A comparative study of different heavy tail index estimators of the flow size from sampled data

Patrick Loiseau
patrick.loiseau@ens-lyon.fr

Paulo Gonçalves
paulo.goncalves@ens-lyon.fr

Pascale Primet
Vicat-Blanc
pascale.primet@ens-lyon.fr

INRIA RHÔNE-ALPES, 655 avenue de l’Europe, Montbonnot, 38334 Saint Ismier Cedex, FRANCE
UNIVERSITÉ CLAUDE BERNARD LYON 1, LIP, ÉNS de Lyon, 46 allé d’Italie, 69364 LYON CEDEX 07, FRANCE

ABSTRACT
Categories and Subject Descriptors

General Terms
WHAT IS THAT ?

Keywords
NOT required for Proceedings

1. INTRODUCTION
1.1 Motivation
Grids are distributed systems, based on large distances networks, using shared computation, stocking and visualisation resources. Compared to clusters, they introduce new scales of heterogeneity, co-operative equipment, users’ community size, inter-dependant processes number, processing capacities, bandwidth, etc. . . The large distances between computation entities leading to large delays, together with the possibility to loose packets turn communication performances into a major challenge in grid networks.

For a decade, research on grid performances have essentially been based on internet transport protocols such as TCP or UDP. However, the particular topology and the specific applications of the grids make that the grid context is also very different from the internet. For example, our grid testbed, Grid5000, is an experimental grid platform gathering 9 sites geographically distributed in France featuring a total of 5000 CPUs with a star topology.

This give rise to a few interrogations : Are these protocols adapted to grid applications ? Do they guarantee optimal Quality of Service (QoS) and security ? What are the influence of the different parameters of these protocols on the performance of grid specific applications ? To answer these questions, traffic characteristics have to be studied in grid context.

Traffic characteristics have already been studied in the internet for a decade. Long Range Dependance (LRD) and self similarity have been observed in internet traces. Typical flow characteristics such as heavy tail have also been observed. Then, the modelling of the traffic has become a very active field of research. A few theoretical and empirical results arose about traffic characterisation. But these results are mainly based on the traffic observed in the internet. Are these results still valid in the grid context ? To what extend can they be adapted to grids ?

A particular result which links the Hurst parameter observed on the traffic to the flow size distribution is the following [5]:

Theorem 1. (Taqqu, Willinger, Sherman) : A simple ON/OFF source model with heavy tail distributed periods generates long range dependance in the aggregated throughputs. Tail parameter \( \alpha \) and LRD parameter \( H \) rely according to :

\[
H = \frac{3 - \alpha}{2}.
\]  

We want to find out to what extend is this theorem valid in grids’ situations. To reach this goal, we want to be able to measure \( H \) and \( \alpha \) in real situations.

When observing the entire traffic (i.e. every packets are picked), there exist a lot a method to measure \( H \) (see [1] and reference within) and \( \alpha \) [4, 2, 3].

However, because of memory and CPU consumption issues, we can’t pick every single packet. It is necessary to sample, i.e. to pick only a sub-sample of the packets going through the link. When observing only a sub-sample of the traffic, the estimation of the LRD and tail parameters is harder. The problem of estimating \( H \) from sampled traffic was addressed by Hohn and Veitch in the case where the traffic is self similar.

The major question addressed in this paper is the estimation of the tail parameter \( \alpha \) from sampled traffic.

1.2 Contribution and outline
The problem is simple, but an analytic solution is difficult to find. In section 2, the problem is explained. Basic definitions are given, and a mathematical formulation of the problem is given. In the section 3, a few different solutions to estimate $\alpha$ are explained. The first solution is based on stochastic counting of the sampled flows, with a functional link between these counting and $\alpha$. It has been proposed by Antunes et al.. The second solution is based on a kernel estimation of the tail parameter directly on the observed distribution, which is theoretically proved to be asymptotically consistent. The kernel estimator is based on a wavelet decomposition. It has been proposed by Gonçalves and Riedi. The last solution is based on an approximate inversion method, using an $a$ priori. The section 4 presents the results of each solution. They are compared on the base of traffic simulation. The contribution of this paper are in two different levels. Firstly a new method of estimating $\alpha$ is proposed using a algebraically decreasing $a$ priori. Then, the second contribution is a systematic study of different kind of estimators, in order to compare them, using a traffic simulator for different values of $\alpha$.

2. FORMULATION AND NOTATIONS

2.1 Variable description

To analyse the traffic characteristics, we adopt a flow-based approach.

2.1.1 Original flows

We consider here a section of the traffic, constituted by a succession of IP packets. The duration of this section is $T$. This section is studied independently of the rest of the traffic. The basic definition of a flow is a set of packets with the same source and destination IP and port, and the same transport protocol (TCP or UDP). Some refinements can be added to this definition, such as considering a maximal time between two consecutive packets belonging to the same flow, namely a timeout. Flows include bulk data transfer, control traffic, workloads, etc ... It is a none application oriented notion.

