Modelling of Ethernet Traffic on Multiple Timescales

Patrik Carlsson, Markus Fiedler and Arne A. Nilsson *

Abstract

Ethernet is one of the most common link layer technologies, used in local area networks, wireless networks and wide area networks. There is however a lack of traffic models for Ethernet that is usable in performance analysis.

In this paper we use such a model. The model operates on matching multiple moments of the bit rate at several timescales. In order to match the model parameters to measured traffic, five methods have been developed.

We use this to model three different links; the BCpOct89 Bellcore trace, an Internet access link and an ADSL link. Our results show that, as the number of sources present on an Ethernet link grows, the model becomes better and less complex.

Keywords

Ethernet Traffic Model, Multi-Timescale, Multifractal, Bit Rate, Moments, Fluid Flow Analysis, Model Matching, Measurement.

1 Introduction

During the last decade an increasing amount of research showed that data traffic exhibits self-similar properties, and recently there are also indications that it is multifractal. During the same period many offices and residential areas became networked using Ethernet technology. Internet became a part of many businesses and private users' daily routine. In this environment, each link is more or less unique and requires detailed analysis in order to find a suitable model and to capture its parameters.

A model proposed by Mannersalo, Norros and Riedi [MNR02] forms the base for the model that we use. Their model exhibit multifractal properties:

> In its simplest form our model is based on the multiplication of independent rescaled stochastic processes $\Lambda^{(i)}(\cdot) \triangleq \Lambda(b^i)$ which are piecewise constant. ... In multiplying rather than adding re-scaled versions of a 'mother' process we obtain a process with novel properties which are best understood not in an additive analysis, but in a multiplicative one. Processes emerging from multiplicative construction ... exhibit typically a 'spiky' appearance [MNR02].

*Blekinge Intitute of Technology, School of Engineering, Dept. of Telecommunication Systems, {Patrik.Carlsson, Markus.Fiedler, Arne.Nilsson}@bth.se

The mother- or base process is a Markov-modulated rate process (MMRP) with two activity states. We initially used this model in [CF00] to perform fluid flow analysis. The model used here is a slighly modified version of the Mannersalo, Norros and Riedi model, modified in such a manner that we have no requirement on the scaling between the independent processes, nor do we require an infinite number of processes.

The model uses the link layer bit rate as the network parameter to match. The bit rate is a parameter that is easy to estimate (for both fixed and varying frame sizes) and it is available for all technologies and layers. To capture the scaling behaviour, the model matches the statistical moments of the bit rate at several timescales.

The outline for this paper is the following; In Section 2 we describe how we measure the network in order to obtain the parameters of interest, followed by a description of the model that we use in Section 3. In Section 4 we give crude descriptions of the matching methods, followed by a set of matching examples based on real-network traffic in Section 5. Conclusions and outlook are discussed in Section 6.

2 Measurements and Moment Estimation

The goal with the measurements is to estimate the bit rate with as few errors as possible. The analysing software operates off-line on files that contain packet traces This allows for almost arbitrary sample rates. These files can be constructed from conversion or capturing. Conversion is used when traces are already available, for instance from TCPDUMP [TC]. Capturing is done when it is possible to install a measurement point (MP), for details see [Car03a]. The MP is the key component in a passive measurement infrastructure (PMI). We use passive measurements in order to get a complete and undistorted view of the properties of the measured link.

Depending on how the traces were obtained, either from conversion or capture, the accuracy of the timestamp is determined by the original capturer. For instance, traces from DAG cards have an accuracy of less than 100 ns [END], whereas traces from PCAP have an accuracy in the order of ms to μs depending on the system that ran PCAP [LW91, MDG01, Don02]. SNMP could have been used, but the accuracy of SNMP based network measurements is not sufficient to capture short timescales [CFT+02]. Based on the timestamp accuracy the sample rate is determined, with regards to the accepted level error in the bit rate estimation. The base scale is determined by the sample rate as $t_0 = 1/F_s$.

