Estimating membership in a multicast session
Sara Alouf, Eitan Altman, Chadi Barakat, Philippe Nain

To cite this version:

HAL Id: hal-00641390
https://inria.hal.science/hal-00641390
Submitted on 10 Jul 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Abstract

We propose two novel on-line estimation algorithms to determine the size of a dynamic multicast group. We first use a Wiener filter to derive an optimal estimator for the membership size of the session in case the join process is Poisson and the lifetime of participants is distributed exponentially. We next develop the best first-order linear filter from which we derive an estimator that holds for any lifetime distribution. We apply this approach to the case where the lifetime distribution is hyperexponential. Both estimators hold under any traffic regime. Applying both estimators on real traces corresponding to video sessions, we find that both schemes behave well, one of which performs slightly better than the other in some cases. We further provide guidelines on how to tune the parameters involved in both schemes in order to achieve high quality estimation while simultaneously avoiding feedback implosion.

keywords – On-line estimation, multicast applications, membership size, $M/G/\infty$ queue, Wiener filter, first-order linear filter

1 Introduction

Since its introduction, IP multicast has seen slow deployment in the Internet. As stated in [2], the service model and architecture do not efficiently provide or address many features required for a robust implementation of multicast. However, the fact remains that IP multicast is very appealing in offering scalable point-to-multipoint delivery specially in satellite communications. Current research efforts tend to propose alternatives to IP multicast like the so-called “application layer multicast” [5 10 15 20], the idea being to deploy multicast at the application layer. Also, new models to support multicast communications in a more effective way have been proposed, such as the EXPRESS multicast [14]. The latter is an extension to IP multicast that provides explicit support for large-scale multicast applications such as real-time stock quote dissemination, live sports video feeds or Internet radio and TV. EXPRESS provides as well a best-effort count of the number of subscribers.

This paper is motivated by the conviction that large-scale multicast applications will be widely deployed in the future as soon as the capability becomes available. We believe that membership estimates will be an essential component of this widespread deployment as they can be very useful for scalable multicast. The membership of a session can be used for feedback suppression as it is the case in current protocols such as RTP [22] and SRM [9]. In order to regulate the amount of session/control messages sent by receivers – the idea being not to exceed 5% of overall session bandwidth – these protocols use delay timers that are tuned based on membership estimates.

The membership of a multicast session can be used for charging the sources in large-scale applications. ISPs traditionally charge their customers on an input-rate basis. An alternative pricing scheme would be to charge sources based on their audience size which is more profitable in the case of millions of subscribers.

Estimating the size of a multicast session can be quite useful to many applications. Bolot, Turletti and Waksman [3] use membership estimation to further estimate the proportion of congested receivers as needed in their videoconference system IVS. Future Internet radios and TVs will need to characterize their audience preferences and to follow the fluctuations of the audience size. Dutta, Schulzrinne and Yemini proposed an architecture for Internet radio and TV called MarconiNet [8] that relies on RTCP [22, 21]. Even though RTCP provides an easy mechanism for collecting statistics on the size of the audience, it does not scale well to large multicast session [8]. In such applications, sampling-based techniques are more appropriate.

There has been a significant research effort in devising sampling-based schemes for the estimation of the membership in multicast sessions [4 18 19 11 17 3]. The feedback algorithms presented in [4 18 19 11 17], are all at-least-one scenarios in the sense that the membership estimation is based on at least one acknowledgement (ACK) coming from the receivers. In these probabilistic schemes, the receivers send ACKs to the source in reply to a specific request, either with a certain probability as in [4] or after some random time like in [18 19 17]. Except for [18] all these schemes assume that the size of the group remains fixed over time. In a recent work [3], we propose a dynamic scheme that tracks the variations of the membership in an optimal way. The estimation algorithm used is quite simple: the source requests from
its receivers to send ACKs with probability $p$ every $S$ seconds; it collects the amount of ACKs received at each observation step and filters out these measurements to estimate the membership. To derive the optimal estimator we rely on a diffusion approximation for the heavy-traffic regime. Under the assumptions of Poisson join times and exponentially distributed connection times, the diffusion approximation yields linear dynamics which enables the design of the optimal filter using Kalman filter theory.

In this paper we propose two novel algorithms for estimating the membership based on the polling scheme presented in [3]. Our purpose is to develop an estimator under more general assumptions than the ones used in [3]. Our first approach is based on a Wiener filter, which provides the optimal dynamic estimator among all linear estimators. The dynamics is not required to be linear as in the case of the Kalman filter, which allows us to relax the heavy-traffic assumption made in [3]. Yet, in order to obtain explicit expressions for the parameters of the Wiener filter, we still have to assume that participants join the session according to a Poisson process, and that the time during which they stay in the multicast session, hereafter referred to as on-time, has an exponential distribution. Under these assumptions we design the optimal linear estimation scheme that turns out to require a filter of order one. Motivated by this structure, we then design an efficient estimation scheme for generally distributed on-times. To that end, we identify the optimal filter among all linear filters of order one. We illustrate this approach in the case of hyperexponentially distributed on-times. Alike the estimator developed with the Wiener filter, the latter estimator is valid under any traffic regime. Both estimators are then tested on real traces. Despite the fact that the traces violate the assumptions under which the estimators have been derived, very good performance are observed.

Remark 1.1 The material presented in this paper does not require the specific use of IP multicast, nor any other multicast protocol. The solutions proposed hereafter are meant to be deployed at the application layer, and only require a multicast delivery of the requests for ACKs. This delivery can be achieved either by IP multicast, or by an application-layer multicast, or even by a new multicast technology. It is however assumed that the IP address of the source is available to all receivers in the session.

The paper is organized as follows: the mathematical model of the membership is introduced in Section 2. The theory of Wiener filters is briefly presented in Section 3, and its application to the $M/M/\infty$ model comes in Section 4. The optimal first-order linear filter is developed in Section 5. Section 6 proposes some guidelines on how to choose parameters $p$ and $S$. The robustness of both estimators is addressed through validations on real traces in Section 6. Finally, open issues are discussed in Section 7 and concluding remarks are given in Section 8.

2 Multicast group modeled as an $M/G/\infty$ queue

We consider a multicast group that participants join and leave at random times. Let $T_i$ and $T_{i+1}$ be the join time and leave time, respectively, of the $i$-th participant. In the following, $D_i$ denotes the on-time of the $i$-th participant. Let $N(t)$ be the number of participants in the multicast group at time $t$ or, equivalently, the size of the multicast session at time $t$. Without loss of generality we assume that $N(0) = 0$. Then,

$$N(t) = \sum_{i \geq 1} 1(T_i \leq t < T_i + D_i) \tag{1}$$

where $1(E)$ equals 1 if the event $E$ occurs and 0 otherwise.

