Analyse statistique du trafic sur le serveur de référence du *World Wide Web*

Rapport de stage
DEA Informatique et Recherche Opérationnelle (Université Paris 6)
effectué à l'INRIA de Sophia Antipolis
du 22 Mai au 22 Septembre 1995

Nicolas NICLAUSSE

Sous la direction de
Zhen LIU et Philippe NAIN

Je remercie mes directeurs de stage Zhen LIU et Philippe NAIN pour leurs conseils et la confiance qu'ils m'ont accordée. Je remercie également toute l'équipe MISTRAL, ainsi que les stagiaires, pour la bonne ambiance qui régnait au sein du projet durant toute la durée du stage.

Table des matières

Table des figures

Chapitre 1

Introduction

1.1 Présentation du stage

On s'est intéressé à l'étude du trafic du World Wide Web (WWW ou Web), plus particulièrement au serveur de référence (*Consortium WWW* ou W^3C) situé au MIT (voir figure 1.3). Ce serveur héberge entre autres les spécifications techniques du Web, une bibliothèque de logiciels dédiés au WWW et l'annuaire de tous les serveurs Web dans le monde. Avec la croissance exponentielle du trafic sur Internet, une stratégie adoptée afin d'augmenter les performances des serveurs est celle de la réplication, c'est-à-dire la mise en place de serveurs miroirs. L'INRIA de Sophia Antipolis est particulièrement impliquée dans ces problèmes de saturations dûs à l'augmentation du trafic dans la mesure où un miroir du serveur de référence sera installé dans ses locaux au cours du mois de septembre 1995. Dans ce contexte, il est important de bien *dimensionner* le serveur (capacité CPU, disques) et les liens de communication. Afin de résoudre ces problèmes, on a habituellement recours à un formalisme de réseaux de files d'attentes. Ce formalisme à été appliqué avec succès pour l'évaluation de performance et le dimensionnement de réseaux de télécommunication. La première étape de cette approche est de trouver un modèle mathématique ayant les propriétés statistiques observées sur les données réelles du trafic. Dans cette optique, le but du stage était de mettre en évidence la présence éventuelle de corrélations sur le long terme dans le trafic généré par les requêtes d'accès à ce serveur. Il est en effet apparu dans divers études sur le trafic dans les réseaux d'ordinateurs (réseaux locaux comme Ethernet [6] ou bien Internet[17]) que celui-ci était fortement corrélé, contrairement à d'autres type de trafic déjà étudiés (en particulier du type réseau téléphonique). De plus, des résultats récents que nous évoquerons plus loin montrent que les performances de systèmes soumis à des modèles de trafic ayant cette propriété de mémoire longue sont sensiblement différentes de celles connues pour les modèles utilisés jusqu'ici. Il est donc important de vérifier si cette propriété est présente sur le trafic du Web. Dans un premier temps, on décrira les outils théoriques permettant de détecter la présence de corrélations et de quantifier celles-ci. On introduira également différents processus stochastiques ayant ces caractéristiques, et susceptibles de servir de modèle au trafic observé. Dans un deuxième temps, on décrira les différentes méthodes statistiques permettant de mesurer en pratique la présence de ce que l'on appelle mémoire longue. Ensuite, on détaillera les résultats obtenus à partir des données collectés sur le serveur Web du W^3C.

1.2 L'INRIA

L'Institut national de recherche en informatique et en automatique est un EPST (établissement public à caractère scientifique et technologique) placé sous la tutelle du ministre chargé de l'enseignement supérieur et de la recherche, conjointement avec le ministre chargé des technologies de l'information et de la poste.

Mille cinq cents personnes travaillent à l'institut, parmi lesquelles un millier de scientifiques : chercheurs et ingénieurs permanents, chercheurs extérieurs de l'industrie, chercheurs de laboratoires de recherche publics, chercheurs étrangers invités, étudiants en thèse, stagiaires.

L'INRIA est actuellement composé de 5 unités recherches:

- Rocquencourt
- Sophia-Antipolis
- Lorraine
- Rennes
- Rhônes-Alpes

Les activités de l'INRIA comprennent la réalisation de systèmes expérimentaux, la recherche fondamentale et appliquée, le transfert de technologie, l'organisation d'échanges scientifiques internationaux et la diffusion des connaissances et du savoir-faire.

Ces activités associent informaticiens, automaticiens, mathématiciens dans le cadre de projets de recherche regroupés dans six programmes :

- Programme 1 : Architectures parallèles, bases de données, réseaux et systèmes
- Programme 2 : Calcul symbolique, programmation et génie logiciel
- Programme 3 : Intelligence artificielle, systèmes cognitifs et interaction homme-machine
- Programme 4 : Robotique, image et vision
- Programme 5 : Traitement du signal, automatique et productique
- Programme 6 : Calcul scientifique, modélisation et logiciels numériques

L'unité de recherche INRIA Sophia Antipolis est quand à elle située au coeur de la plus grande technopole européenne, et représente un potentiel de 300 personnes.

Ses grands domaines d'activité sont :

- génie logiciel, langages et calcul symbolique,
- robotique et vision par ordinateur,
- mécanique des fluides numérique, hypersonique, combustion,
- réseaux, systèmes à événements discrets.

Le stage s'est effectué au sein du projet MISTRAL (Modélisation en informatique et systèmes de télécommunication : recherche et applications logicielles) qui fait partie des projets du programme 1.

MISTRAL se consacre à la construction et à l'analyse de modèles mathématiques pour l'évaluation des performances de systèmes aléatoires à événements discrets. La modélisation des systèmes informatiques et télématiques des systèmes de production et celle des problèmes de trafic dans les réseaux de communication donnent actuellement naissance à une nouvelle théorie des systèmes à événements discrets. Plusieurs formalismes sont utilisables dans le cadre de cette théorie. Les travaux du projet se concentrent sur deux d'entre eux : les réseaux de files d'attente et les réseaux de Pétri.

Axes de recherche:

- Élaboration de modèles mathématiques (le plus souvent stochastiques) puis leur résolution analytique ou numérique.
- Distribution de la simulation traditionnelle à événements discrets et développement de nouvelles techniques comme l'analyse de sensibilité et la simulation par équations récursives. Des prototypes sont en cours de développement sur réseau de Transputers et sur architecture SIMD (Connection Machine). Ces derniers sont à leur tour générateurs de problèmes alimentant l'activité théorique du projet.
- À partir de problèmes concrets, de méthodes et d'outils de portée générale, proposition permettant d'obtenir des résultats quantitatifs sur le rendement, la stabilité ou le contrôle de telle ou telle structure.

1.3 Le World Wide Web

Le World Wide Web (ou plus simplement WWW, Web ou encore W^3) est un ensemble de documents hypermédias (hypertextes notamment) reliés entre eux sur le réseaux *Internet*. Le World Wide Web a été initialement développé par Tim Berners-Lee et Robert Cailliau dans les laboratoires du CERN, à Genève, afin de donner aux physiciens des particules du CERN une infrastructure leur permettant de partager l'information. Comme ceux-ci étaient situés dans différentes organisations et utilisait une grande variété de systèmes informatiques et de logiciels (notamment traitement de texte), le World Wide Web a été développé en utilisant un architecture client-serveur, qui assure une portabilité multi-plateforme.

Plus précisément, le Web englobe trois composantes:

- **HTML** langage de description de documents hypertexte, multimédia (images, sons, vidéo, ...), multi-protocole;
- **HTTP** protocole permettant la récupération de documents HTML localisés sur des serveurs;
- **WWW** réseau mondial de serveurs délivrant des documents au format HTML;

L'avantage du WWW sur les autres applications utilisés sur Internet (notamment ftp) est sa grande facilité d'utilisation. C'est pourquoi depuis le mois d'avril 1995, le WWW est devenu l'application générant le plus de trafic sur Internet (environ 25% du trafic total).

