarité, ont un impact sur les oscillations et donc la QoS offerte par le réseau, et que notre objectif final est d'arriver à juguler le phénomène oscillatoire du trafic Internet.

5. Conclusion

Cet article a présenté et illustré l'importance de la métrologie et de l'analyse des traces de trafic dans l'ingénierie des réseaux de l'Internet. Il en a découlé la définition d'une nouvelle méthodologie qui intègre dans chacune de ces étapes des mesures sur le trafic pour caractériser, évaluer ou analyser le trafic, les protocoles et tous les composants réseaux et leur fonctionnement. L'utilisation de notre méthodologie est illustrée dans l'article au travers de quelques exemples de modification ou de remplacement du protocole TCP. L'objectif est de réguler le débit généré par chaque flux, en fonction de sa typologie, ce qui permet d'augmenter les performances du réseau et de stabiliser sa QoS. Cette méthodologie, en s'appuyant sur une phase initiale de caractérisation, permet d'apporter, dans un premier temps, les informations nécessaires pour l'analyse et la compréhension des phénomènes complexes de variabilité qui régissent le trafic. Ces nouvelles connaissances permettent, dans un deuxième temps, d'agir avec précision sur les mécanismes qui produisent un tel trafic. Ensuite, les traces issues de la métrologie sont utilisées afin d'améliorer le réalisme des simulations faites lors de la phase de conception et d'évaluation de nouveaux mécanismes protocolaires pour l'Internet. L'idée avancée ici est d'utiliser la caractérisation du trafic pour déterminer une topologie réaliste du réseau à mettre en place dans le simulateur. Ensuite, les sources de trafic vont rejouer du trafic réel, capturé par les outils de métrologie, de façon à confronter les nouvelles architectures protocolaires à évaluer à des conditions d'utilisation du réseau réalistes. Toutes les phases de cette méthodologie ont été décrites dans cet article, ainsi que les outils qui ont été développés pour ce faire.

L'originalité principale de cette méthodologie, au delà de l'idée d'utiliser de traces de métrologie, se situe au niveau de la caractérisation du trafic, et de l'approche proposée qui consiste à évaluer plus la dynamique du trafic que des valeurs moyennes sur de longues durées. En effet, les observations du trafic ont montré que des phénomènes d'oscillations se répétaient quels que soient les granularités d'observation du trafic et que ces oscillations entraînaient des baisses de performance globale du réseau et des difficultés à assurer des services stables. Une première approche, présentée dans cet article, a consisté à assimiler ces mécanismes oscillatoires à de la dépendance longue mémoire et à essayer de les quantifier par le paramètre de Hurst. Sur l'exemple présenté dans cet article, ce choix s'avère assez judicieux, même si l'étude de l'impact de la LRD sur la QoS reste un sujet à approfondir. En effet, les principes de création de la LRD dans le trafic doivent être analysés plus en détail, de même que leur impact sur la QoS. Lorsque tous les secrets de cette caractéristique du trafic auront été élucidés, des solutions pour améliorer, voire garantir, la QoS du trafic Internet pourront être proposées. Même si beaucoup reste à faire, la métrologie représente aujourd'hui un outil indispensable pour l'ingénierie et la recherche en réseau par toutes les informations qu'elle peut apporter sur le trafic et les conditions d'utilisation des réseaux. En particulier, les premières étapes de caractérisation, de modélisation et

d'analyse du trafic sont des étapes essentielles pour aller plus avant dans l'évolution des réseaux.