We shall assume that the joint times form a homogeneous Poisson process (with constant intensity $0 < \lambda = 1/E[T_{i+1} - T_i]$) and that the on-times form a renewal sequence of random variables (rvs) with common probability distribution $\Psi(x) = P(D_i < x)$, $0 < E[D_i] < \infty$, further independent of the join times. In the following $D$ will denote a generic rv with probability distribution $\Psi(x)$.

In the queueing terminology, $\{N(t), t \geq 0\}$ represents the occupation process (number of busy servers) in an $M/G/\infty$ queue [16].

At times $t = nS$, $n = 0, 1, \ldots$, with $S > 0$ a constant, each participant to the multicast session sends an ACK to the source with probability $0 < p < 1$ and does not send any feedback information to the source with probability $1 - p$. We assume that ACKs cannot be lost. However, this assumption can be relaxed if one knows the loss probability as it is possible to incorporate it in our feedback mechanism.

The ACK interval $S$ between two consecutive polling instants has to be larger than the largest round-trip time between a receiver and the source, so that all ACKs produced in a round reach the source before the (automatic) start of the next round. Note that in practice the source will have to regularly multicast the pair $(p, S)$ to ensure that each participant will know these values. Throughout the paper, $p$ and $S$ are held fixed (see Section 5 for possible extensions).

Let $Y_n$ be the number of ACKs received by the source at time $nS$. Based on the knowledge of $Y_1, \ldots, Y_n$, our objective is to find an optimal estimator (in a sense to be defined below) $N_n$ for $N_n := N(nS)$, the size of the multicast group at time $nS$. In filtering parlance, $Y_n$ is an input signal and we want to generate another signal $\hat{N}_n$ that is as close as possible to an unknown signal $N_n$ (typically by minimizing the mean square error).
For later use we briefly review some results on the 

\[ M/G/\infty \] 
queue. In steady-state, the number \( N \) of busy 
servers is a Poisson random variable with parameter \( \lambda \), namely,
\( P[N = j] = \rho^j \exp(-\rho)/j! \). In particular, both the mean and the variance of the 
number of busy servers are equal to \( \rho \). The autocovariance function 
of the stationary version of the process \( \{N(t), t \geq 0\} \), also 
denoted by \( \{N(t), t \geq 0\} \), is given by [\text{Eqn (5.39)}]

\[
\text{Cov}(N(t), N(t+h)) = \lambda \int_{|h|}^{\infty} P(D > u) \, du.
\] (2)

In the following we will denote by \( \text{Cov}_X(\cdot) \) the auto-
covariance function of any second-order discrete-time sta-
tionary process \( \{X_n\}_n \). With this notation and the de-
nition of the process \( \{N_n\}_n \), we see from (2) that

\[
\text{Cov}_N(k) = \rho \cdot \gamma^{\, |k|}, \quad k = 0, \pm 1, \ldots,
\]
with \( \gamma := \exp(-\mu \delta) \), when the on-times \( \{D_i\}_i \) are 
exponentially distributed with mean \( 1/\mu \).

Throughout the paper, we will assume that

\[
\sum_{k \geq 0} \text{Cov}_N(k) < \infty.
\]
(4)

In other words, we will exclude the situation where the

on-times are heavy-tailed (e.g. Pareto distribution).

### 3 Wiener filter

Our objective is to transform a signal \( Y_n \) (noisy observation) into another signal \( \hat{N}_n \) (estimator) that is the closest to an unknown signal \( N_n \). By closest we mean that the mean error is zero (i.e. \( \mathbb{E}[Y_n] = \mathbb{E}[N_n] \)) and that the mean square error is minimized.

Such a transformation can be achieved by the Wiener filter that identifies the optimal linear filter [12] (i.e. the filter that is optimal among all linear filters). This approach gives the transfer function of the linear filter, which can be transformed back to the time domain to obtain the impulse response of the filter. From the impulse response of the filter, the expression of \( \hat{N}_n \) as a function of \( Y_n \) and, possibly, of \( \hat{N}_{n-1}, \hat{N}_{n-2}, \ldots \), can be found. We will detail this procedure below.

Since a filter that minimizes the mean square error when the underlying processes are centered also minimizes the mean square error when the same processes are non-centered, we will derive the Wiener filter for the centered (stationary) versions of processes \( \{N_n\}_n \), \( \{N_n\}_n \) and \( \{Y_n\}_n \), denoted by \( \{\nu_n\}_n \), \( \{\nu_n\}_n \) and \( \{y_n\}_n \), respectively. We have observed in the previous section that \( \mathbb{E}[N_n] = \rho \). On the other hand

\[
\mathbb{E}[Y_n] = \mathbb{E}[\mathbb{E}[Y_n | N_n]] = \mathbb{E}[\nu N_n] = \rho \,
\]
(5)

Therefore \( \nu_n = N_n - \rho, \hat{\nu}_n = \hat{N}_n - \rho \) and \( y_n = Y_n - \rho \).

Throughout the paper, \( z \) is a complex number such that \( |z| = 1 \). Introduce

\[
S_y(z) = \sum_{k=-\infty}^{\infty} \text{Cov}_y(k) z^{-k}
\]
the \( z \)-transform of the autocovariance function (also called the power spectrum) of \( \{y_n\}_n \).

Let \( \text{Cov}_{\nu y}(k) = \mathbb{E}[\nu_{n-k} y_n] \) be the cross-correlation function of processes \( \{\nu_n\}_n \) and \( \{y_n\}_n \). We also introduce

\[
S_{\nu y}(z) = \sum_{k=-\infty}^{\infty} \text{Cov}_{\nu y}(k) z^{-k}
\]
the \( z \)-transform of \( \text{Cov}_{\nu y}(k) \). We can express \( \text{Cov}_y(k) \) and \( \text{Cov}_{\nu y}(k) \) in terms of \( \text{Cov}_\nu(k) \) as follows

\[
\text{Cov}_y(k) = p^2 \text{Cov}_\nu(k) + 1(k = 0)p(1 - p)
\]
(6)

\[
\text{Cov}_{\nu y}(k) = p \text{Cov}_\nu(k)
\]
(7)

where we have used the identity \( \text{Cov}_\nu(k) = \text{Cov}_N(k) \). We are now in position to derive the Wiener filter. First, we write \( S_y(z) \) as

\[
S_y(z) = \sigma G(z) G(z^{-1})
\]
(8)

where \( \sigma \) is a constant. This operation is called the canonical factorization of the power spectrum of \( \{y_n\}_n \). The function \( G(z) \) is the part of \( S_y(z) \) that has all its zeros and poles lying in the unit disk. The function \( 1/G(z) \) is the transfer function of the whitening filter: it transforms \( \{y_n\}_n \) into a white noise process with variance \( \sigma \).