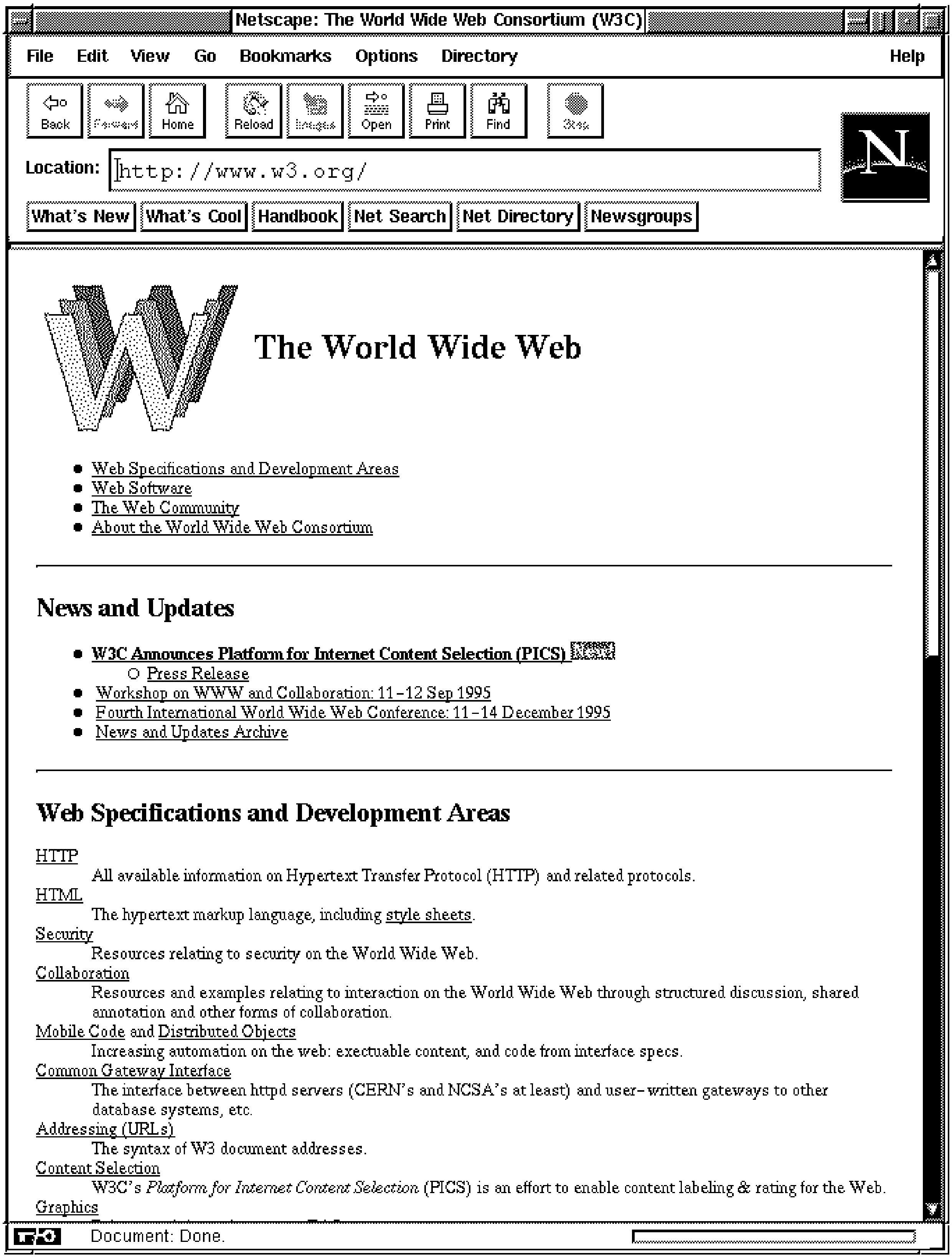

FIG. 1.1 – *Serveur de référence du World Wide Web (http://www.w3.org)*

1.4 Le Consortium WWW

Le W^3C est un organisme dirigé conjointement par le MIT/LCS, laboratoire de recherche informatique du MIT (Massachusetts Institute of Technology), et l'INRIA.

Les objectifs du Consortium sont les suivants :

- promouvoir l'avancement de la technologie dans le domaine des réseaux, du graphique et des interfaces utilisateurs, en dotant le World Wide Web d'une infrastructure complète d'information,
- encourager l'industrie à adopter un ensemble de protocoles pour le World Wide Web.

Le MIT/LCS et l'INRIA devront jouer un rôle moteur et impartial dans la conception de l'architecture du Web, son ingénierie et sa gestion, en visant :

- la conception d'un ensemble de protocoles World Wide Web,
- leur promotion dans le monde entier,
- l'incitation du milieu industriel à créer des produits conformes à ces protocoles,
- le développement d'un code de référence disponible publiquement.

Tim Berners-Lee, concepteur du Web au CERN (Suisse), qui a rejoint le Laboratoire de Science Informatique du MIT, occupe à présent les fonctions de directeur du consortium. Un directeur adjoint sera nommé par l'INRIA et conduira les travaux menés en Europe. Un comité consultatif d'experts est dès à présent chargé d'étudier les propositions de spécifications ainsi que les plans annuels, d'évaluer les développements menés, et de proposer de nouveaux axes de travail.

Participation au Consortium :

Le Consortium est autofinancé par les droits d'adhésion des participants. L'adhésion au Consortium est ouverte à toute entreprise et organisme.

Il existe deux types de participations : les membres à part entière (Full Members) et les membres affiliés (Affiliate Members). Le droit d'adhésion pour un membre à part entière s'élève à 150 000 $, et pour un membre affilié à 15 000 $. Les organismes dont le chiffre d'affaires s'élève à plus de 50 millions de $ par an doivent s'inscrire comme membre à part entière. Les autres organismes, y compris ceux à vocation éducative, peuvent s'inscrire comme membre à part entière ou membre affilié. Les droits d'adhésion couvrent les trois premières années du Consortium et sont payables en versements annuels d'un montant identique.

Le Consortium sera opérationnel jusqu'en 1997. La décision de prorogation du Consortium sera alors prise par les membres.

Activités du consortium:

De nombreux sujets seront traités, notamment dans les domaines suivants :

- les protocoles concernant la sécurité dans la communication entre deux parties, ainsi que dans les systèmes de paiement entre parties,
- les protocoles sur la reproduction et sur l'utilisation des caches,

- les protocoles portant sur les réseaux bas débits et l'informatique nomade,
- l'intégration de canaux temps réel,
- le travail collaboratif, la représentation des connaissances, et les technologies agents,
- les versions 3 et 4 HTML,
- le Web et SGML,
- la définition de feuilles de style,
- la labellisation du contenu et la propriété intellectuelle,
- l'internationalisation.

Chapitre 2

Modèles mathématiques

2.1 Processus stationnaires

Le cadre mathématique de cette étude est celui des processus stochastiques à temps discret (on évoquera également le cas continu); plus particulièrement, on s'intéresse aux processus stochastiques *stationnaires* au sens large, c'est-à-dire aux processus à covariance stationnaire. Rappelons tout d'abord quelques définitions:

Définition 1 *Un processus stochastique* $X = \{X_t\}$ *est* **stationnaire** *si* $\forall\ n \in \mathbb{N},\ \forall h,\ \forall\ t_0 < t_1 < \ldots < t_n$*, on a:*

$$(X_{t_0}, \ldots, X_{t_n}) =_{\mathcal{L}} (X_{t_{0+h}}, \ldots, X_{t_{n+h}})$$

où $=_{\mathcal{L}}$ désigne l'égalité en loi.

Définition 2 *Un processus stochastique* $X = \{X_t\}$ *est à* **covariance stationnaire** *si la moyenne et la variance existent et sont indépendantes du temps, et si l'auto-covariance est indépendante par translation dans le temps:*

1. $\mathbb{E}X_t = \mu < \infty,\ \forall t \in \mathbb{N}$
2. $\mathbb{E}(X_t - \mu)^2 = \sigma^2 < \infty,\ \forall t \in \mathbb{N}$
3. $\mathbb{E}(X_t - \mu)(X_{t+k} - \mu) = \gamma_k,\ \forall t \in \mathbb{N}, k \in \mathbb{N}$

Un processus stationnaire est évidemment à covariance stationnaire (si sa moyenne et sa variance sont finies).

Rappelons également le définition de la fonction d'auto-corrélation ρ

$$\rho(k) = \frac{\mathrm{cov}(X_t, X_{t+k})}{\sqrt{\mathrm{var}(X_{t+k})\mathrm{var}(X_t)}} \tag{2.1}$$

pour un processus à covariance stationnaire, l'auto-corrélation est donc de la forme suivante

$$\rho(k) = \frac{\gamma(k)}{\sigma}$$

La densité spectrale S est la transformée de Fourier de la fonction d'auto-corrélation, c'est-à-dire:

$$S(w) = \frac{1}{2\pi} \sum_{k=-\infty}^{\infty} e^{-\imath wk} \rho(k)$$

2.2 Processus à mémoire longue

On reprend les définitions introduites par COX [2].

Étant donné une processus $X = \{X_t\}$, introduisons le processus des moyennes issu de X, $X_t^{(m)} = (X_{tm-m+1} + \cdots + X_{tm})/m$, qui nous sera très utile pour décrire les propriétés de mémoire longue. On peut remarquer que si X_t est à covariance stationnaire, alors $X_t^{(m)}$ l'est aussi pour tout m.

Définition 3 *Soit $X = \{X_t\}$ un processus stochastique (à covariance) stationnaire à temps discret, on dit que X_t est à* **mémoire longue** *s'il vérifie les propriétés suivantes:*

1. $\sum_{t=0}^{\infty} \rho(t) = \infty$
2. *La densité spectrale S est singulière à l'origine*
3. $m\mathrm{var}X^{(m)} \to \infty$ *quand* $m \to \infty$.

Les deux premières propriétés sont équivalentes, mais la troisième n'est équivalentes aux deux premières que sous certaines conditions (COX [2]).

Ces propriétés ne sont pas vérifiées par les processus généralement considéré pour la modélisation de trafic: les processus de Markov (par exemple, le processus de Poisson et le processus de Poisson doublement stochastique) vérifient les propriétés suivantes (caractéristique d'un processus à mémoire courte):

1. $\sum_{t=0}^{\infty} \rho(t) < \infty$
2. La densité spectrale est finie à l'origine
3. $m\mathrm{var}X^{(m)}$ est fini quand $m \to \infty$.