The number of packets in a flow will be referred to as the flow length or flow size equivalently. The random variable corresponding to the length of a flow will be noted $X$. The original flow distribution is the distribution of the random variable $X$ :

$$P(X = i) = \phi_i \propto E\{f_i\},$$

where $f_i$ is the frequency of $i$-size flows.

A random variable $Z$ is called heavy tailed with tail parameter $\alpha$ if

$$P(|Z| > z) = z^{-\alpha} L(z),$$

where $L$ is a slowly varying function, i.e. $L(tz)/L(z) \to 1$ as $z \to \infty$ for any $t > 0$. In the sequel, we will systematically assume heavy tailed original distribution, and we will look for the tail parameter of the original distribution $\alpha$.

From the observation of the frequencies $f_i$, or equivalently from the observation of the random variable $X$, we can use a lot a different methods to estimate $\alpha$. The Hill estimator is based on maximum likelihood principle. Other method exists, such as the wavelet based method from Gonçalves and Riedi, the method from Nolan, etc ... The parameter $\alpha$ can also be simply estimated via a linear regression in log-log, as we will see later. However, because of memory consumption and CPU consumption issues for high speed links, we cannot analyse each packet on the link : sampling is necessary. Then we don’t observe directly the random variable $X$ neither its distribution.

2.1.2 Sampled flows

The simplest way of performing packet sampling is to pick 1-out-of-$N$ packets, where $N$ is an integer. This method is called deterministic sampling. It is implemented in Cisco routers, (cite...). However, considering random packet sampling may drastically simplify theoretical considerations. Random packet sampling with rate $p = \frac{1}{N}$ consists in picking each packet with a probability $p$. If we assume a sufficiently large number of flows and a good mixing of the flows, the two different method are similar (cite deter and the first paper from Duffield...). Then we will assume probabilistic sampling for theoretical discussions whereas deterministic sampling is used experimentally.

After sampling, the size of sampled flows (i.e. the number of picked packet in a flow) is observed. The random variable corresponding to the size of a sampled flow is denoted as $Y$. The sampled distribution is the distribution of the random variable $Y$ :

$$P(Y = j) \propto E\{g_j\},$$

where $g_j$ is the frequency of $j$-size sampled flows observed in a time window of length $T$.

The problem addressed in this paper is : how to estimate the tail parameter of the original distribution $\alpha$ from the observed sampled distribution ?

2.2 Problem formulation

2.2.1 Link between the original and sampled distribution

Assuming random packet sampling, the probability that a sampled flow has the size $j$ given that the original flow has the size $i$ is binomial :

$$P(Y = j|X = i) = B_p(i,j) = \left(\frac{i}{j}\right) p^i (1-p)^{i-j}, \text{ where } p = \frac{1}{N}.\quad(5)$$

Then, the link between the sampled distribution and the original distribution can be written using the complete probability formula :

$$P(Y = j) = \sum_{l=j}^{\infty} B_p(l,j)P(X = l),$$

or equivalently

$$E\{g_j\} = \sum_{l=j}^{\infty} B_p(l,j)E\{f_l\}.$$  

This is the key equation which we wish to inverse in order to estimate the original distribution.
2.2.2 Maximum likelihood estimation

As in every problem of estimation, we firstly want to perform maximum likelihood estimation. The maximum likelihood estimator of the original distribution $\phi = \{\phi_i, i = 1, \ldots, \infty\}$ based on the observations $g = \{g_j, j = 1, \ldots, \infty\}$ is the original distribution $\phi^{\text{ML}}$ which make maximal the probability to observe the distribution $g$ given the distribution $\phi$ (or equivalently the log of this probability, the log-likelihood $L(\phi) = \log P(g|\phi)$) with respect to the constraint $\phi \in \Delta = \{\phi : \phi_i \geq 0 \forall i, \sum_i \phi_i = 1\}$:

$$\phi^{\text{ML}} = \arg \max_{\phi \in \Delta} L(\phi). \quad (8)$$

To give the expression of the log-likelihood, we assume that all the flows are picked, that mean that we pick at least one packet from each flow. In a real situation, some flows are missed, and we have to first estimate the probability that a flow from each flow. In a real situation, some flows are missed. Assuming that we can estimate the original distribution with the use of an EM algorithm. Secondly, we show how we can infer the original distribution from the sampled distribution, then we can estimate the tail parameter on the inferred distribution via a linear regression as explained previously. Firstly, the problem of maximum likelihood estimation is addressed, and we briefly show how the MLE can be approximated via the EM algorithm. Secondly, we show how we can infer the original distribution with the use of an a priori.

2.2.3 A priori probability

An other possibility is to use an a priori for the original distribution, in order to calculate the conditional probability $P(X = i|Y = j)$ with the Bayes Formula:

$$P(X = i|Y = j) = \frac{P(Y = j|X = i)P(X = i)}{P(Y = j)} = \frac{B_p(i, j)P(X = i)}{\sum_{i} B_p(i, j)P(X = i)} \quad (10)$$

Then, we can infer the original distribution via:

$$P(X = i) = \sum_{j=1}^{\infty} P(X = i|Y = j)P(Y = j) \quad (11)$$

2.2.4 Estimation of the tail index via linear regression

Assuming that we can estimate the original distribution $E\{F_i\}$, then we can calculate the corresponding complementary cumulative density function:

$$E\{F_i\} \propto P(X \geq i). \quad (12)$$

Now, assuming a Pareto law for the ccdf:

$$E\{F_i\} \propto \left(\frac{b}{x}\right)^{\alpha}, \quad (13)$$

for some $b$, we have

$$\log P_i = (\alpha) \log i + \beta, \quad (14)$$

holding for some values of $i$. Then, assuming that the logarithms of the random variables $F_i$ (number of flows with more than $i$ packets) are gaussian, we can estimate $\alpha$ via a linear regression based on mean-square method in some scale range.