Next, we form the ratio \( S_{\nu y}(z)/G(z^{-1}) \). This ratio is interpreted as the transfer function of a linear filter. The impulse response of this filter has values at the left and at the right of the time origin which means that this filter is non-causal. In order to have a causal filter, we should isolate the part of the impulse response having values at the right of the time origin solely, and compute its transfer function. This can be done by expanding \( S_{\nu y}(z)/G(z^{-1}) \) into fractions and by considering only the fractions with zeros and poles in the unit disk. In other words, we transform the non-causal filter into a causal one. We denote the transfer function of the causal version of the filter by

\[
H(z) = \left[ \frac{S_{\nu y}(z)}{G(z^{-1})} \right]_+
\]

The transfer function of the optimal filter is given by [12]

\[
H_o(z) = \frac{H(z)}{\sigma G(z)}.
\]

It remains to invert this transfer function back into the 
time domain to find the desired recurrence between \( \hat{\nu}_n \) and

---

1. Observe from [3] and [4] that both \( S_y(z) \) and \( S_{\nu y}(z) \) are well-defined for \( |z| = 1 \) under the assumption [4].
y_n and, subsequently, between the non-centered variables \( \tilde{N}_n \) and \( Y_n \). This procedure is illustrated in Section 3.1 for the case that the underlying model is the \( M/M/\infty \) queue.

### 3.1 Application to the \( M/M/\infty \) model

In light of the results reported in Section 4, all what we have to do is to find expressions for \( S_y(z) \) and \( S_{vy}(z) \). This can easily be done when the underlying model is the \( M/M/\infty \) queueing model, as shown below.

Let us first determine \( S_y(z) \). By using (11) and (13) together with the property that \( \text{Cov}_N(k) = \text{Cov}_v(k) \), we find

\[
\text{Cov}_y(k) = \begin{cases} \rho^2 \gamma r^{(k)}, & k \neq 0 \\ \rho \gamma, & k = 0 \end{cases}
\]

Since \( \gamma < 1 \) and \( |z| = 1 \), the z-transform of \( \text{Cov}_y(k) \) is

\[
S_y(z) = \frac{\rho \gamma (p-1)z^2 + (1 + \gamma^2(1 - 2p))z + \gamma(p-1)}{z(1 - \gamma z)(1 - \gamma z^{-1})}.
\]

The second-order polynomial in the variable \( z \) in the numerator has two positive real roots given by \( r \) and \( 1/r \), with

\[
r = \frac{1 + \gamma^2(1 - 2p) - \sqrt{(1 - \gamma^2)(1 - \gamma^2(1 - 2p)^2)}}{2\gamma(1 - p)}.
\]

Note that \( r < 1 \). Hence

\[
S_y(z) = \frac{\gamma \rho p(1 - p)}{r} \left\{ \frac{(1 - rz)(1 - rz^{-1})}{(1 - \gamma z)(1 - \gamma z^{-1})} \right\}
\]

\[
= \sigma G(z) G(z^{-1})
\]

with

\[
\sigma := \frac{\gamma \rho p(1 - p)}{r} \quad \text{and} \quad G(z) := \frac{1 - rz^{-1}}{1 - \gamma z^{-1}}.
\]

We now compute \( S_{vy}(z) \). From (17) and (13) we find

\[
\text{Cov}_{vy}(k) = \rho \gamma r^{(k)}
\]

so that

\[
S_{vy}(z) = \frac{\rho \gamma (1 - \gamma^2)}{(1 - \gamma z)(1 - \gamma z^{-1})}.
\]

The transfer function \( H(z) \) is given by

\[
H(z) = \left[ S_{vy}(z) \right] \frac{G(z^{-1})}{G(z)} = \frac{\rho \gamma (1 - \gamma^2)}{(1 - \gamma r)(1 - \gamma r^{-1})}
\]

and the transfer function \( H_o(z) \) of the optimal filter takes here the simple form

\[
H_o(z) = \frac{\rho \gamma (1 - \gamma^2)}{(1 - \gamma r)(1 - rz^{-1})} = \frac{B}{1 - Az^{-1}}
\]

where

\[
A = r, \quad B = \frac{\rho \gamma (1 - \gamma^2)}{\sigma(1 - \gamma r)} = \frac{r(1 - \gamma^2)}{(1 - p)(1 - \gamma r)}.
\]

The impulse response of this linear filter is given by the first-order recurrence relation [12]

\[
\hat{\nu}_n = A\hat{\nu}_{n-1} + BY_n + \rho(1 - A - pB).
\]

It is interesting to compare this filter with the Kalman filter derived in [8]. They appear to be the same! This result is somehow expected, since both the Kalman filter and the Wiener filter are optimal (among the class of linear filters) in the sense that they minimize the mean square error. The key point is that the Kalman filter used in [8] was derived under a heavy traffic assumption, while the Wiener filter computed in the present paper holds for any value of the model parameters \( \lambda \) and \( \mu \). This partly explains why the estimator in [8] behaves well under light or moderate traffic as experimentally observed in that paper.

We conclude this section by computing the mean square error \( \epsilon_{min} := \mathbb{E}[\left( \bar{N}_n - \bar{N}_n \right)^2] \) of our estimator. It is known that [12]

\[
\epsilon_{min} = \sum_{k=1}^{M} \text{Res} \left[ F(z), z_k \right]
\]

with

\[
F(z) := \frac{1}{z} \left[ S_v(z) - H_o(z)S_{vy}(z^{-1}) \right]
\]

where \( z_1, \ldots, z_M \) are the poles (if any) of the function \( F(z) \) lying in the unit disk. The notation \( \text{Res} \left[ F(z), z_k \right] \) stands for the residue of \( F(z) \) at point \( z = z_k \), namely, the coefficient of \( 1/(z - z_k) \) in the Laurent series expansion of \( F(z) \) in the vicinity of \( z_k \).