En pratique, on adoptera plutôt la définition suivante un peu plus restrictive: Les processus vérifiant les propriétés équivalentes suivantes sont à mémoire longue de paramètre $\beta \in]0, 1[$:

1. $\rho(h) \sim_{h\to\infty} C_1 h^{-\beta}$
2. $S(\omega) \sim_{\omega\to 0} C_2 \omega^{-(1-\beta)}$
3. $\mathrm{var}X^{(m)} \sim_{m\to\infty} C_3 m^{-\beta}$

où les C_i sont des constantes.

A partir de ces propriétés, on va pouvoir définir des tests assez simples permettant d'estimer le paramètre de mémoire longue.

Une classe de processus à mémoire longue est la classe des processus auto-similaires de paramètre $H \in]\frac{1}{2}, 1[$.

2.3 Processus auto-similaires

Définition 4 *Un processus $X = \{X_t\}, t \in \mathbb{N}$ est* **exactement auto-similaire** *de paramètre $H \in]0,1[$ si le processus $\tilde{X}^{(m)}$ défini par $\tilde{X}_t^{(m)} = (X_{tm-m+1} + \cdots + X_{tm})/m^H$ à la même loi que X pour tout m.*

Intuitivement, le passage de X à $\tilde{X}^{(m)}$ correspond à un changement d'échelle. Un processus auto-similaire à donc la même loi, quelle que soit l'échelle de temps considérée (modulo un coefficient dépendant de l'échelle et du paramètre d'auto-similarité H). Le paramètre H est souvent appelé paramètre de Hurst.

Comme on s'intéresse uniquement à la structure de corrélation, on définit également la notion d'auto-similarité au second ordre:

Définition 5 *Un processus $X = \{X_t\}, t \in \mathbb{N}$ est* **exactement auto-similaire (au second ordre)** *de paramètre $H \in]0,1[$ si le processus $X^{(m)}$ a la même fonction d'auto-corrélation que X.*

On peut montrer (COX [2]) que cette dernière définition implique:

$$\begin{cases} \rho^{(m)}(k) & = & \frac{1}{2}\delta^2(k^{2H}) \\ \mathrm{var}(X^{(m)}) & = & \sigma^2 m^{2H-2} \end{cases}$$

où δ est l'opérateur de différence centrale appliqué à une fonction: $\delta^2(f(k)) = (f(k+1) - 2f(k) + f(k-1))$

En utilisant l'équivalence asymptotique de δ et de l'opérateur de dérivation, un processus exactement auto-similaire vérifie la propriété suivante:

$$\forall m,\ \rho^{(m)}(k) \sim_{k\to\infty} H(2H-1)k^{-(2-2H)} \tag{2.2}$$

Par conséquent, un processus auto-similaire possède de la mémoire longue si $H > 1/2$ (en prenant $\beta = 2 - 2H$)

Ceci conduit à la définition suivante:

Définition 6 *Un processus $\{X_t\}, t \in \mathbb{N}$ est* **asymptotiquement auto-similaire (au second ordre)** *de paramètre H si*

$$\lim_{m\to\infty} \rho^{(m)}(k) = \frac{1}{2}\delta^2(k^{2H}). \tag{2.3}$$

On a alors $\rho^{(m)}(k) \sim_{k\to\infty} \rho(k)$ pour m assez grand.

Pour $H > \frac{1}{2}$, un processus asymptotiquement auto-similaire est à mémoire longue.

Pour avoir une idée de l'influence du paramètre H, se référer au graphique 2.1 représentant une simulation d'un processus auto-similaire, le bruit Gaussien fractionnaire (cf. équation 3.2), pour différentes valeurs de H. On constate que plus le paramètre de Hurst est grand (donc plus les corrélations sont fortes), plus le processus à tendance à s'écarter de la moyenne, *i.e.* on observe un phénomène de non-stationnarité locale. Par conséquent, un processus non stationnaire (ayant une moyenne variant périodiquement ou aléatoirement, ou bien ayant

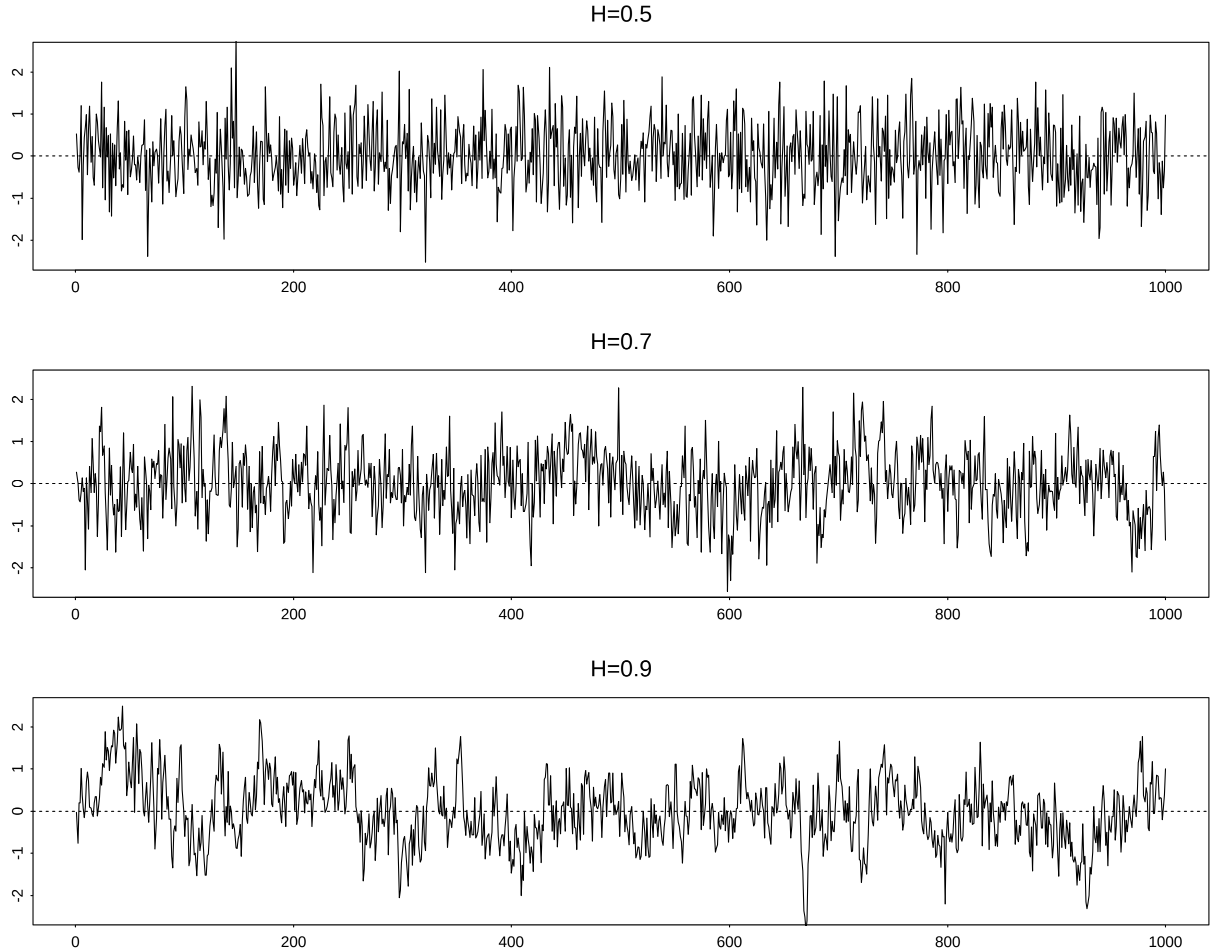

FIG. 2.1 – *Simulations du bruit Gaussien fractionnaire pour différentes valeurs de H*

une tendance linéaire) peut reproduire l'effet de Hurst, c'est-à-dire, donner une valeur de H strictement supérieure à $0,5$. Il faut donc en pratique faire bien attention à ce que les échantillons étudiés soient bien stationnaire.

Évoquons maintenant le cas continu:

Définition 7 *Un processus stochastique $X = \{X_t\}$ (à temps continu) est* **auto-similaire** *de paramètre $H > 0$ si $\forall c > 0$, $\{X_{ct}\}$ et $\{c^H X_t\}$ sont égaux en loi.*

On considère maintenant le processus des accroissements d'un p.s. à temps continu auto-similaire, $Z(n) = X(n) - X(n-1)$, $(n \in \mathbb{N}^*)$. Si Z est stationnaire alors l'auto-corrélation de Z est la suivante :

$$\rho(k) = \frac{1}{2}\delta^2(k^{2H}) \tag{2.4}$$

i.e. Z est exactement auto-similaire de paramètre H.

Une conséquence immédiate des propriétés précédentes est que si $H > 1/2$, le processus est à mémoire longue.

Chapitre 3

Exemples de processus a mémoire longue

Considérons le processus suivant (appelé *mouvement Brownien fractionnaire*) introduit par B. MANDELBROT [11].