3. DIFFERENT ESTIMATORS

3.1 Statistical inference

The aim of the statistical inference methods is to infer the original distribution from the sampled distribution, then we can estimate the tail parameter on the inferred distribution via a linear regression as explained previously. Firstly, the problem of maximum likelihood estimation is addressed, and we briefly show how the MLE can be approximated via the EM algorithm. Secondly, we show how we can infer the original distribution with the use of an a priori.

3.1.1 MLE via EM algorithm

The MLE is given by equation 8. An analytic formulation of this estimator is hard to find. But we can approximate the MLE iteratively with the EM algorithm. This approach for the estimation of the original distribution were first introduced by Duffield et al. (cite), and was used with a few transformations by Liu et al. (cite).

The EM algorithm was introduced by Dempster et al. (cite), it is well developed in the book from McLachlan and Krishnan (cite). It is a broadly applicable algorithm that provides an iterative procedure for computing MLE in situations where, but for the absence of some additional data, ML estimation would be straightforward. The observed data is viewed as being incomplete and is regarded as an observable function of the so-called complete-data. Let $f_{ij}$ denote the frequency of $i$-size original flows giving rise to $j$-size sampled flows. The set of $f_{ij}$ for every $i$ and $j$ is the complete data. The observation $g$ can simply be recovered from the complete data $g = \sum_{i=1}^{\infty} f_{ij}$. Assuming known the complete data, we form the complete data log-likelihood:

$$L_c(\phi) = \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} f_{ij} \log \phi_i c_{ij}. \quad (15)$$

After an initialisation (pick some initial flow length distribution $\phi^{0}$, for example a uniform distribution), an iteration of the algorithm can be decomposed as follows:

E step. Form the expectation $Q(\phi, \phi^{(k)})$ of the complete data log-likelihood, conditional on the observation, according to the distribution at the $k$-th iteration $\phi^{(k)}$:

$$Q(\phi, \phi^{(k)}) = \mathbb{E}_{\phi^{(k)}} \{L_c(\phi)|g\} \quad (16)$$

M step. Define

$$\phi^{(k+1)} = \arg \max_{\phi \in \Delta} Q(\phi, \phi^{(k)}). \quad (17)$$

In our case, with direct calculation of the expectation and utilisation of the Lagrange formalism for the maximisation
with respect to the constraint, we have the expression:

\[ \phi_i^{(k+1)} = \frac{\phi_i^{(k)}}{\sum_{j \geq 1} g_j} \sum_{i \in I_k} g_j c_{ij} \frac{g_j c_{ij}}{\sum_{j \in \mathcal{L}} \phi_i^{(k)} c_{ij}}. \] (18)

As usually when using a MLE, this is a computationally costly estimation. Moreover, we have to find out the number of iteration to perform before stopping. As we will discuss later, this is an important problem in our particular problem. For this reason, the utilisation of MLE of the original distribution for the estimation of the tail parameter is will not prove to be a good solution. It is why we now explore the method for estimating the original distribution with an a priori distribution as explained in section 2.2.7, that we call approximate inversion method.

### 3.1.2 Approximate inversion

**Scaling-based method.** For this method, we consider a uniform a priori, i.e., \( \phi_i \) does not depend on \( i \). Thus, we can compute the conditional probability 10:

\[ \mathbb{P}(X = i | Y = j) = \frac{B_p(i, j) \mathbb{P}(X = i)}{\sum_l B_p(l, j) \mathbb{P}(X = l)} = \frac{B_p(i, j)}{\sum_l B_p(l, j)} = \frac{B_p(i, j)}{N} \] (19)

Then we use for this conditional probability the approximation:

\[ \mathbb{P}(X = i | Y = j) = \begin{cases} \frac{1}{N} & \text{for } i = jN, \ldots, (j + 1)N - 1 \\ 0 & \text{else.} \end{cases} \] (20)

This approximation leads to a very simple expression for the inferred original frequency:

\[ \hat{f}_i = \frac{g_i}{N}, \forall i. \] (21)

This is a very natural expression where a \( j \)-size sampled flow comes from a \( i \)-sized original flow with \( i \) verifying \( \lceil \frac{i}{j} \rceil \). It corresponds to the simple idea of "multiplying by \( N \)."

To improve the accuracy of the original distribution inferred via this method, we can improve the approximation of the conditional probability 19. Improving the approximation of the conditional probability 19 would smooth the inferred distribution. However, as we want to estimate the tail parameter of a distribution which is supposed to be heavy tailed, we believe that it is important to take an a priori including a heavy tail. Rather than working on improvement of the conditional probability, we take another a priori.