2.1 Bit rate Estimation

The bit rate B_i in interval i is calculated as the number of bits that have arrived in sample interval i, divided by the sample interval duration T_s.

$$B_i = \frac{b_0 + \sum_{k=1}^{N} b_k + b_{N+1}}{T_s} \tag{1}$$

Here, b_0 are the bits belonging to interval i from a frame that started arriving prior to this interval. Similarly, b_{N+1} identify the bits of a frame that started arriving in this interval but was not completed. b_k are the bits of frames that were completed within the interval. See Figure 1 for an example. The sample interval T_s is determined by the desired base scale t_0.

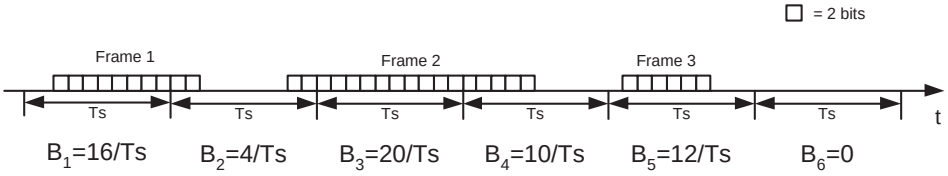

Figure 1: Estimation of the bit rate.

Table 1: Percentage of Error in Bit rate estimations, w.r.t. timestamp accuracy and sample interval.

T_S	$T_A = 100$ ns	$T_A = 10$ μs	$T_A = 1$ ms	$T_A = 10$ ms	$T_A = 1$ s
1 s	10^{-5}	10^{-3}	10^{-1}	1	10^{2}
1 ms	10^{-2}	1	10^{2}	10^{3}	10^{5}
1 μs	10	1000	10^{5}	10^{6}	10^{8}

Since all measurement equipments, both hardware and software, have a fixed timestamp accuracy, there will be errors in the bit rate estimation. The size of the error is related to the accuracy of the timestamp, T_A, and the sample interval, T_S. The timestamp accuracy specifies how much a frame can be shifted, with regards to the timestamp. A frame can at most be shifted T_A seconds. In the worst case, this can cause $T_A \times C$ bits to be placed in incorrect interval(s). Thus, a rough estimate of the error is given by Error $= \frac{T_A C}{T_S C} = \frac{T_A}{T_S}$. In Table 1 a list of the error is given in percent, based on some typical timestamp accuracies. This error sets the lower limit on the timescales that we can use. Given a trace obtained from a DAG card, the base scale should not be smaller than 10 μs; for a PCAP-based trace, a lower bound of 1 ms applies.

2.2 Moment Estimation

Based on the bit rate estimations we can evaluate the moments at different timescales. To do this we need to supply the desired base scale t_0 which decides $t_0 = T_s$, as well as a scaling factor. For simplicity, we have chosen to use the same scaling factor S, which is an integer larger or equal to 2. This factor specifies the distance between timescales $t_i = S \times t_{i-1}$, i.e. timescale i consists of S samples from timescale $i - 1$.

Let us define X_k as the bit rate samples at the k^{th} timescale. Then the i^{th} moment at the k^{th} timescale, $\gamma_{i,k}$, is calculated from

$$\gamma_{i,k} = \frac{1}{M_k} \sum_{j=1}^{M_k} (X_{k,j})^i \tag{2}$$

where M_k is the number of samples that are present in X_k. This is formed from the samples

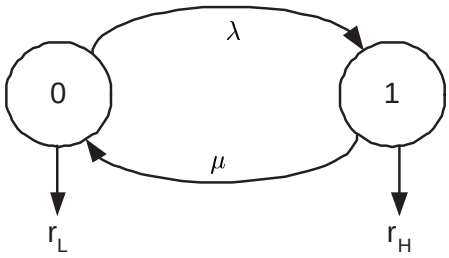

Figure 2: State diagram for a 2-state Markov Modulated Rate Process.

of the lower timescales

$$X_{k,j} = \frac{1}{S} \sum_{m=(j-1)S+1}^{jS} X_{k-1,m} \tag{3}$$

with X_0 being the base scale obtained from the measurements. See [Car03b] for more details and examples.