3.1 Mouvement Brownien Fractionnaire

Le *mouvement Brownien fractionnaire* normalisé de paramètre de Hurst $H \in]0,1[$ est un processus stochastique à temps continu $X = \{X_t\}_{t\geq 0}$ défini par :

$$X_H(t) = \frac{1}{\Gamma(H+\frac{1}{2})}\left\{\int_{-\infty}^{0}[(t-s)^{H-\frac{1}{2}} - (-s)^{H-\frac{1}{2}}]dB(s) + \int_0^t (t-s)^{H-1/2}dB(s)\right\} \tag{3.1}$$

où $\{B_t\}$ est le mouvement Brownien standard.

On peut également le caractériser par les propriétés suivante:

- X_t à des accroissements stationnaires;
- $X_0 = 0$ et $\forall t \in \mathbb{R},\ \mathbb{E}X_t = 0$
- $\forall t \in \mathbb{R},\ \mathbb{E}X_t^2 = |t|^{2H}$
- X_t à des trajectoires continues
- X_t est Gaussien

Le mBf est un processus auto-similaire, et pour $H = \frac{1}{2}$, c'est le mouvement Brownien usuel. On définit maintenant le *bruit Gaussien fractionnaire* $Z(n)$ comme le processus (discret) des accroissements d'une unité du mBf:

$$Z(n) = X_n - X_{n-1}, \quad n = 1, 2, \ldots \tag{3.2}$$

D'après la propriété générale sur les accroissements d'un processus auto-similaire rappelée précédemment (équation 2.4), le bruit Gaussien fractionnaire est un processus exactement auto-similaire au second ordre, et donc en particulier pour $H > 1/2$, c'est un processus à mémoire longue. L'intérêt de ce modèle vient du fait de sa simplicité, en ce sens qu'il suffit de seulement 3 paramètres pour le caractériser: moyenne, variance et coefficient d'auto-similarité. Les figures 3.1 et 3.2 donnent un exemple de trajectoires du mBf et du bruit Gaussien fractionnaire pour $H = 0.7$.

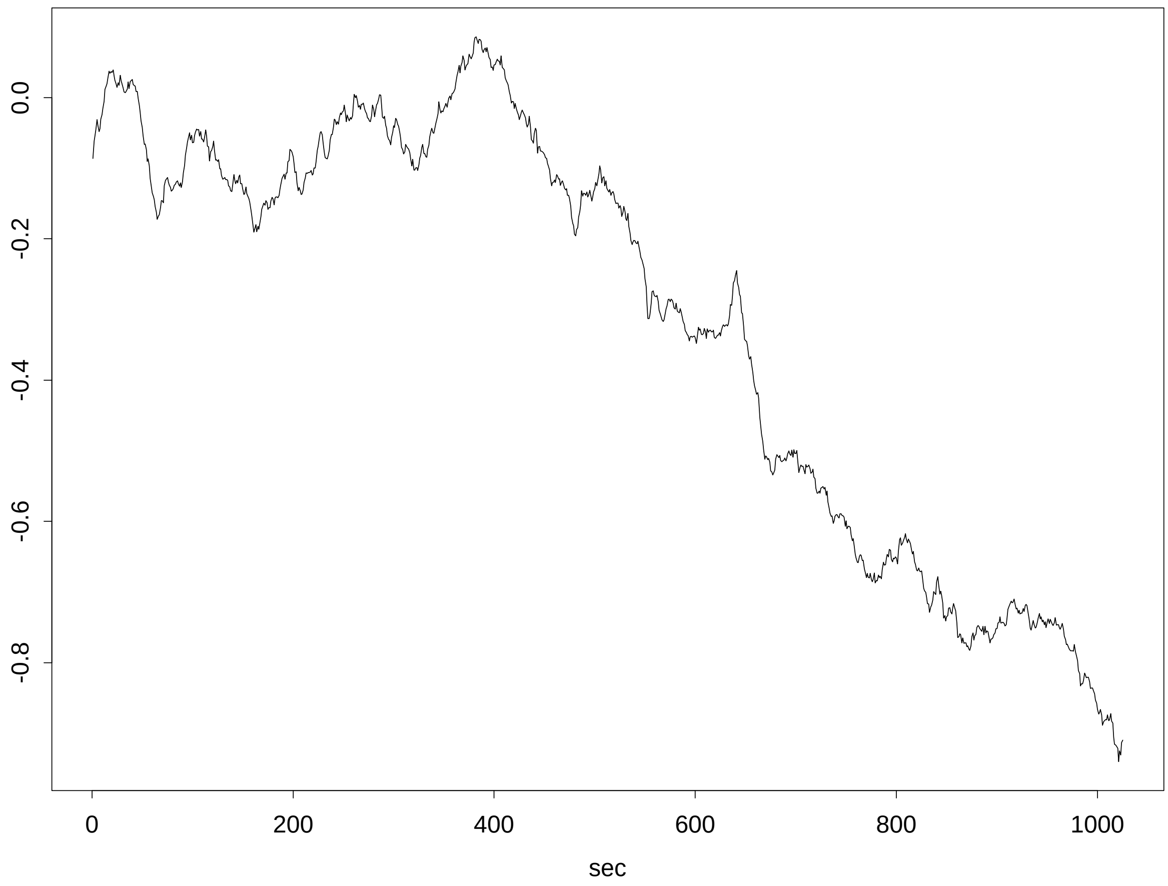

FIG. 3.1 – *Mouvement Brownien Fractionnaire, H=0.7*

Un autre processus, cette fois-ci à temps continu, construit construit a partir du MBF normalisé susceptible de modéliser un trafic auto-similaire est analysé par NORROS dans [13] et [14]:

$$A_t = mt + \sqrt{am}X_t, \quad t \in]-\infty, \infty[\tag{3.3}$$

En reprenant la terminologie introduite dans [14], on appellera le processus A_t *trafic Brownien fractionnaire*. Les paramètres s'interprettent de la façon suivante: m est le taux d'arrivée moyen, a est un coefficient de variance.

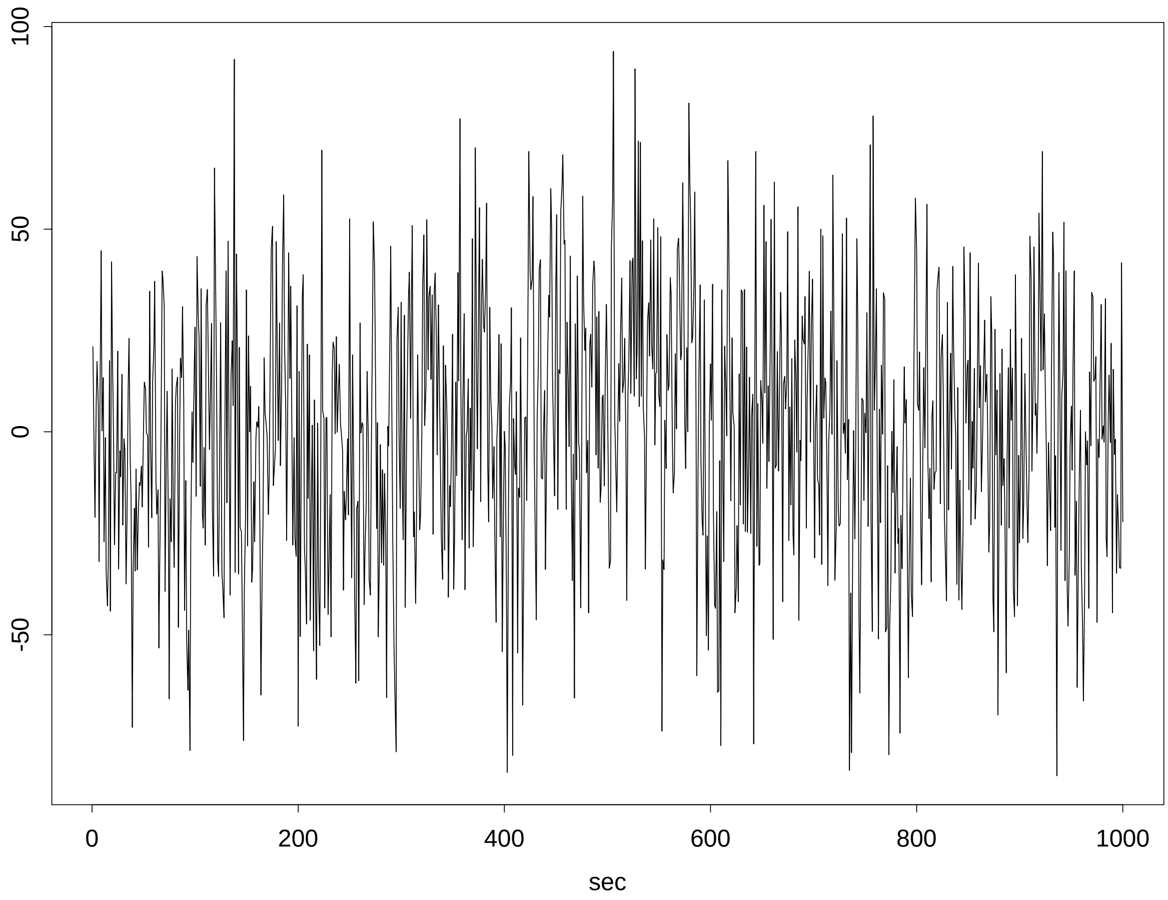

FIG. 3.2 – *Bruit Gaussien Fractionnaire, H=0.7*

On suppose maintenant que ce trafic arrive dans une file d'attente de capacité infinie ayant un serveur de capacité fixe $C > m$. L'occupation de la file à l'instant t vaut $X_t = \sup_{s \leq t}(A_t - A_s - C(t-s))$. Dans ce cas là, on a le résultat suivant pour la queue de distribution de X:

$$\mathbb{P}(X > x) \sim \exp\left(-\frac{(C-m)^2 H}{2H^{2H}(1-H)^{2H} am} x^{2-2H}\right) \quad (x \to \infty) \tag{3.4}$$

Cette probabilité est du type Weibull, alors que dans le cas où le trafic d'entrée est Poissonnien, la distribution est exponentielle (cf. [3]). Ceci montre bien les modèles auto-similaires offrent des caractéristiques différentes (et ce d'autant plus que H est proche de 1) des modèles classiques. Ceci montre que pour les problèmes de dimensionnement (où ces probabilités de dépassement de capacité $\mathbb{P}(X > x)$ jouent un rôle important), on doit tenir compte de la présence de mémoire longue (et donc du paramètre de Hurst).