**Pareto a priori.** As to estimate the tail parameter, the original distribution is supposed to follow a Pareto law (see 2.2.4), we propose to take a Pareto a priori with \( b = 1 \) and \( \alpha = \alpha^{op} \) (see equation 13) for the cdf. This is equivalent to taking a Pareto a priori with \( \alpha = \alpha^{op} + 1 \) for the probability density function. This leads to the conditional probability:

\[ \mathbb{P}(X = i | Y = j) = \frac{B_p(i, j)}{B_p(i, j + \alpha^{op})} \left( \frac{\sum_{l = j}^{\infty} B_p(l, j)}{B_p(i, j + \alpha^{op})} \right). \] (22)

We denote by \( P_{ij} \) this conditional probability. The choice of the value to be used for \( \alpha^{op} \) is obviously crucial. We have to choose without knowing the value of \( \alpha \) to be estimated. We chose \( \alpha^{op} = 1 \) because it proved to be the best choice for a set of \( \alpha \) to be estimated, when looking at the final estimates.

Now we want to find an approximation of this conditional probability which make the final estimation of \( \alpha \) accurate. We propose the approximation:

\[ \mathbb{P}(X = i | Y = j) = \begin{cases} 1 & \text{for } i = \bar{i} \\ 0 & \text{else,} \end{cases} \] (23)

where \( \bar{i} \) is the geometric mean of the suit of values \( i = j, \ldots, \infty \) weighted with the conditional probability \( P_{ij} \):

\[ \bar{i} = \exp \left( \frac{\sum_{i = j}^{\infty} P_{i,j} \ln i}{\sum_{i = j}^{\infty} P_{i,j}} \right). \] (24)

The choice of the approximation (23) comes from the fact that our final goal is to estimate the tail parameter. Then, we believe that it is not useful to "duplicate" each observed sampled frequency as we do with the approximation (20). Instead, we propose to affect each observed sampled frequency to a suitable value of \( \bar{i} \). A few different choice are possible for this suitable value \( \bar{i} \). The first choice we can think of is to take the value of \( i \) which make maximal the conditional probability (22). However, this choice is not suitable to the estimation of the index of Pareto law because it does not take into account the \( \frac{1}{\bar{i}} \) decreasing of the distribution. We believe that the right way to estimate the index of the Pareto law is to take the geometric mean weighted by the conditional probability as written in (24).

### 3.2 Stochastic counting estimation

The method referred to as stochastic counting estimation has been introduced by Antunes et al. (cite ...).

The section of traffic (see section 2.1.1) is divided into time windows of length \( T \), with \( T < T \). A new variable is introduced: \( W_k \) is the number of flows from which \( k \) packets are picked in a time window of length \( T \) when performing the sampling. This variable is observed by stochastic counting. Let us denoted the total number of packet and flows in a time window by \( P \) and \( K \) respectively. Then, the exact expression of the expectation of the random variable \( W_k \) is:

\[ \mathbb{E}(W_k) = KE \left\{ \frac{p^k}{k^k} \left( \frac{X}{P} \right)^k \left( 1 - \frac{X}{P} \right)^{P-k} \mathbb{I}_{\{X \geq k\}} \right\} \] (25)

If \( pP \) is large, and \( \frac{1}{P} \) is small, we have the approximation:

\[ \mathbb{E}(W_k) \sim KE \left\{ e^{-pX} \frac{p^k X^k}{k!} \right\}, \] (26)

which correspond to a Poisson approximation of the above probability. If \( X \) has a Pareto distribution 13, and \( K \) is
sufficiently large,

\[ \mathbb{E}(W_k) \sim R_k^{\frac{\alpha}{k!}} \alpha b^\alpha I_k, \]

(27)

where

\[ I_k = \int_k^{\infty} x^{k-\alpha-1} e^{-px} dx \]

(28)

One make the assumption that we can find some \( k_0 \) such as for \( k \geq k_0 \), small flows sizes flows (not following the Pareto law) have no impact on \( W_k \). Then, after a few calculation, a very simple estimation of \( \alpha \) is found for \( k \geq k_0 \):

\[ \hat{\alpha} = (k + 1) \left( 1 - \frac{\mathbb{E}(W_{k+1})}{\mathbb{E}(W_k)} \right) - 1. \]

(29)

### 3.3 Wavelet based estimation

The utilisation of the wavelet based estimator consists in the estimation of the heavy tail index of the sampled distribution \( g''(T) \). The consistency of this method relies on the theoretical result that if the original distribution is heavy tailed, then the sampled distribution is asymptotically heavy tailed with the same index:

\[ \lim_{j \to \infty} \frac{P(Y \geq j)}{P(X \geq \frac{1}{2})} = 1, \]

(30)

(see cite Deter...). To estimate the heavy tail index of the sampled distribution, we use a kernel estimator based on a wavelet analysis of the characteristic function near the origin. The maximal existing moment, which is the heavy tail index can be linked to the degree of Lipschitz continuity of the characteristic function of the distribution at the origin. This continuity can be studied via a wavelet transform. With some hypothesis on the wavelet used, it leads to a simple estimation procedure of heavy tail index of a distribution:

- From a sample data set \( \{Y_k\}_{k=1}^n \), we compute the wavelet coefficients at zero of the characteristic function, as it appears, this amounts to computing the kernel estimator \( \tilde{W}(s) \):

\[ \tilde{W}(s) = \frac{1}{N} \sum_{k=1}^n \Psi(s \cdot Y_k) \]

(31)

where the kernel \( \Psi \) is the Fourier transform of a semi-definite wavelet.