Specializing \( F(z) \) to the values of \( S_v(z), S_{vy}(z) \) and \( H_o(z) \) found earlier, yields

\[
F(z) = \frac{\rho(1 - \gamma^2)((1 - Bp)z - A)}{(1 - \gamma z)(z - \gamma)(z - A)}
\]

This function has two poles inside the unit circle which are located at \( z = A \) and \( z = \gamma \); the residues of \( F(z) \) at these poles are given by \( -\rho p A B (1 - \gamma^2)/(1 - \gamma)(A - \gamma) \) (2). Recall that a Kalman filter is the optimal filter under the condition of linear dynamics and observation, which does not hold in our case. However, the dynamics does converge to a linear diffusion as the traffic load tends to infinity, allowing us in [8] to obtain a Kalman filter which is optimal for the asymptotic heavy traffic regime.
and \( \rho(1 + pB \gamma/(A - \gamma)) \), respectively. Summing up these residues gives
\[
\epsilon_{min} = \rho \left( 1 - \frac{Bp}{1 - \gamma A} \right).
\]
By using the expressions of \( A \) and \( B \), we finally obtain
\[
\epsilon_{min} = \rho \left( 1 - \frac{1 - \gamma^2 + \sqrt{(1 - \gamma^2)(1 - \gamma^2(1 - 2p)^2)}}{2\gamma^2} \right).
\]
This expression for \( \epsilon_{min} \) can be used to tune the parameters \( p \) and \( \gamma \) or equivalently \( S \) (see Section 5).

### 4 Optimal first-order linear filter

The theory reported in Section 3 applies to any on-time distribution \( \Psi(x) \) (with the exception of heavy-tailed distributions). It is not easy to identify the function \( G(z) \) that appears in the canonical factorization of the spectrum \( S_{\nu}(z) \) (see Section 3) and thereby the optimal filter, except when the on-times are exponentially distributed rvs (see Section 3.1).

In this section we will determine the first-order linear filter that minimizes the mean square error. Observe that, unlike the Wiener filter, the proposed approach will not return the optimal filter among all linear filters but simply the optimal linear filter among all first-order linear filters. We will illustrate this approach at the end of this section in the case where \( \Psi(x) \) is an hyperexponential distribution.

Recall the definition of the centered processes \( \{\nu_n\}_n \), \( \{\tilde{\nu}_n\}_n \) and \( \{y_n\}_n \) made at the beginning of Section 3.

The methodology is simple: we want to find constants \( A \in (0, 1) \) and \( B \) such that \( \epsilon := E[(\nu_n - \tilde{\nu}_n)^2] \) is minimized when the process \( \{\tilde{\nu}_n\}_n \) satisfies the following first-order recurrence relation
\[
\tilde{\nu}_n = A\tilde{\nu}_{n-1} + By_n.
\]
In steady-state this implies that
\[
\tilde{\nu}_n = B \sum_{k=0}^{\infty} A^k y_{n-k}.
\]

The mean square error \( \epsilon \) is equal to
\[
\epsilon = E[\tilde{\nu}_n^2] + E[\nu_n^2] - 2E[\tilde{\nu}_n \nu_n].
\]
We have \( E[\tilde{\nu}_n^2] = E[(N_n - \rho)^2] = \rho \) (see Section 2). From (12) and (7) we find
\[
E[\tilde{\nu}_n \nu_n] = pB \sum_{k=0}^{\infty} A^k \text{Cov}_\nu(k) = pBg(A)
\]
where
\[
g(z) := \sum_{k=0}^{\infty} z^k \text{Cov}_\nu(k).
\]
The power series \( g(z) \) converges for \( |z| < 1 \) (Hint: \( k \to \text{Cov}_\nu(k) \) is nonincreasing) and is therefore differentiable for \( |z| < 1 \). We will denote by \( g'(z) \) its derivative.

It remains to express \( E[\tilde{\nu}_n^2] \) in terms of the parameters \( A \) and \( B \). Squaring both sides of (11) and then taking the expectation, yields
\[
E[\tilde{\nu}_n^2] = \left( \frac{B}{1 - A^2} \right) \left( 2AE[\nu_{n-1}y_n] + BE[y_n^2] \right).
\]
With the identities \( E[y_n^2] = \text{Cov}_\nu(0) = \rho p \) (see (6)) and \( E[\nu_{n-1}y_n] = Bp^2 (g(A) - \rho)/A \) (Hint: use (12), (7) and \( \text{Cov}_\nu(0) = \rho \), we obtain
\[
E[\tilde{\nu}_n^2] = \left( \frac{pB^2}{1 - A^2} \right) (2pg(A) + \rho(1 - 2p)).
\]
Finally, the mean square error is
\[
\epsilon = \rho - 2pBg(A) + \left( \frac{pB^2}{1 - A^2} \right) (2pg(A) + \rho(1 - 2p)).
\]

In order to minimize \( \epsilon \), \( A \in (0, 1) \) and \( B \) must be the solution of the following system of equations:
\[
\begin{align*}
\frac{\partial \epsilon}{\partial A} &= \frac{2pB}{1 - A^2} \left( AB \left[ \frac{2pg(A) + \rho(1 - 2p)}{1 - A^2} \right] \\
&\quad + g'(A)(pB - (1 - A^2)) \right) = 0 \\
\frac{\partial \epsilon}{\partial B} &= 2p \left( B \left[ \frac{2pg(A) + \rho(1 - 2p)}{1 - A^2} \right] - g(A) \right) = 0.
\end{align*}
\]
The 2nd equation gives
\[
B = \frac{g(A)(1 - A^2)}{2pg(A) + \rho(1 - 2p)}.
\]
Substituting this value of \( B \) into the 1st equation shows that \( A \) must satisfy
\[
Ag(A)(2pg(A) + \rho(1 - 2p)) - g'(A)(1 - A^2)(pg(A) + \rho(1 - 2p)) = 0.
\]
If this equation has a unique solution \( A \in (0, 1) \), then substituting this value of \( A \) into (15) will give the optimal pair \((A, B)\).

It is shown in Appendix A that (16) has always a unique solution in \([0, 1)\) (in particular) if \( g'(x) > 0 \) in \([0, 1)\). This condition will hold as long as \( P(D > S) > 0 \). In practice, one can always select \( S \) such that this condition is true.

The reader can check that the filter defined in (11) with the optimal pair \((A, B)\) is the same as the Wiener filter found in Section 3.1 when the on-times are exponentially distributed.