3.2 La file $M/G/\infty$

Introduisons d'abord la notion de loi « heavy-tailed »:

Définition 8 *On dit d'une loi de probabilité qu'elle est* **heavy-tailed** *si elle vérifie*

$$\mathbb{P}(X > x) \sim_{x \to \infty} L(x) x^{-\beta} \tag{3.5}$$

*où L est une fonction variant lentement à l'infini (*i.e. $\lim_{t\to\infty} L(xt)/L(t) = 1$*), et $\beta \in]1,2[$.*

Cette définition implique une espérance finie, mais aussi une variance infinie. L'exemple le plus classique est la loi de Pareto.

Définition 9 *Densité d'une loi de Pareto de paramètre (α, β)*

$$f(x) = \beta \alpha^{\beta} x^{-\beta-1}, \; x \geq a \tag{3.6}$$

Une v.a. de loi de Pareto est « heavy-tailed » quand $1 < \beta < 2$

On considère maintenant la file d'attente $M/G/\infty$, c'est-à-dire une file dont le processus d'arrivée est poissonnien, les temps de services sont indépendants identiquements distribués, et le nombre de serveur infini. Si on suppose que la loi des services est « heavy-tailed », alors le processus X_t du nombre de clients dans la file à l'instant t est asymptotiquement auto-similaire (au second ordre) de paramètre $H = (3-\beta)/2$ (Cox [2]).

Signalons un résultat récent dû à Parulekar et Makowski [15]. En reprenant le même modèle que précédemment, c'est-à-dire un trafic arrivant dans une file d'attente infinie ayant un serveur de capacité fixe C, on obtient les approximations suivantes pour le bruit Gaussien fractionnaire et le processus du nombre de clients dans une file $M/G/\infty$:

- *Trafic d'entrée:* Bruit Gaussien fractionnaire

  $$\mathbb{P}(q_\infty > x) = \exp(f(H, C, m) x^{2(1-H)}) \; (x \to \infty) \tag{3.7}$$

  q_∞ désigne l'occupation de la file à l'état stationnaire.

  On constate que, comme pour le trafic Brownien stationnaire (3.4), la distribution est du type Weibull.

– *Trafic d'entrée:* égal au processus du nombre de clients dans une file $M/G/\infty$:

$$\mathbb{P}(q_\infty > x) \geq x^{-(2-2H)(C-m)} \; (x \to \infty) \tag{3.8}$$

On constate que la queue de distribution est différente pour ces deux trafic à mémoire longue. Par conséquent quand on s'intéresse aux probabilités asymptotiques de « dépassement de capacité », le paramètre de Hurst seul n'est pas suffisant pour caractériser les propriétés de mémoire longue du trafic.

3.3 Superposition de sources ON/OFF

On considère maintenant un processus composé de la superposition d'un grand nombre de sources, qui émettent des données pendant certaines périodes (dite périodes ON) à un taux constant, et n'émettent rien pendant le reste du temps (les périodes dites OFF).

On suppose que les lois des durées des périodes sont indépendantes et identiquement distribuée.

Si on fait l'hypothèse supplémentaires que la loi des durées des périodes ON est « heavy-tailed » alors le processus composé de la superposition d'un nombre m de sources ON/OFF indépendantes est asymptotiquement auto-similaire quand m tend vers l'infini. (ce résultat à été démontré dans [9])

Chapitre 4

Statistique des processus auto-similaires

4.1 Introduction

L'objectif de cette étude est de détecter le présence de mémoire longue dans le trafic du Web. D'après les définitions données dans le chapitre 2, une méthode directe est d'estimer le paramètre de Hurst H. Si celui-ci est strictement supérieur à 0.5, alors on peut dire que le trafic exhibe de la mémoire longue; si la valeur estimée est proche de 0.5, alors cette hypothèse est fausse (*i.e.* on est en présence de mémoire courte). La suite de ce chapitre décrit les méthodes d'estimation du paramètre de Hurst qu'on a utilisé en pratique.

4.2 Statistique R/S

Étant donné un processus $\{X_t\}_{t\geq 0}$ stationnaire, on défini « rescaled adjusted range » de la façon suivante:

Posons

$$R(n) = \max(0, W_1, \dots, W_n) - \min(0, W_1, \dots, W_n) \tag{4.1}$$

où $W_k = (X_1 + \dots + X_k) - k\bar{X}(n)$

où $\bar{X}(n)$ (resp. $S(n)$) est l'espérance (resp. la variance) du processus considéré jusqu'à l'ordre n.

Si X_k est un processus a mémoire « courte », alors $\mathbb{E}R(n)/S(n)$ est asymptotiquement proportionnel à $n^{1/2}$. Cependant, pour les processus auto-similaires, on a $\mathbb{E}R(n)/S(n) \sim n^H$ (voir [10] pour une démonstration).

Afin d'estimer le paramètre H décrit ci-dessus, une méthode empirique basée sur une approche graphique est fréquemment utilisée (cf. [10] par exemple). Étant donné un échantillons de N observations, on le subdivise en K sous-échantillons distincts, et on estime le *rescaled adjusted range* $R(t_i, n)/S(t_i, n)$ pour tous les « points de départs » t_i. Les t_i sont définis de la façon suivante: $t_1 = 1, t_2 = N/K + 1, \dots, t_i = iN/K + 1$ pour tout i vérifiant $(t_i - 1) + n \leq N$. $R(t_i, n)$ est défini comme dans 4.1 en remplaçant W_k par $W_{t_i+k} - W_{t_i}$ et

$S(t_i, n)$ est la variance empirique de $\{X_{t_i}, \ldots, X_{t_i+n}\}$. Par conséquent, on a plusieurs échantillon de R/S pour chaque valeur de n (autant que K pour les n « petits», et une seule pour les n proches de N). L'estimation de H se fait en traçant $\log(R(t_i, n)/S(t_i, n))$ en fonction de $\log(n)$. Le graphique obtenu (le *rescaled adjusted range plot* ou *pox diagram* de R/S) est dans les bons cas proche d'une droite de pente H; il suffit donc d'évaluer cette pente pour obtenir une estimation de H. Ceci est fait par une régression linéaires des moindres carrés.

4.3 Graphique de Variance

Cette méthode graphique est basée sur le fait que pour un processus à mémoire longue de paramètre β, la variance du processus des moyennes est asymptotiquement équivalent à $m^{-\beta}$. Donc en traçant $\log \mathrm{var} X^{(m)}$ en fonction de $\log m$, on doit obtenir une droite de pente $-\beta$ pour m assez grand. Il suffit donc de faire une régression linéaire pour obtenir une estimation de β, et donc de H (qui vaut $1-\beta/2$).

4.4 Périodogramme

Pour un processus à mémoire longue, on a:

$$S(\omega) \sim_{\omega \to 0} C_2 \omega^{-(1-\beta)}$$

En procédant donc de la même manière que pour le graphique de variance, on obtient une estimation du paramètre de Hurst

Les trois méthodes d'estimation du paramètre de Hurst décrites ci-dessus ne permettent pas d'obtenir un intervalle de confiance. Cependant elles ont été utilisées dans notre étude car elles donnent de bons résultats en pratique (cf. [6] par exemple). D'autre part, la méthode du graphique de variance et du rescaled adjusted range sont assez simples à implémenter et sont rapides, même avec des échantillons très grands (environ 5mn sur une SPARC20 pour un échantillon de taille 250000). Il faut signaler cependant qu'il existe un estimateur du maximum de vraisemblance du paramètre de Hurst (l'estimateur de Whittle, voir l'article de Beran [1] pour plus de détails) qui permet de calculer un intervalle de confiance, mais d'une part cet estimateur suppose que le processus a des lois marginales gaussiennes, et d'autre part, les temps de calculs sont beaucoup plus longs (environ 30 minutes sur une Sparc20 pour un échantillon de taille 20000). Dans le cadre du stage, nous nous sommes contentés d'utiliser les trois méthodes décrites précédemment qui permettent déjà de distinguer la présence de mémoire longue.