- Then the index can be estimated via a linear regression of \( \log(\tilde{W}(s)) \) against \( s \) within some scale interval. The choice of the scale interval will be discussed later.

The regularity of the wavelet must be larger than the heavy tail index to be estimated. This wavelet based estimation has been introduced by Gonçalves and Riedi. It is well developed in (cite ...).

### 4. RESULTS AND CONCLUSION

#### 4.1 Comparison methodology

Now, we want to compare the accuracy of the different estimators, for different values of \( \alpha \). Our comparison will be based on synthetic traffic generated with Matlab@. In the next two sections, we describe the generator of traffic and its major characteristics, and we present the analysis methodology.

#### 4.1.1 Traffic traces generation

The traces of traffic generated with our simulator have the following general characteristics:

- the total time of each trace was set to two hours,
- the maximal bandwidth of the link was set to 1 Gbps and the mean load of the link was set to 0.43,
- the packet size in a flow is either constant equal to 64 or 1514 Bytes, or uniformly distributed between these two values.

Now, about the flows, we have the following characteristics:

- the inter-arrival of flows is a gamma process,
- the inter-arrival of packets in a flow is a poissonian process.

These characteristics have been chosen because they are currently observed. However they have no impact on the flow size distribution, but we checked that they guarantee a good mixing of flows, necessary to suppose a probabilistic sampling.

The flow-size distribution is constant for small flows (from size 1 to size 10), then the distribution is a Pareto law with tail parameter \( \alpha \) (see 13). The original distribution is then heavy tailed with a tail parameter \( \alpha \).

#### 4.1.2 Traffic analysis methodology

We generated traces corresponding to 2 hours of traffic, for 5 different values of \( \alpha \) between 1 and 2 : 1.1, 1.3, 1.5, 1.7 and 1.9.

Each trace is divided into time window of length \( T \). This time windows are the section of traffic introduced in the section 2.1.1, which are independently studied. We chose two values of \( T \) : 20 s and 100 s because we think that they are likely to be characteristic of the dynamic for some application. Then, for \( T = 20 \) s and \( T = 100 \) s, we have respectively 360 and 72 different time windows in which we can estimate the tail parameter. For each time window, we form the original distribution \( f \).

Now, for each trace, we perform sampling for two different values of \( N \) : 10 and 100. Then for each time window, we form the sampled distribution \( g \), from which we are now trying to estimate the tail parameter with the methods previously introduced.

#### 4.2 Results

##### 4.2.1 Statistical inference

In this section, we present the inferred original distribution for the different methods of statistical inference. To present the figures, we chose an arbitrary time window of size \( T = 100 \) s, and with \( \alpha = 1.9 \).
MLE via EM. The figure 1 shows the inferred distribution obtained after 50 iterations of the EM algorithm for $N = 10$ and $N = 100$, together with the corresponding real original distribution. By looking at these two figures, we can draw a few conclusions. For $N = 10$, it seems to exist a scale range within which a linear regression could lead to a correct estimation of $\alpha$. However, this is absolutely not the case for $N = 100$. Moreover, already for $N = 10$, a few oscillations appears in the inferred distribution. This phenomenon means that too many iterations have been performed. But, as previously mentioned, the main problem with the EM algorithm is that it is very difficult to find a stopping criterion which work well in every situations.

Though the EM algorithm is very efficient in many situations, we believe that the inference of the original distribution via EM algorithm is not a good choice if the final goal is to estimate $\alpha$ for the following reasons. Firstly, it is difficult to find a stopping criterion valid in every situations, that is for every $N$ and $T$. Secondly, each step of the algorithm is very costly with respect to CPU consumption: this is a highly complex method.

However, we believe that EM can be a really good approach to the problem if we can reformulate the problem to directly estimate $\alpha$ without estimating the whole original distribution. This approach is worth further developments.

Approximate inversion. The figure 2 shows the inferred distribution for $N$ equal to 10 and 100 respectively, with the scaling method, together with the corresponding real original distribution. This figure shows that the coarseness of the scaling based estimator of the original distribution increase when $N$ increase. Moreover, the accuracy is much better for large flows than for small flows. It appears to exist both for $N = 10$ and $N = 100$ a scale range where a linear regression could be possible. The scale range chosen not to take small flows into account is as follows. The minimal value of $j$ taken into account is chosen as follows: $j_{\text{min}}$ is the smaller value of $j$ such as $B_{g,i}(i,j) < th$ and $j > \frac{4}{\alpha}$ where $th$ is a threshold (we chose $th = 0.01$) and $B$ is the value of $i$ delimiting the body from the tail. In our simulations, $B = 10$. Concretely, the values of $j_{\text{min}}$ are 5 for $N = 10$ and 3 for $N = 100$. The last value $j_{\text{max}}$ of $j$ used is the last values for which $g_j$ is not equal to zero. For the the scaling method, the regression is made within the scale range $[N_{j_{\text{min}}}, N_{j_{\text{max}}}]$.