We now illustrate the approach developed in this section by considering the situation where on-times have a
hyperexponential distribution. More precisely, we assume that
\[ \Psi(x) = 1 - \sum_{l=1}^{L} p_l e^{-\mu_l x} \] (17)
with \( 0 < p_l < 1 \), \( l = 1, 2, \ldots, L \), and \( \sum_{l=1}^{L} p_l = 1 \). In this
setting the underlying queuing model can be seen as
\( L \) independent \( M/M/\infty \) queues in parallel. The arrival
rate to queue \( l \) is \( \mu_l \) and the service rate is \( \mu_l \). Define
\( \gamma_l := \exp(-\mu_l) \), \( \rho_l := p_l \mu_l/\mu_l \) so that \( \rho = \sum_{l=1}^{L} \rho_l \).
The autocovariance function of the process \( \{\nu_n\}_n \) is equal to
\[ \text{Cov}_{\nu}(k) = \begin{cases} \rho_l & \text{for } k = 0 \\ \sum_{l=1}^{L} \rho_l \gamma_l & \text{for } k \neq 0 \end{cases} \]
so that
\[ g(A) = \sum_{l=1}^{L} \frac{\rho_l}{1 - A^{\gamma_l}}. \]

**Numerical example**: \( L = 2 \), \( p = 0.0106 \) and \( S = 2.5s \).
Also
\begin{align*}
1/\mu_1 &= 3897.7 \quad \rho_1 = 19.5 \quad \gamma_1 = 0.999359 \\
1/\mu_2 &= 4800.6 \quad \rho_2 = 75.1 \quad \gamma_2 = 0.999995 \\
1/\mu &= 1831.6 \quad \rho = 94.7 \\
\end{align*}
The optimal first-order filter is
\[ \hat{N}_n = 0.99879456 \hat{N}_{n-1} + 0.10720289 Y_n + 0.006540864. \]
For comparison the Wiener filter found in Section 3.1 (for
exponential on-times) is
\[ \hat{N}_n = 0.99828589 \hat{N}_{n-1} + 0.14885344 Y_n + 0.012900081. \]

**5 Guidelines on choosing ACKs parameters**

A “good” pair \((p, S)\) should (i) limit the feedback implosion
while at the same time (ii) achieve a good quality of the estimator.
Of course (i) and (ii) are antinomic and therefore a trade-off must be found. This trade-off will be formalized as follows: we want to select a pair \((p, S)\)
so that the mean number of ACKs generated every \( S \) seconds (see (15)) and the relative error of the variance of the estimator (denoted as \( \eta \)) are bounded from above by given constants, namely
\[ \begin{cases} E[Y_n] = p \rho \leq \alpha \\ \eta = \frac{\text{Var}(N_n) - \text{Var}(\hat{N}_n)}{\text{Var}(N_n)} \leq \beta. \end{cases} \] (18)

When \( \hat{N}_n \) is optimal then \( \text{Var}(N_n) - \text{Var}(\hat{N}_n) = \text{E}[\{(N_n - \hat{N}_n)^2\}] \) and \( \eta \) becomes the “normalized mean square error”

\[ \text{E}[Y_n] = p \rho \leq \alpha \]

with \( \epsilon_{\text{min}} \) given in (10).

For given constants \( \alpha \) and \( \beta \), it is easy to solve the constrained
optimization problem defined in (13), provided that \( \eta \) is known.

For the \( M/M/\infty \) model, where \( \epsilon_{\text{min}} \) is given in (10),
we find that \( p = \alpha/\rho \) and that \( S \), or equivalently \( \gamma \), is the
unique positive solution of the equation \( \epsilon_{\text{min}} = \rho \beta \).

The problem now is to choose constants \( \alpha \) and \( \beta \) so that
conditions (i) and (ii) are satisfied. We have found that \( \alpha \) in the
range \([0.5, 1]\) and \( \beta \leq 0.15 \) give satisfactory results.

We conclude this section with general remarks on how to adapt
the parameters \( p \) and \( S \) to important variations in the
membership. The estimation schemes in Sections 3.1 and 4 can
still be used if \( p \) and/or \( S \) change over time, provided that these modifications do not prevent the system to be most of the time in steady-state. In that setting, a new filter will have to be recomputed after each modification (e.g. each time the number of ACKs collected by the source in a given period is significantly different from \( \rho p \), the current expected number of ACKs).

**6 Validation with real traces**

In this section we apply the estimators developed in Sections 3.1 and 4 to four traces corresponding to video multicast sessions. Two types of estimators will be used: the estimator – denoted as \( \hat{N}_n^F \) – found in (10) when the dynamics of population is modeled as an \( M/M/\infty \) queue; the estimator – denoted as \( \hat{N}_n^H_2 \) – derived in Section 4 in case the join times are Poisson and the on-times have a 2-stage hyperexponential distribution \( (M/H_2/\infty) \) model.

The objective is twofold: we want to investigate the quality of both estimators when compared to real life conditions, and we want to identify the best one.

We have collected four MBone traces – denoted \( \text{video}_i \), \( i = 1, \ldots, 4 \) – between August 2001 and September 2001 using the MListen tool [1]. Each trace corresponds to a long-lived video session (see duration of each session in Table 1) where the superscript “d” stands for “days”). We have run both algorithms (estimators) on each trace.

For each trace we have identified the parameters of the \( M/M/\infty \) model (parameters \( \lambda \) and \( \mu \), or equivalently parameters \( \rho \) and \( \mu \)) and of the \( M/H_2/\infty \) model (parameters \( \rho, \mu_1, \mu_2, p_1 \) and \( p_2 = 1 - p_1 \) – see definitions in Section 3.1). The values of these parameters are reported in columns 3–8 in Table 1. Details on how these values have been obtained are given in Appendix B.