Chapitre 5

Application: Étude du trafic du WWW

5.1 Généralités

Le principal outil utilisé pour cette étude est le logiciel S+ (dans sa version 3.3) utilisé sur une station de travail SUN Sparc20. Ce logiciel fourni une vaste bibliothèque de fonctions statistiques, ainsi que de nombreuses possibilités graphiques. Cependant, certaines fonctions (calcul du rescaled adjusted range et variance du processus des moyennes) ont dues être écrites en langage compilé (C++) pour des raisons d'efficacité. D'autre part, plusieurs scripts en langage Perl ont été écrits afin extraire les données qui nous intéressent à partir des fichiers de logs (où sont archivées toutes les connections au serveur) décrits dans la section suivante.

5.2 Description des données

Les données étudiées sont les fichiers de « logs » du serveur Web du W^3C, *i.e.* l'URL `http://www.w3.org` durant les mois de Juin et Juillet 1995. Dans chacun de ces fichiers se trouve toutes les requêtes de documents adressées au serveur (cf. exemple de fichier dans la table 5.1). Pour chacune d'entre elle, l'adresse IP d'origine est précisée, ainsi que la date de réception de cette requête (au format: hh:mm:ss) et la taille en octets du fichier. Il faut préciser que la taille en question correspond à la quantité de données effectivement transmise. Il se peut donc que cette quantité soit nulle dans certains cas, en raison du mécanisme de mémoire cache au niveau de l'origine de la requête. En effet, lorsque le fichier correspondant à une requête est présent en mémoire cache (et à condition que la version du fichier en cache soit la même que la version du fichier se trouvant sur le serveur) la quantité d'octets transmise est nulle (cela correspond au code 304, cf. exemple 5.1).

On ne connaît pas par contre la date de fin de transmission de ce fichier, et si la transmission a été interrompue par le client, la requête ne se retrouve pas dans le fichier (elle apparaît dans un fichier d'erreurs), et la taille des données effectivement transmises avant l'interruption n'est pas connue. On peut remarquer tout de suite que ces données ne ne permettent pas de connaître le trafic réel à la sortie du serveur (*i.e.* on ne connaît pas le le nombre d'octets par secondes réellement transmis par le serveur, ni le nombre de paquets

TCP). Néanmoins, celles-ci permettent déjà de donner une indication sur la structure de corrélation du trafic à l'entrée du serveur, et de donner une première indication sur la présence où non de corrélations sur le long terme. Les figures 5.2 (pour le nombre de requêtes par seconde) et 5.3 (pour le nombre d'octets par seconde) donnent un aperçu des différents résultats obtenus à l'aide des estimateurs évoqués dans le chapitre 4.

```
131.107.1.105 - - [06/Aug/1995:23:59:52 +0500] "GET /hypertext/DataSources/WWW/
Servers.html HTTP/1.0" 200 17853
203.3.164.146 - - [06/Aug/1995:23:59:53 +0500] "GET /hypertext/WWW/Icons/WWW/Vi
rtual_Library.gif HTTP/1.0" 304 0
203.3.164.146 - - [06/Aug/1995:23:59:53 +0500] "GET /hypertext/WWW/Icons/WWW/st
ar11t.gif HTTP/1.0" 304 0
203.3.164.146 - - [06/Aug/1995:23:59:54 +0500] "GET /hypertext/WWW/Icons/WWW/ne
w.gif HTTP/1.0" 304 0
204.220.36.202 - - [06/Aug/1995:23:59:56 +0500] "GET /hypertext/WWW/MarkUp/html
3/overview.html HTTP/1.0" 200 3294
204.31.232.92 - - [06/Aug/1995:23:59:49 +0500] "GET /hypertext/DataSources/WWW/
Servers.html HTTP/1.0" 200 17853
199.171.21.8 - - [06/Aug/1995:23:59:38 +0500] "GET /hypertext/Dist/src/WWWDaemo
n.tar.Z HTTP/1.0" 200 325605
203.3.164.146 - - [06/Aug/1995:23:59:51 +0500] "GET /hypertext/DataSources/bySu
bject/Overview.html HTTP/1.0" 200 18835
128.141.201.214 - - [07/Aug/1995:00:00:00 +0500] "GET /hypertext/WWW/Clients.ht
ml HTTP/1.0" 200 5934
```

TAB. 5.1 – *Exemple de fichier contenant les requêtes*

5.3 Résultats

Étant donné les données auxquelles on avait accès, on s'est intéressé à deux types de processus: le processus du nombre de requêtes par seconde, et celui du nombre d'octets à transmettre par seconde. Pour avoir un ordre de grandeur du trafic, voici quelques chiffres significatifs (pour une journée de la semaine):

- Nombre de requêtes: ~ 400000
- Quantité d'octets transmis: $\sim 2,7$Go
- Nombre d'adresses IP différentes: ~ 55000

On voit d'après ces chiffres que le nombre moyens de requêtes par seconde et par adresse IP (à chaque adresse IP correspond une machine unique, donc un ou plusieurs utilisateurs) est d'environ 8. Ce chiffre peut paraître élevé, mais il fait se rappeler qu'à une page HTML correspond en fait plusieurs requêtes: une pour le fichier HTML lui-même, et une pour chaque image (au format GIF dans la plupart des cas) éventuellement située dans cette page.

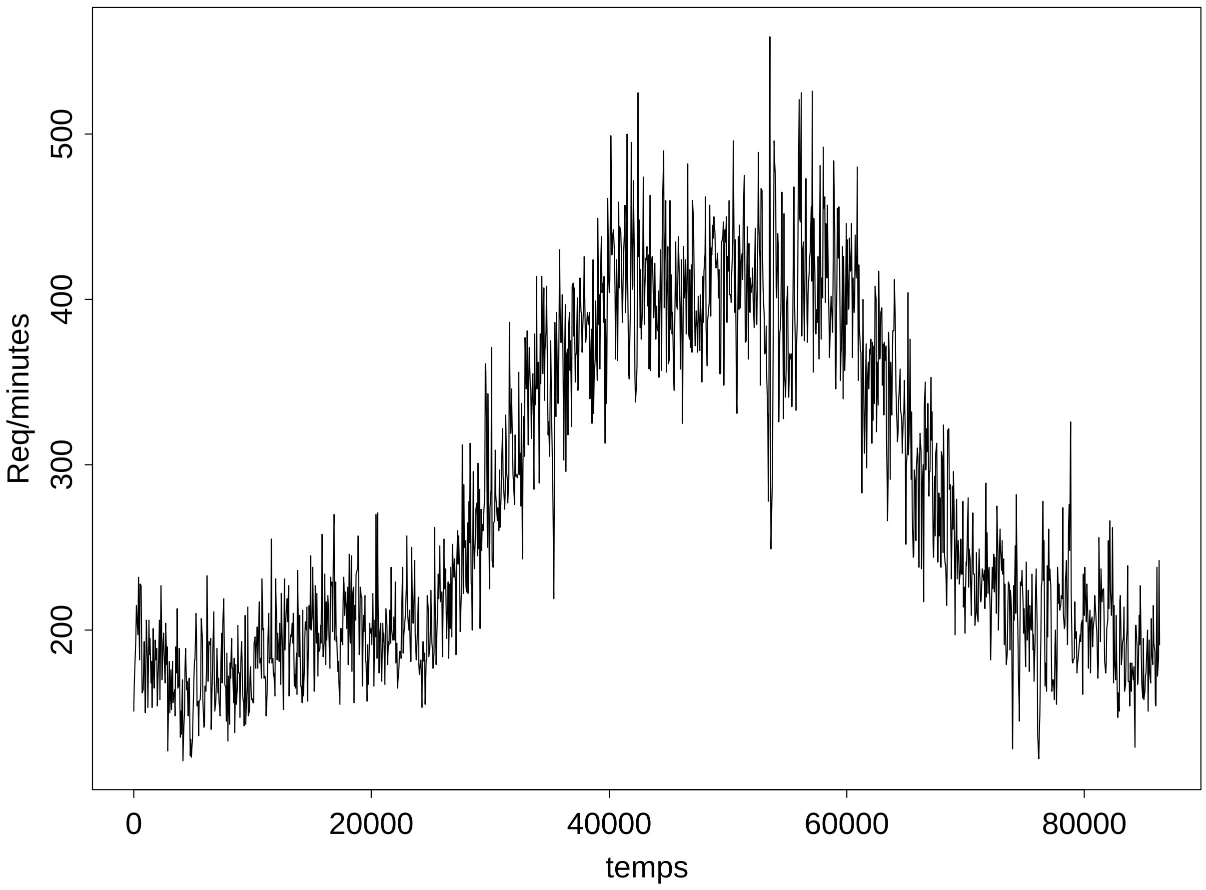

FIG. 5.1 – *Nombre de requêtes par minute le 07/09/95*

Étant donné les remarques faites précédemment, l'estimation du paramètre n'a de sens que pour un processus stationnaire (ou au moins à covariance stationnaire). Cependant, le processus du nombre de requêtes n'est pas stationnaire à l'échelle d'une journée comme le montre clairement la figure 5.1. On constate sans surprise que le nombre de requêtes est beaucoup plus important entre 10h et 18h (heure locale au MIT), ce qui correspond aux heures de travail aux États-Unis. Cependant, le trafic reste élevé même en dehors de cette période de temps.