The results obtained with this estimator are summarised in table 1. Its behaviour can also be seen in blue on the figure 5. Both bias and standard deviation increase when $\alpha$ increase for the two different values of $N$. For each value of $\alpha$, the standard deviation is bigger for $T = 20$ s than for $T = 100$ s, and it is bigger for $N = 100$ than for $N = 10$. This is simply because the smaller the time window is, or the bigger $N$ is, the fewer flows are observed. Then if we have fewer flows for the estimation, the standard deviation is bigger. This phenomenon will be observed for all the methods.

To improve the accuracy of the estimator, we propose to change the uniform $a$ priori into a Pareto $a$ priori, and to take the approximation of the conditional probability given in the equation 23 (see section ??). The figure 3 shows the inferred distribution for $N = 10$ and $N = 100$ respectively, using the approximate inversion method with a Pareto $a$ priori with $\alpha^{ap} = 1$. The accuracy of the inferred original distribution seems to be good both for $N = 10$ and $N = 100$, especially for large flows, which is the most important for the estimation of $\alpha$. To perform the estimation via a linear regression, we chose to use the observed sampled frequencies for the values of $j$ between $j_{\text{min}}$ and $j_{\text{max}}$ as defined above.

The results obtained with this estimator are summarised in table 2. Its behaviour can also be seen in green on the figure 5. The value $\alpha^{ap} = 1$ was chosen to have the best final estimation for each $\alpha$. The obtained estimator has a very little bias, for every $\alpha$. We still can distinguish an increasing of the bias with $\alpha$, but far less than with the scaling method.
Table 1: Estimator based on scaling method: results for different values of N, T and α. In each case, the estimated value of alpha (\(\hat{\alpha}\)), the standard deviation (std) and the generalised bias (g. bias = \(\sqrt{\text{bias}^2 + \text{std}^2}\)) are presented.

<table>
<thead>
<tr>
<th>T</th>
<th>N</th>
<th>est - α</th>
<th>1.1</th>
<th>1.3</th>
<th>1.5</th>
<th>1.7</th>
<th>1.9</th>
</tr>
</thead>
<tbody>
<tr>
<td>20 s</td>
<td>10</td>
<td>α</td>
<td>1.08</td>
<td>1.32</td>
<td>1.54</td>
<td>1.76</td>
<td>2.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>std</td>
<td>0.05</td>
<td>0.05</td>
<td>0.07</td>
<td>0.09</td>
<td>0.12</td>
</tr>
<tr>
<td></td>
<td></td>
<td>g.bias</td>
<td>0.05</td>
<td>0.05</td>
<td>0.08</td>
<td>0.11</td>
<td>0.15</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>α</td>
<td>1.04</td>
<td>1.35</td>
<td>1.64</td>
<td>1.93</td>
<td>2.18</td>
</tr>
<tr>
<td></td>
<td></td>
<td>std</td>
<td>0.08</td>
<td>0.12</td>
<td>0.17</td>
<td>0.25</td>
<td>0.35</td>
</tr>
<tr>
<td></td>
<td></td>
<td>g.bias</td>
<td>0.10</td>
<td>0.13</td>
<td>0.22</td>
<td>0.34</td>
<td>0.45</td>
</tr>
<tr>
<td>100 s</td>
<td>10</td>
<td>α</td>
<td>1.11</td>
<td>1.32</td>
<td>1.54</td>
<td>1.74</td>
<td>1.96</td>
</tr>
<tr>
<td></td>
<td></td>
<td>std</td>
<td>0.02</td>
<td>0.03</td>
<td>0.03</td>
<td>0.05</td>
<td>0.07</td>
</tr>
<tr>
<td></td>
<td></td>
<td>g.bias</td>
<td>0.02</td>
<td>0.03</td>
<td>0.05</td>
<td>0.07</td>
<td>0.10</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>α</td>
<td>1.11</td>
<td>1.35</td>
<td>1.64</td>
<td>1.85</td>
<td>2.11</td>
</tr>
<tr>
<td></td>
<td></td>
<td>std</td>
<td>0.04</td>
<td>0.07</td>
<td>0.09</td>
<td>0.12</td>
<td>0.16</td>
</tr>
<tr>
<td></td>
<td></td>
<td>g.bias</td>
<td>0.04</td>
<td>0.08</td>
<td>0.14</td>
<td>0.19</td>
<td>0.26</td>
</tr>
</tbody>
</table>

Figure 3: Inferred original distribution using the Pareto a priori method for T = 100. The black dots represent the real original distribution. The red curve is the inferred distribution. The tail parameter chosen for the Pareto a priori distribution is \(\alpha^{ap} = 1\).