[1] The values of the parameters come from the trace called \text{video}_1 investigated in Section 6.
Table 1: Parameter identification

<table>
<thead>
<tr>
<th>Trace</th>
<th>Session lifetime</th>
<th>$\rho$</th>
<th>$1/\mu_1$</th>
<th>$1/\mu_2$</th>
<th>$p_1$</th>
<th>$p_2$</th>
<th>$p$</th>
<th>$S$</th>
<th>$\alpha$</th>
<th>$\beta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>video01</td>
<td>3$^e$ 13$^e$ 33$^m$ 20$^a$</td>
<td>94.7</td>
<td>18316</td>
<td>3897</td>
<td>480061</td>
<td>0.97</td>
<td>0.03</td>
<td>0.011</td>
<td>2.5</td>
<td>1.0</td>
</tr>
<tr>
<td>video02</td>
<td>11$^d$ 1$^b$ 46$^d$ 8$^s$</td>
<td>14.1</td>
<td>16476</td>
<td>1</td>
<td>226498</td>
<td>0.93</td>
<td>0.07</td>
<td>0.034</td>
<td>3.2</td>
<td>0.5</td>
</tr>
<tr>
<td>video03</td>
<td>50$^d$ 22$^d$ 13$^m$ 20$^a$</td>
<td>8.1</td>
<td>66823</td>
<td>1</td>
<td>900854</td>
<td>0.93</td>
<td>0.07</td>
<td>0.062</td>
<td>20.0</td>
<td>0.5</td>
</tr>
<tr>
<td>video04</td>
<td>29$^d$ 16$^b$ 43$^m$ 13$^s$</td>
<td>17.9</td>
<td>83390</td>
<td>1</td>
<td>473268</td>
<td>0.82</td>
<td>0.18</td>
<td>0.028</td>
<td>10.0</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Parameters $p$ and $S$ have been chosen by following the guidelines presented in Section 5 namely $\alpha \in \{0.5, 1\}$ and $\beta \in \{0.1, 0.15\}$. Values of these parameters are listed in columns 9–10 in Table 1. The performance of estimators $N^E$ and $N^H_2$ are reported in Tables 2 and 3.

Table 2 reports several order statistics (columns 3–7) and the sample mean of the relative error achieved on trace $\text{video}_0$ (see (column 2)). For instance, the estimator is uniformly better than the other over an entire session than the theoretical mean square error. It is given by $(10)$ and by $\epsilon$ for $N^H_2$ (see (13)). The expected average $E[N_n - \tilde{N}_n]$ is zero in both approaches. Both estimators $N^E$ and $N^H_2$ have almost no bias (see column 2), and their empirical variances closely match the theoretical ones given by $\epsilon_{\text{min}}$ and $\epsilon$, respectively. It is of interest to point out that for the 4 traces studied, $\epsilon$, the theoretical mean square error provided by $N^H_2$, is smaller than $\epsilon_{\text{min}}$, the theoretical mean square error provided by $N^E$ (however, this result is reversed if we consider the empirical mean square errors). Thus, $N^E$ is more efficient than $N_n$ (again, $N^E$ is empirically more efficient than $N^H_2$).

Table 3: Empirical mean and variance of $N_n - \tilde{N}_n$

<table>
<thead>
<tr>
<th>Trace</th>
<th>Mean</th>
<th>25</th>
<th>50</th>
<th>75</th>
<th>90</th>
<th>95</th>
<th>$\epsilon_{\text{min}}$</th>
<th>$\epsilon$</th>
<th>$\eta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>video01</td>
<td>N^E</td>
<td>6.82</td>
<td>1.09</td>
<td>2.42</td>
<td>5.25</td>
<td>11.5</td>
<td>19.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>N^H_2</td>
<td>6.12</td>
<td>1.08</td>
<td>2.55</td>
<td>6.31</td>
<td>13.5</td>
<td>20.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>video02</td>
<td>N^E</td>
<td>4.19</td>
<td>1.41</td>
<td>3.08</td>
<td>5.43</td>
<td>8.66</td>
<td>11.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>N^H_2</td>
<td>4.12</td>
<td>0.98</td>
<td>2.14</td>
<td>4.41</td>
<td>8.78</td>
<td>12.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>video03</td>
<td>N^E</td>
<td>4.20</td>
<td>1.55</td>
<td>3.26</td>
<td>5.71</td>
<td>8.71</td>
<td>11.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>N^H_2</td>
<td>3.98</td>
<td>1.07</td>
<td>2.36</td>
<td>4.83</td>
<td>9.35</td>
<td>12.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>video04</td>
<td>N^E</td>
<td>3.79</td>
<td>1.23</td>
<td>2.57</td>
<td>4.51</td>
<td>7.50</td>
<td>11.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>N^H_2</td>
<td>4.06</td>
<td>1.02</td>
<td>2.21</td>
<td>4.39</td>
<td>8.98</td>
<td>14.7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>overall</td>
<td>N^E</td>
<td>4.44</td>
<td>1.33</td>
<td>2.88</td>
<td>5.22</td>
<td>8.60</td>
<td>12.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>N^H_2</td>
<td>4.34</td>
<td>1.02</td>
<td>2.26</td>
<td>4.73</td>
<td>9.61</td>
<td>14.2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The last column provides the relative error on $\text{Var}(\tilde{N}_n)^E$, called $\eta = \epsilon_{\text{min}} / \rho$ in Section 5.

In Fig. 1 (resp. 2, 3 and 4) we plot the variations of membership for session $\text{video}_0$ (resp. $\text{video}_2$, $\text{video}_3$ and $\text{video}_4$), together with the estimates returned by $N^E$ and $N^H_2$. Among all 4 sessions, session $\text{video}_0$ presents the highest variations in $N_n$. Fig. 1(a) (resp. 2(a), 3(a) and 4(a)) displays three curves: the membership of the video session, the estimation returned by $N^E$, labeled “Exponential”, and the estimation returned by $N^H_2$, labeled “Hyperexponential”. It is clearly visible, especially at the left-hand side of graph 1(a), that $N^E$ tracks better the session dynamics than $N^H_2$. Both estimators $N^E$ and $N^H_2$ have been derived under some specific and restrictive assumptions: Poisson join times for both of them, exponential (resp. 2-stage hyperexponential) on-times for the first (resp. second) one. It is interesting to know...

---

4 An estimator is said to be more efficient if it has a smaller variance.
whether or not these assumptions were violated in each session video\textsubscript{i}, \(i = 1, \ldots, 4\). We have therefore carried out a statistical analysis of each trace in order to determine the nature of their join time process and of their on-time sequence.

As shown in Table 4 and Figs. 1, 2, 3 and 4, parts (b) and (c), neither is the join time process Poisson nor are the on-times exponentially distributed (or hyperexponentially distributed) for any of the traces. The inter-join times and the on-times appear to follow subexponential distributions (Weibull and Lognormal distributions), a situation quite different from the assumptions under which the estimators have been obtained. Despite these significant differences, the estimators behave well and therefore show a good robustness to assumption violations.