Par conséquent, afin d'estimer le paramètre de Hurst, on a considéré des intervalles de temps durant lequel le trafic est (en première approximation) stationnaire, par exemple entre $\sim 11h$ (40000s) et $\sim 16h30$ (60000s). La figure 5.2 montre les résultats obtenus pour l'estimation du paramètre de Hurst pendant cet intervalle de temps quand on considère le processus du nombre des requêtes par seconde.

La figure 5.3 montre les mêmes estimations, mais en considérant cette fois le processus du nombre d'octets demandé au serveur.

On constate sur cette figure que les trois méthodes d'estimation de H donnent des résultats du même ordre, mais cependant non strictement identiques.

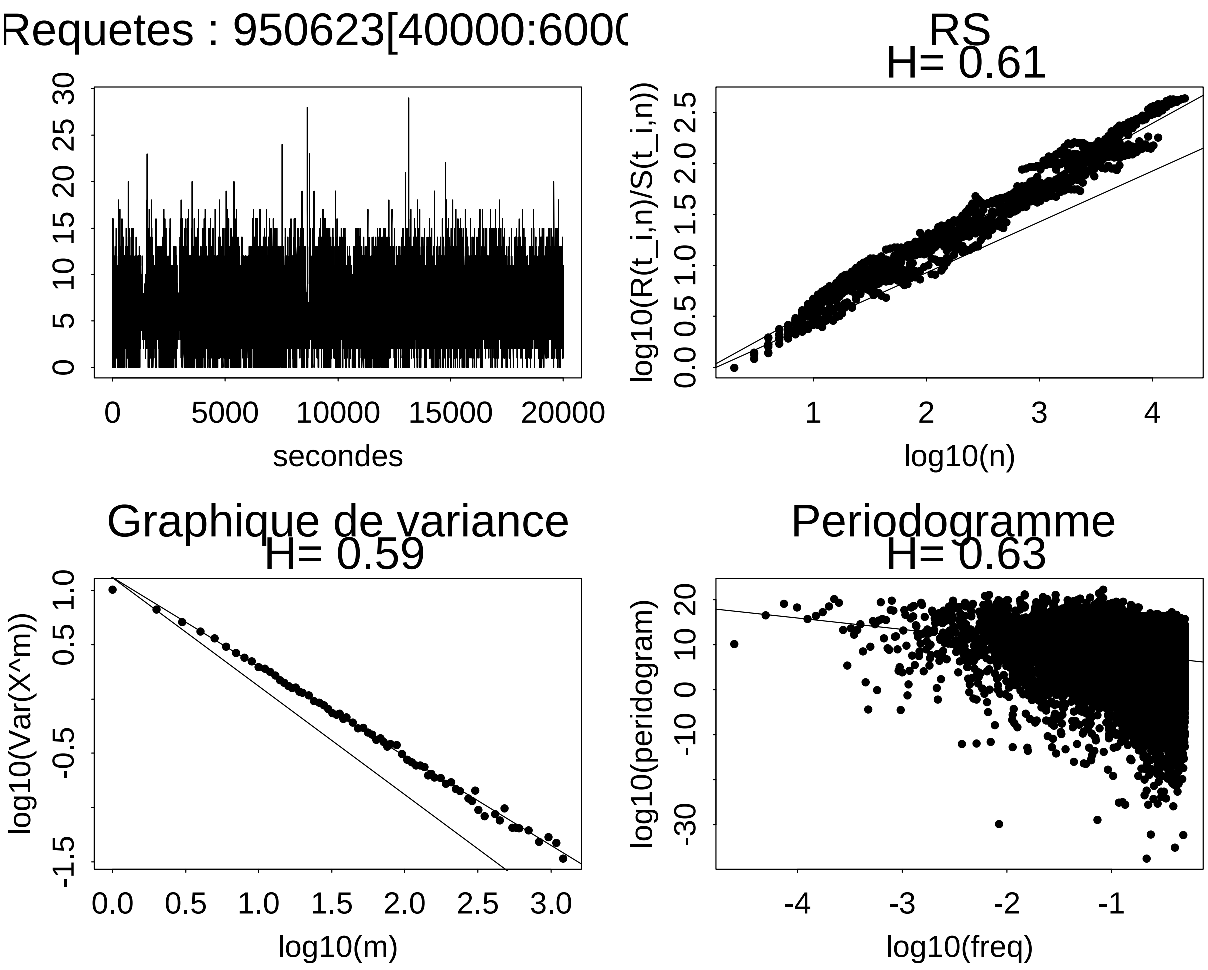

FIG. 5.2 – *Estimation de H pour le processus des requêtes*

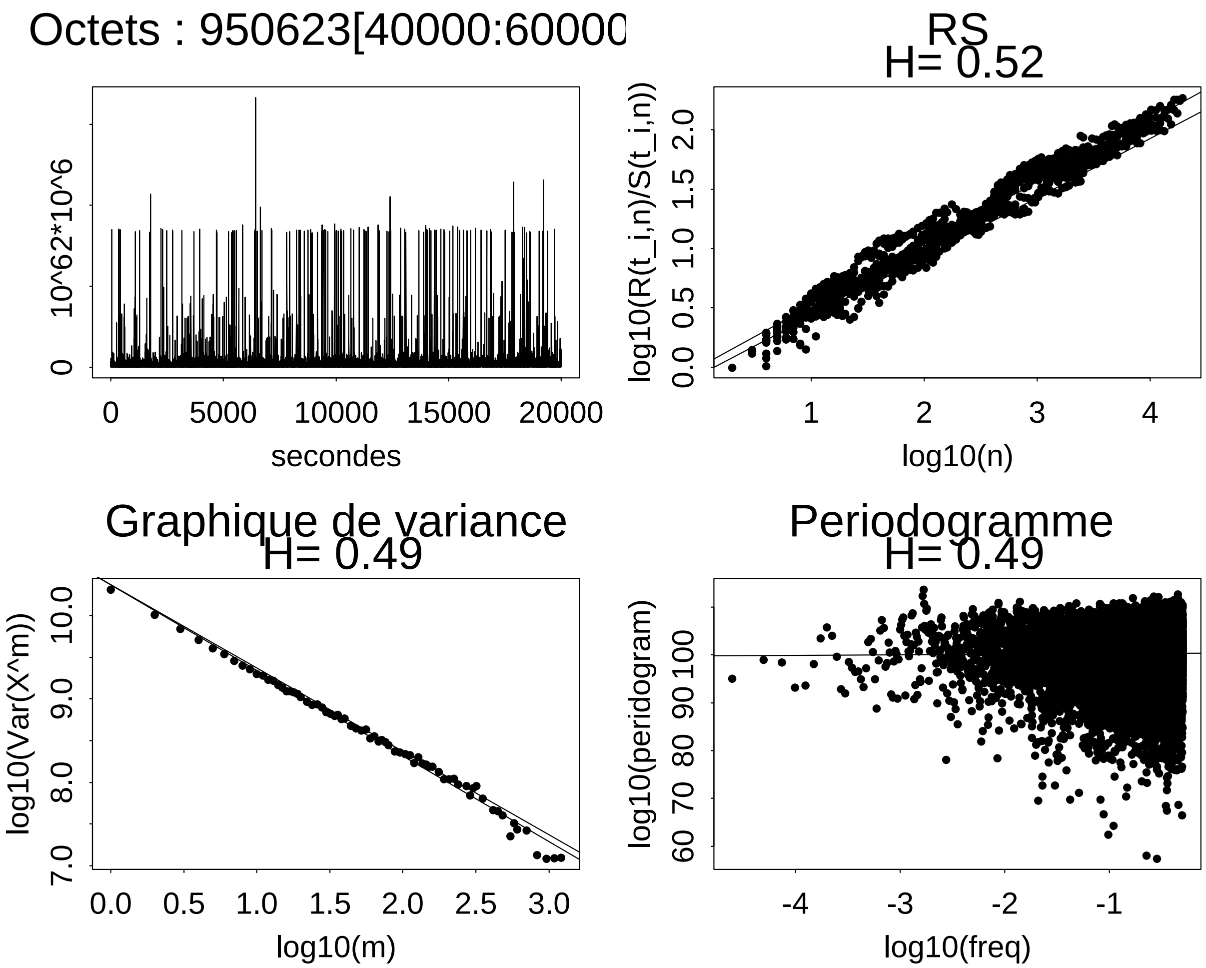

FIG. 5.3 – *Estimation de H pour le processus du nombre d'octets*

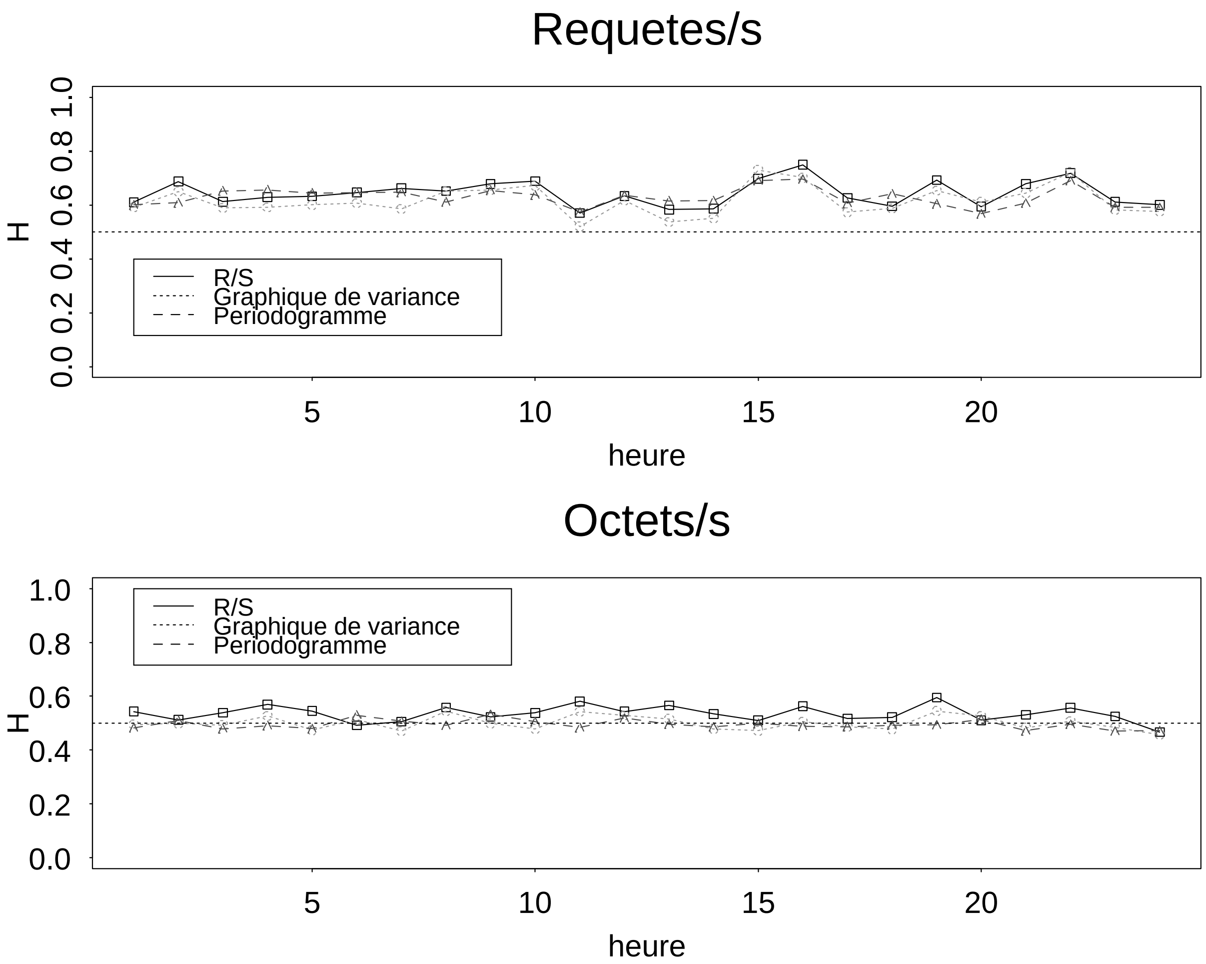

FIG. 5.4 – *Estimation de H heure par heure sur 1 journée*

On à également estimé les variations du paramètre de Hurst au cours d'une journée entière en calculant le paramètre heure par heure (figure 5.4).

D'autres tests ont été effectués sur d'autres intervalles de temps et sur d'autres journées. Tous les résultats observé indiquent une nette différences entre les deux processus. En effet, le paramètre de Hurst du processus du nombre des requêtes est estimé à 0.6 (ou légèrement plus), alors que celui du nombre d'octets est très proche de 0.5

Étant donné la différence de comportement observé (du point de vue des corrélations) entre le trafic des requêtes et du trafic en octets, on a également analysé le processus des tailles de fichier par requête (au lieu de considérer le nombre d'octets à transmettre par seconde, on considère le nombre d'octets à transmettre par requête). L'estimation du paramètre de Hurst pour ce processus est proche de 0.5.

Chapitre 6

Conclusion

L'étude que nous avons mené montre que, contrairement à ce que l'on pouvait s'attendre au vu des différentes études qui ont été faites ces derniers temps ([6],[17] et [5]), la mémoire longue est, selon les processus analysés, soit absente, soit de moindre importance que dans les études citées ci-dessus (paramètre de Hurst légèrement supérieur à 0.6 contre 0.8 voire 0.9 dans [6]). Il faut remarquer que l'étude s'est faite à partir des requêtes d'accès au serveur, et nom au trafic engendré par celui-ci dans les liens de communication. Il serait donc intéressant, si on s'intéresse au dimensionnement de ces liens, de recueillir des données sur le trafic réel en sortie du serveur afin d'en étudier les propriétés statistiques, et d'en proposer un modèle. Dans cette optique, l'installation d'un serveur miroir à l'INRIA va permettre de collecter les données nécessaires. Si l'on s'intéresse au dimensionnement du serveur lui-même, le processus des demandes d'octets par secondes semble être celui qui permette le mieux de décrire la charge à l'entrée de ce serveur. Comme notre étude indique que ce processus ne manifeste pas de signes significatifs de mémoire longue, le recours à des modèles plus classiques est envisageable. Cependant, il faut tenir compte des réserves déjà évoquées dans la partie « *description des données* » sur le manque de précision des données recueillient à partir des fichiers de logs.