Table 2: Estimator based on approximate inversion method with a Pareto a priori, with \(\alpha^{ap} = 2\): results for different values of N, T and \(\alpha\). In each case, the estimated value of alpha (\(\hat{\alpha}\)), the standard deviation (std) and the generalised bias (g. bias = \(\sqrt{\text{bias}^2 + \text{std}^2}\)) are presented.

<table>
<thead>
<tr>
<th>T</th>
<th>N</th>
<th>est - α</th>
<th>1.1</th>
<th>1.3</th>
<th>1.5</th>
<th>1.7</th>
<th>1.9</th>
</tr>
</thead>
<tbody>
<tr>
<td>20 s</td>
<td>10</td>
<td>α</td>
<td>1.05</td>
<td>1.28</td>
<td>1.49</td>
<td>1.70</td>
<td>1.91</td>
</tr>
<tr>
<td></td>
<td></td>
<td>std</td>
<td>0.04</td>
<td>0.05</td>
<td>0.06</td>
<td>0.08</td>
<td>0.10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>g.bias</td>
<td>0.07</td>
<td>0.05</td>
<td>0.06</td>
<td>0.08</td>
<td>0.11</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>α</td>
<td>0.95</td>
<td>1.20</td>
<td>1.44</td>
<td>1.69</td>
<td>1.91</td>
</tr>
<tr>
<td></td>
<td></td>
<td>std</td>
<td>0.06</td>
<td>0.09</td>
<td>0.12</td>
<td>0.18</td>
<td>0.24</td>
</tr>
<tr>
<td></td>
<td></td>
<td>g.bias</td>
<td>0.17</td>
<td>0.13</td>
<td>0.13</td>
<td>0.18</td>
<td>0.24</td>
</tr>
<tr>
<td>100 s</td>
<td>10</td>
<td>α</td>
<td>1.08</td>
<td>1.28</td>
<td>1.50</td>
<td>1.69</td>
<td>1.90</td>
</tr>
<tr>
<td></td>
<td></td>
<td>std</td>
<td>0.02</td>
<td>0.03</td>
<td>0.03</td>
<td>0.05</td>
<td>0.07</td>
</tr>
<tr>
<td></td>
<td></td>
<td>g.bias</td>
<td>0.03</td>
<td>0.03</td>
<td>0.03</td>
<td>0.05</td>
<td>0.07</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>α</td>
<td>1.03</td>
<td>1.24</td>
<td>1.47</td>
<td>1.66</td>
<td>1.88</td>
</tr>
<tr>
<td></td>
<td></td>
<td>std</td>
<td>0.04</td>
<td>0.06</td>
<td>0.07</td>
<td>0.09</td>
<td>0.12</td>
</tr>
<tr>
<td></td>
<td></td>
<td>g.bias</td>
<td>0.08</td>
<td>0.08</td>
<td>0.08</td>
<td>0.10</td>
<td>0.12</td>
</tr>
</tbody>
</table>

4.2.2 Wavelet estimator

The figure show the log-scale diagrams for the wavelet method. On these diagrams, we can see three different regions. For the small scales, the diagram is linear with a slope corresponding to the wavelet regularity (6 in our case). For the middle scales, the diagram is also linear, and the slope corresponds the the tail parameter to be estimated. Then, for the large scales, the diagram has a negative slope linked to the negative moments of the distribution.

Once again, we have to choose a scale range for a linear regression, such as we catch the slope of the middle scales. The lower scale bound can be taken as \(\log_{10}(\frac{1}{\max(Y)})\) under theoretical arguments (see Gonc..), where \(Y\) is the random variable corresponding to the sampled flow size in the studied time window. However, because of the time window, big flows can be cut. Then the larger values of \(Y\) are more likely to come from a split flow. Then, instead of taking max(\(Y\)), we take the second larger value. The upper scale bound can be taken as the number of flows decrease while \(k\) increase, the variance would get greater.

We chose \(a_{\text{max}} = \log_{10}(\frac{1}{\max(Y)})\) where Per is the value of \(Y\) corresponding to a percentile of the observed distribution. The chosen percentile depends on \(N\).

The results of this estimator are summarised in table 3. Its behaviour can also be seen in black on the figure 5.

4.2.3 Stochastic counting estimator

For this method we have to choose \(k_0\) (see ??) such as for \(k \geq k_0\), small flows flows (not following the Pareto law) have no impact on \(W_k\). In cite(Rob), \(k_0 = 3\) is suggested for \(N = 100\). As this value is the same as \(j_{\text{min}}\) described above, we chose \(k_0 = j_{\text{min}}\) for \(N = 10\) too. Then the estimation is performed via equation ?? with \(k = k_0\). Other values of \(k\) could be taken, be as the number of flows decrease while \(k\) increase, the variance would get greater.

The results of this estimator are summarised in table 4. Its behaviour can also be seen in red on the figure 5. We observe
Figure 4: Log-scale diagrams. The blue dots represent the wavelet transform of the characteristic function of the distribution. The vertical black bars delimit the scale range chosen for the regression.