6. Références

- [Abry et al., 1998] P. Abry, D. Veitch, "Wavelet Analysis of Long Range Dependent Traffic", *Trans. Info. Theory*, Vol.44, No.1 pp.2-15, Jan 1998.
- [Allman et al., 2002] Allman M., Floyd S. and Partridge C., "Increasing TCP's initial window", Internet Draft, IETF, 12 pages, February 2002.
- [Beran et al., 1995] J. Beran, R. Sherman, M. S. Taqqu and W. Willinger, 'Long-range dependence in Variable-Bit-Rate video traffic', *IEEE Trans. Comm.*, Vol. 43, n° 2/3/4, pp. 1566-1579, 1995.
- [Black, 1992] U. Black, "TCP/IP and related protocols", McGraw-Hill, 1992.
- [Cao et al., 2001] J. Cao, W.S. Cleveland, D. Lin, D.X. Sun, « Internet traffic tends to Poisson and independent as the load increases », Bell Labs report, available at <http://cm.bell-labs.com/cm/ms/departments/sia/InternetTraffic/webpapers.html>
- [Crovella et al., 1997] M. Crovella and A. Bestavros, "Self-similarity in World Wide Web traffic : Evidence and possible causes", *IEEE/ACM Trans. on Networking*, Vol. 5, n° 6, pp. 835-846, Dec. 1997.
- [Crovella et al., 1998] Crovella, M. E., M. S. Taqqu, and A. Bestavros (1998), "Heavy-Tailed Probability Distributions in the World Wide Web," *In A Practical Guide To Heavy Tails*, chapter 1, Chapman & Hall, New York, pp. 3--26.
- [Erramilli et al., 1996] A. Erramilli, O. Narayan, W. Willinger, "Experimental queuing analysis with long range dependent packet traffic", *IEEE/ACM Transactions on Networking*, Vol. 4, No. 2, pp 209--223, 1996
- [Feldmann et al., 1998 a] A. Feldmann, A. C. Gilbert and W. Willinger, "Data networks as cascades : Explaining the multifractal nature of Internet WAN traffic", *Proc. of ACM SIGCOMM '98*, Aug. 1998.
- [Feldmann et al., 1998 b] A. Feldmann, A. C. Gilbert, W. Willinger and T. G. Kurtz, 'The changing nature of network traffic : Scaling phenomena', *Computer Communication Review*, Vol. 28, n° 2, April 1998.
- [Floyd et al., 1998] Floyd S. & Fall K., "Promoting the use of end-to-end congestion control in the Internet", In Proc. *IEEE ACM Transactions on Networking*, 14 pages, February 1998.
- [Floyd et al., 2000] Floyd S., Handley M., Padhye J. and Widmer J., "Equation-based congestion control for unicast applications", In Proc. *ACM SIGCOMM*, 14 pages, 2000.
- [Floyd et al., 2001 a] Floyd S., Kohler E. and Padhye J., « Profile for DCP congestion control ID 3 : TFRC congestion control », Internet Draft, IETF, 8 pages, November 2001.
- [Floyd et al., 2001 b] Floyd S. and Paxson V., "Difficulties in Simulating the Internet" *IEEE/ACM Transactions on Networking*, Vol.9, No.4, pp. 392-403, August 2001

- [Floyd et al., 2002] S. Floyd, E. Kohler, "Internet research needs better models", first workshop on hot topics in networks (HotNets-I), Princetown, New Jersey, USA, 28-29 October 2002
- [Garcia et al., 2001] Garcia J.M., Gauchard D., Brun O., Bacquet P., Sexton J., Lawless E., « Modélisation différentielle du trafic et simulation hybride distribuée », *Calculateurs parallèles*, Vol. 18, No. 3, 2001
- [Huang et al., 1999] X.W.Huang, R. Sharma, and S. Keshav, "The ENTRAPID Protocol Development Environment", in *Proc. Infocom '99*, March 1999.
- [Heyman, 1998] D. Heyman, "Some issues in performance modeling of data teletraffic", *Performance Evaluation*, Vol. 34, pp. 227-247, 1998.
- [Joo et al., 2001] Y. Joo, V. Ribeiro, A. Feldmann, A. Gilbert, W. Willinger, "TCP/IP traffic dynamics and network performance: A lesson in workload modelling, flow control, and trace-driven simulations", *Computer Communication Review*, April 2001
- [Leland et al., 1993] Leland W., Taqqu M., Willinger W., Wilson D., "On the self-similar nature of Ethernet traffic", *ACM SIGCOM*, September 1993
- [Leland et al., 1994] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the self-similar nature of Ethernet traffic (extended version)," *IEEE/ACM Transactions on Networking*, vol. 2, pp. 1--15, Feb 1994.
- [Paxson et al., 1995] V. Paxson, S. Floyd, "Wide-Area Traffic: The Failure of Poisson Modeling", *IEEE/ACM Trans. on Networking*, vol. 3, no. 3, June 1995
- [PlanetLab] Site web du projet PlanetLab : <http://www.planet-lab.org/>.
- [Roberts, 2000] J. Roberts, "Engineering for Quality of Service", In "Self-similar network traffic and performance evaluation", edited by K. Park and W. Willinger, J. Wiley & Sons, 2000.
- [Thompson et al., 1997] K. Thompson, G. Miller and M. Wilder, "Wide-area internet traffic patterns and characteristics", *IEEE Network*, Vol. 11, n° 6, Nov./Dec. 1997.
- [Tran Gia et al., 2000] Ph. Tran-Gia and N. Vicari (editors), 'Impacts of new services on the architecture and performance of broadband networks (Final report of COST 257)', Chapter on 'Traffic measurement and data analysis', 2000.
- [Veres, 2000] VERES, A., « The Chaotic Nature of TCP Congestion Control », *In proceedings of the INFOCOM conference*, March 2000.
- [White et al., 2002] White, Lepreau, Stoller, Ricci, Guruprasad, Newbold, Hibler, Barb, and Joglekar, "An Integrated Experimental Environment for Distributed Systems and Networks", appeared at OSDI 2002, December 2002.
- [Zhang et al., 2001] Y. Zhang, N. Duffield, V. Paxson, and S. Shenker, "On the Constancy of Internet Path Properties", *Proc. ACM SIGCOMM Internet Measurement Workshop (IMW'2001)*, San Francisco, California, USA, November 2001.