Bibliographie

[1] J. Beran. "Statistical Methods for data with long-range dependence". *Statistical Science*, 7(4):404–427, 1992.

[2] D.R. Cox. "Long-Range Dependence : A Rewiew". In H.D.David and H.T.David, editors, *Statistics : An Appraisal*, pages 55–74. The Iowa State Universety Press, 1984.

[3] N.G. Duffield and N. O'connell. "Large deviations and overflow probabilities for the general single server queue, with applications". DIAS-STP-93-30, Dublin Institute for Advandced Studies, 1993.

[4] N.G. Duffiels, J.T. Lewis, N. O'connell, R. Russel, and F. Toomey. " Statistical issues by the Bellcore data ". Preprint.

[5] S.M. Klivansky, A. Mukherjee, and C. Song. "On Long-Range Dependence in NSFNET Traffic". Preprint, 1994.

[6] W.E. Leland, M.S. Taqqu, W. Willinger, and D.V. Wilson. "On the Self-Similar Nature of Ethernet Traffic". *IEEE/ACM Transactions on Networking*, 2(1):1–15, February 1994.

[7] W.E. Leland, M.S. Taqqu, W. Willinger, and D.V. Wilson. "Self-Similarity in high-speed packet traffic: Analysis and modeling of Ethernet traffic measurements". *Statistical Science*, 10(1):67–85, 1995.

[8] W.E. Leland and D.V. Wilson. "High Time Resolution Measurements and Analysis of LAN Traffic: Implications for LAN Interconnection". In *Proceedings of IEEE INFOCOM'91*, Bal Harbour, 1991.

[9] N. Likhanov, B. Tsybakov, and N.D. Georganas. "Analysis of an ATM buffer with self-similar input traffic". In *Proceedings of Infocom'95*, April 1995.

[10] B. Mandelbrot and M.S. Taqqu. "Robust R/S analysis of long run serial correlation". In *Proc. 42nd Session ISI*, volume XLVIII of *2*, pages 69–99, 1979.

[11] B.B. Mandelbrot and J.W. Van ness. "Fractal Brownian motions, fractal noises, and applications". In *SIAM Rewiew 10*, pages 422–437, 1968.

[12] J. Migge. "Recherches sur la mémoire longue dans les suites de nombres pseudo-aléatoires". Mémoire de licence, Université Libre de Bruxelles, 1994.

[13] I. NORROS. “A storage model with self-similar input”. *Queueing Systems*, 14:387–396, 1994.

[14] I. NORROS. “On the use of Fractionnal Brownian Motion in the theory of connectionless Networks”. *IEEE Journal on Selected Area of Communications*, 13(6), August 1995.

[15] M. PARULEKAR and A. MAKOWSKI. “Buffer overflow probabilities for a multiplexer with self-similar traffic”. Submitted to Infocom’96.

[16] V. PAXSON. “Fast approximation of self-similar network traffic”. Preprint, 1995.

[17] V. PAXSON and S. FLOYD. " Wide-Area Traffic: The failure of Poisson Modeling ". In *Proceedings of ACM/Sigcomm’94*, pages 257–268, September 1994.

[18] C. ROLLAND. *LaTeX, guide pratique.* Addison-Wesley, 1993.

[19] G. SAPORTA. *Probabilités, analyse de données et statistique.* Technip, 1990.

[20] R. SHERMAN, M.S. TAQQU, W. WILLINGER, and D.V. WILSON. “Statistical analysis of Ethernet LAN Traffic at the Source Level”. To appear in: Proc. of the ACM/Sigcomm’95.