In summary, both estimators perform very well when applied to real traces and are robust to significant deviations from their (theoretical) domain of validity. Estimator \(\hat{N}^H_\infty\) returns the best global performance for the relative error criterion, but does not track high fluctuations as well as \(\hat{N}^F_n\). Overall, we have found that \(\hat{N}^F_n\) is a good estimator, both in terms of its performance and its usability since it only requires the knowledge of two parameters: \(\rho\) and \(\mu\).

### 7 Open issues

The main pending issue concerns the knowledge of parameters \(\rho\) and \(\mu\) (or equivalently any couple of parameters among \(\rho\), \(\lambda\) and \(\mu\), since \(\rho = \lambda/\mu\) in steady-state). When these parameters are not known, the source should estimate them. Again, the source could estimate any two parameters among \(\rho\), \(\lambda\) and \(\mu\) and infer the third one.

One possible way of estimating \(\lambda\) is to let a newly arrived receiver send a “heartbeat” to the source with a certain (constant) probability \(q\) (\(q\) should be small enough to avoid overwhelming the source with heartbeats). The source would then use the arrival time \(t_m\) of the \(m\)th heartbeat to estimate \(\lambda\). The maximum likelihood estimator is \(\hat{\lambda} = m/(qt_m)\). This estimator is unbiased and consistent by the strong law of large numbers (\(\lim_{m \to \infty} t_m/m = 1/(q\lambda)\)).

In a similar way, the source can estimate \(\mu\) if receivers probabilistically send a “goodbye” message reporting their lifetime when they leave the session. Let \(\tau_m\) be the lifetime indicated in the \(m\)th goodbye message received at the source, then the maximum likelihood estimator of \(\mu\) is simply \(\hat{\mu} = n/(\sum_{i=1}^{m} \tau_m)\). The estimator \(\hat{\mu}\) is unbiased and consistent.

A natural estimator for \(\rho\) is \(\hat{\rho} = E[\hat{N}_n]\). As long as there is no estimation of both \(\rho\) and \(\mu\), it is not possible to compute the filter coefficient \(A\) and \(B\). Then only a naive estimator for \(N_n\) can be used, defined as the ratio of the number of ACKs received \(Y_n\) over the ACK probability \(p\) (this estimator does not perform well as shown in [2]). Observe that \(E[Y_n/p] = \rho\).

One might want to treat the estimation of \(N_n\), \(\mu\) and \(\rho\) as a joint parameter estimation problem. Unfortunately, the problem becomes much more complicated, and we are not able at the moment to say anything about its resolution. Alternatively, one might want to replace \(\rho \approx \hat{\rho} = E[\hat{N}_n]\) in (9) and investigate on the performance of the resulting algorithm. It is possible to go even farther by replacing \(\rho\) by \(Y_n/p\), instead of using \(E[Y_n/p]\). This will add more burstiness to the dynamics of \(\hat{N}_n\) and (9) becomes

\[
\hat{N}_n = A\hat{N}_{n-1} + (1 - A)\frac{Y_n}{p}.
\]

The latter equation is nothing but an exponential weighted moving average (EWMA) estimator for the membership process \(N_n\). We expect this EWMA estimator to perform better than \(\hat{N}_n\), as given in (9), over highly dynamic sessions, as its auto-regressive equation has been derived by replacing the constant term \(\rho\) by the bursty term \(Y_n/p\). It is also expected that this EWMA estimator will react faster to important and sudden changes in the membership process. However, its performance over real sessions still needs to be investigated more carefully. Observe that the use of this EWMA estimator relies on the prior knowledge of \(\gamma\) = exp(\(-\mu\delta\)) solely, and therefore either “goodbye” or “hello” messages are still required to compute \(\hat{\mu}\).

The first option is to directly estimate \(\mu\) as indicated above, but the first estimate will be delayed until the reception of the first goodbye message, which might necessitates the departure of several customers. The second option is to estimate both \(\rho\) and \(\lambda\) as indicated earlier in the section and to estimate \(\mu\) as \(\hat{\lambda}/\hat{\rho}\), which is not expected to perform as well as the first option, but will return a first estimate much faster. At the expense of a larger warm-up period, we believe that the first option is
preferable. Another advantage of this option concerns the possible modification of $g'$ by the source, in order to control the volume of goodbye messages sent, which clearly cannot be done with the hello messages.

8 Conclusion

The major contribution of this work is the design of two novel estimators for evaluating the membership in multicast sessions. We have designed estimators capable of efficiently tracking the dynamics of multicast sessions while simultaneously avoiding feedback implosion. In contrast to the estimator proposed in [3] which was designed under heavy traffic assumptions, our schemes do not place such restrictions.

Relying on the Wiener filter theory, we have computed the optimal linear estimator for session membership when the underlying model is an $M/M/\infty$ queue. The optimality refers to the unbiasedness of the estimator and to the fact that the mean square error is minimized. We have also developed the optimal first-order linear filter in case the on-time distribution is arbitrary and have derived the associated estimator in case the on-times have a 2-stage hyperexponential distribution.

Both obtained estimators have been validated on real traces. Their performance have been shown to be excellent, one of them showing a good ability to adapt to highly dynamic multicast sessions.

9 Acknowledgments

The authors wish to thank Prof. Patrick Thiran for helpful suggestions.

A Existence and uniqueness of the solution

Lemma A.1 Define

$$ f(x) := (2pg(x) + \rho(1 - 2p))xg(x) - (pg(x) + \rho(1 - 2p))(1 - x^2)g'(x), $$

where $g(x)$ is given in (13).

If $g'(x) > 0$ for $x \in [0, 1]$, then $f(x)$ has a unique zero in $[0, 1]$.

Proof. Write $f(x)$ as $f(x) = f_+(x) - f_-(x)$ where

$$ f_+(x) = [2p(g(x) - \rho) + \rho]xg(x), $$

$$ f_-(x) = [p(g(x) - \rho) + \rho(1 - p)](1 - x^2)g'(x). $$

The derivative of $f_-(x)$ is given by $f'_-(x) = -\alpha x^2 - \beta x + \alpha$

using the derivation

$$ \alpha := p(g'(x))^2 + [p(g(x) - \rho) + \rho(1 - p)]g''(x), $$

$$ \beta := 2[p(g(x) - \rho) + \rho(1 - p)]g'(x). $$

Since $g'(x) > 0$ and $g(x) > \rho$ (see (13)), it is seen that $\alpha > 0$ and $\beta > 0$ which implies that $f''_-(x) = -2\alpha x - \beta < 0$ for $x \in [0, 1)$, we therefore have that $f'_-(x)$ is strictly decreasing in $[0, 1)$, with $f'_-(0) = \alpha > 0$ and $f'_-(1) = -\beta < 0$. Thus, the function $f'_-(x)$ has only one zero in $[0, 1)$.