Table 3: Estimator based on wavelet method: results for different values of $N$, $T$ and $\alpha$. In each case, the estimated value of alpha ($\hat{\alpha}$), the standard deviation (std) and the generalised bias (g. bias = $\sqrt{\text{bias}^2 + \text{std}^2}$) are presented.

<table>
<thead>
<tr>
<th>$T$</th>
<th>$N$</th>
<th>$\alpha$</th>
<th>$\hat{\alpha}$</th>
<th>std</th>
<th>g. bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>20 s</td>
<td>10</td>
<td>1.11</td>
<td>1.32</td>
<td>0.05</td>
<td>0.03</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.03</td>
<td>0.04</td>
<td>0.05</td>
<td>0.04</td>
</tr>
<tr>
<td></td>
<td>g. bias</td>
<td>0.03</td>
<td>0.05</td>
<td>0.07</td>
<td>0.04</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>1.30</td>
<td>1.54</td>
<td>0.08</td>
<td>0.04</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.04</td>
<td>0.05</td>
<td>0.08</td>
<td>0.04</td>
</tr>
<tr>
<td></td>
<td>g. bias</td>
<td>0.07</td>
<td>0.05</td>
<td>0.09</td>
<td>0.12</td>
</tr>
<tr>
<td>100 s</td>
<td>10</td>
<td>1.11</td>
<td>1.33</td>
<td>0.11</td>
<td>0.03</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.03</td>
<td>0.04</td>
<td>0.04</td>
<td>0.03</td>
</tr>
<tr>
<td></td>
<td>g. bias</td>
<td>0.03</td>
<td>0.04</td>
<td>0.06</td>
<td>0.04</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>1.33</td>
<td>1.57</td>
<td>0.05</td>
<td>0.11</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.03</td>
<td>0.04</td>
<td>0.05</td>
<td>0.11</td>
</tr>
<tr>
<td></td>
<td>g. bias</td>
<td>0.04</td>
<td>0.05</td>
<td>0.09</td>
<td>0.11</td>
</tr>
</tbody>
</table>

Table 4: Estimator based on stochastic counting method: results for different values of $N$, $T$ and $\alpha$. In each case, the estimated value of alpha ($\hat{\alpha}$), the standard deviation (std) and the generalised bias (g. bias = $\sqrt{\text{bias}^2 + \text{std}^2}$) are presented.

<table>
<thead>
<tr>
<th>$T$</th>
<th>$N$</th>
<th>$\alpha$</th>
<th>$\hat{\alpha}$</th>
<th>std</th>
<th>g. bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>20 s</td>
<td>10</td>
<td>1.06</td>
<td>1.25</td>
<td>0.17</td>
<td>0.19</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.18</td>
<td>0.16</td>
<td>0.14</td>
<td>0.15</td>
</tr>
<tr>
<td></td>
<td>g. bias</td>
<td>0.19</td>
<td>0.17</td>
<td>0.17</td>
<td>0.19</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>1.22</td>
<td>1.42</td>
<td>0.20</td>
<td>0.21</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.22</td>
<td>0.19</td>
<td>0.18</td>
<td>0.18</td>
</tr>
<tr>
<td></td>
<td>g. bias</td>
<td>0.22</td>
<td>0.19</td>
<td>0.20</td>
<td>0.21</td>
</tr>
<tr>
<td>100 s</td>
<td>10</td>
<td>1.07</td>
<td>1.25</td>
<td>0.07</td>
<td>0.07</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.09</td>
<td>0.07</td>
<td>0.07</td>
<td>0.07</td>
</tr>
<tr>
<td></td>
<td>g. bias</td>
<td>0.10</td>
<td>0.09</td>
<td>0.12</td>
<td>0.13</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>1.06</td>
<td>1.26</td>
<td>0.12</td>
<td>0.13</td>
</tr>
<tr>
<td></td>
<td>std</td>
<td>0.08</td>
<td>0.08</td>
<td>0.08</td>
<td>0.09</td>
</tr>
<tr>
<td></td>
<td>g. bias</td>
<td>0.09</td>
<td>0.09</td>
<td>0.11</td>
<td>0.13</td>
</tr>
</tbody>
</table>

a bias decreasing when $\alpha$ increase. This bias does not depend on the value of $N$. Though the standard deviation is a little bigger for $N = 100$ than for $N = 10$, this method seems to lead to an estimation which is identically accurate for any value of $N$. Let us recall here that this estimation is really simple since it is only based on a report of the expectation of two frequencies obtained by stochastic counting of the number of flows $k$ times in a sub-time window of size $T' = 5$ s for two values of $k$. To improve the estimation, we could use more than two values of $k$. For example, we could calculate the estimate of $\alpha$ via equation ?? for $k = k_0$ and $k = k_0 + 1$ and then take the mean of the two estimates.

4.3 Comparison of the methods
The figure 5 summarise the results of the estimation of $\alpha$ with the four different methods.

5. REFERENCES
Figure 5: Estimation of $\alpha$: comparison of the different methods for five values of $\alpha$: 1.1, 1.3, 1.5, 1.7, 1.9 (a little ?? is introduced for clarity purposes). The methods are: scaling method (blue), Pareto a priori method (green), wavelet-based method (black), stochastic counting method (red).