Therefore, and under the assumptions of the lemma, it is seen that:

(i) $f_+(x)$ is continuous and strictly increasing in $[0, 1)$ with $f_+(0) = 0$;

(ii) $f_-(x)$ is continuous in $[0, 1)$ and $f_-(1) = 0$. There exists $x_0 \in (0, 1)$ such that $f_-(x_0)$ is strictly increasing in $(x_0, 1)$ and $f'_-(x_0) = 0$.

We deduce from the above that $f(x)$ has a unique zero in $[0, 1)$ if $g'(x) > 0$ in $[0, 1)$. This condition will hold as long as $P(D > S) > 0$. In practice, one can always select $S$ such that this condition is true.

B Computing parameters from trace

Each trace records $(T_i, D_i), i \geq 1$. To use the $M/M/\infty$ queue model, we identify $1/\lambda = E[T_{i+1} - T_i]$ and $1/\mu = E[D]$, and deduce $\rho = \lambda/\mu$. To use the $M/H_2/\infty$ queue model, $\lambda$ is computed as before. Identifying $\mu_1, \mu_2, p_1$ and $p_2$ requires the knowledge of the first three moments of $D$ (recall that $p_2 = 1 - p_1$). For a 2-stage hyperexponential distribution, the $k$th moment is given by

$$ E[D^k] = \sum_{i=1}^{2} \frac{p_i k!}{(\mu_i)^k} = k! \left( \frac{p_1}{(\mu_1)^k} + \frac{p_2}{(\mu_2)^k} \right), \quad k \geq 1. $$

The parameters $\mu_1, \mu_2, p_1$ and $p_2$ are then solution to the following system of four equations, where $\sigma_i$ stands for $1/\mu_i$ with $l = 1, 2$.

$$ p_1 + p_2 = 1 \quad (19) $$
$$ p_1 \sigma_1 + p_2 \sigma_2 = E[D] \quad (20) $$
$$ p_1 \sigma_1^2 + p_2 \sigma_2^2 = E[D^2]/2 \quad (21) $$
$$ p_1 \sigma_1^3 + p_2 \sigma_2^3 = E[D^3]/6 \quad (22) $$

Equations (19) and (20) readily give

$$ p_1 = \frac{\sigma_2 - E[D]}{\sigma_2 - \sigma_1}, \quad p_2 = \frac{E[D] - \sigma_1}{\sigma_2 - \sigma_1}. \quad (23) $$

Substituting Equations (23) for $p_1$ and $p_2$ into (21) yields

$$ E[D^2]/2 = E[D](\sigma_2 + \sigma_1) - \sigma_1 \sigma_2. \quad (24) $$

Substituting them into (22) yields

$$ E[D^3]/6 = E[D]((\sigma_2 + \sigma_1)^2 - \sigma_1 \sigma_2) - \sigma_1 \sigma_2(\sigma_2 + \sigma_1). \quad (25) $$
Introduce now $S_\sigma$ and $P_\sigma$ as the sum and the product of $\sigma_1$ and $\sigma_2$, respectively. Equations (24) and (25) become

$$E[D^2]/2 = E[D]S_\sigma - P_\sigma$$
$$E[D^3]/6 = E[D](S_\sigma^2 - P_\sigma) - P_\sigma S_\sigma$$

where the latter identity is obtained when using the first one. It then follows that

$$S_\sigma = \frac{3E[D][E[D^2] - E[D^3]]}{3(2E[D]^2 - E[D^3])},$$
$$P_\sigma = \frac{6(2E[D]^2 - E[D^3])}{6(2E[D]^2 - E[D^3])},$$

and $\sigma_1 = 1/\mu_1$ and $\sigma_2 = 1/\mu_2$ are the (positive) solutions of $x^2 - S_\sigma x + P_\sigma = 0$. Namely,

$$\sigma_{1,2} = 1/2 \times \left(S_\sigma \pm \sqrt{(S_\sigma)^2 - 4P_\sigma}\right).$$

We now can compute $p_1$ and $p_2$ as given in Equations (23). It is then possible to calculate $p_l = p_0\lambda/\mu_l$ and $\gamma_l = \exp(-\mu_lS)$ for $l = 1, 2$. Last $\rho = p_1 + p_2$.

References


(a) Membership evolution of long-lived session $video_1$: $\rho = 94.67, \ p = 0.0106, \ S = 2.5s$.

(b) inter-join times
(c) on-times

Exponential fit
Lognormal fit
Lognormal linear fit for inter-join times, $\mu = 5.20, \ d = 1.68$

Weibull fit
Weibull linear fit for on-times, shape 0.26, scale 1400

Figure 1: Membership estimation of session $video_1$ and corresponding probability plots

(a) Membership evolution in long-lived session $video_2$: $\rho = 14.138, \ p = 0.034, \ S = 3.2s$.

(b) inter-join times
(c) on-times

Lognormal linear fit for inter-join times, $\mu = 3.38, \ d = 1.49$

Weibull linear fit for on-times, shape 0.345, scale 3700

Figure 2: Membership estimation of session $video_2$ and corresponding probability plots
Figure 3: Membership estimation of session video3 and corresponding probability plots.

(a) Membership evolution in long-lived session video3; \( \rho = 8.12, \ p = 0.0616, \ S = 20s. \)

(b) inter-join times

(c) on-times

Weibull linear fit for inter-join times, shape 0.65, scale 3500

Exponential fit

Weibull fit

Exponential fit

Weibull fit

Lognormal fit

Weibull linear fit for on-times, shape 0.18, scale 4000

Exponential fit

Weibull fit

Lognormal fit

Figure 4: Membership estimation of session video4 and corresponding probability plots.

(a) Membership evolution in long-lived session video4; \( \rho = 17.92, \ p = 0.02787, \ S = 10s. \)

(b) inter-join times

(c) on-times

Weibull linear fit for inter-join times, shape 0.55, scale 2700

Exponential fit

Weibull fit

Exponential fit

Weibull fit

Lognormal fit

Weibull linear fit for on-times, shape 0.18, scale 4400

Exponential fit

Weibull fit

Lognormal fit

13