3 Process Description

The process is formed by multiplying the output of N independent sub-processes. Denote the output from sub-process i at time t with $R_i(t)$. The output from the process will be:

$$R(t) = \prod_{i=0}^{N-1} R_i(t) \quad \text{[bps]} \tag{4}$$

$R_0(t)$ can be considered to be modulated by other (unit-less) sub-processes $R_i(t), i \geq 1$. The key item is that $R(t)$ will be in bps. Since the process is formed by independent sub-processes, a detailed knowledge of the sub-process properties and parameters is desirable, hence a detailed analysis will follow.

3.1 Sub-process Analysis

The sub-processes are 2-state Markov-Modulated Rate Processes (MMRPs), each having four parameters: λ and μ are the transition rates in-between the states, r_L is the output rate in the 'low' state 0 and r_H is the output when the process is in the 'high' state 1. In Figure 2 a state diagram is shown. Using this notation, we can express the transition matrix $\mathbf{M}$ and rate matrix $\mathbf{R}$ for sub-process i as:

$$\mathbf{M}_i = \begin{bmatrix} -\lambda_i & \lambda_i \\ \mu_i & -\mu_i \end{bmatrix} \qquad \mathbf{R}_i = \begin{bmatrix} r_{L,i} & 0 \\ 0 & r_{H,i} \end{bmatrix}$$

We define the cycle time as $\tau_i = 1/\lambda_i + 1/\mu_i$. This time specifies the mean time that it takes for the sub-process to go from one state to the other, and back to the original state. Thus, for a

sub-process with a large cycle time the state changes will be rather infrequent, whereas a small cycle time indicates a sub-process with frequent state changes.

In [Car03b] we give a detailed analysis for the first three moments, and their behavior at the limits. Here we simply show the analytical expressions for them:

$$\mathbf{E}\left[R_i^1(T)\right] = \frac{r_{L,i}\mu_i + r_{H,i}\lambda_i}{\lambda_i + \mu_i} \tag{5}$$

$$\mathbf{E}\left[R_i^2(T)\right] = \frac{2\lambda_i\mu_i(r_{L,i} - r_{H,i})^2}{(\lambda_i + \mu_i)^3}\left(\frac{1}{T} + \frac{e^{-(\lambda_i+\mu_i)T} - 1}{(\lambda_i + \mu_i)T^2}\right) + \frac{(r_{L,i}\mu_i + r_{H,i}\lambda_i)^2}{(\lambda_i + \mu_i)^2} \tag{6}$$

$$\mathbf{E}\left[R_i^3(T)\right] = \frac{6C}{\lambda_i + \mu_i}\left(\frac{1}{T^2} - \left(\frac{1 - e^{-(\lambda_i+\mu_i)T}}{(\lambda_i + \mu_i)T^3}\right)\right) + \frac{3D}{\lambda_i + \mu_i}\frac{1}{T} + \frac{6F}{(\lambda_i + \mu_i)^3}\left(\frac{1 - (\lambda_i T + \mu_i T + 1)e^{-(\lambda_i+\mu_i)T}}{T^3}\right) + \frac{G^3}{(\lambda_i + \mu_i)^3} \tag{7}$$

where

$$\begin{aligned}
A &= (r_{i,L}^3\mu_i + \lambda_i r_{i,H}^3) \\
B &= (2\mu_i^2 r_{i,L}^3 + 2\mu_i\lambda_i r_{i,L}^2 r_{i,H} + 2\lambda_i r_{i,L} r_{i,H}^2\mu_i + 2\lambda_i^2 r_{i,H}^3) \\
C &= \frac{A}{(\lambda_i + \mu_i)^2} - \frac{2B}{(\lambda_i + \mu_i)^3} + \frac{3G^3}{(\lambda_i + \mu_i)^4} \\
D &= \frac{B}{(\lambda_i + \mu_i)^2} - \frac{2G^3}{(\lambda_i + \mu_i)^3} \\
F &= \frac{B}{(\lambda_i + \mu_i)^2} - \frac{A}{\lambda_i + \mu_i} - \frac{G^3}{(\lambda_i + \mu_i)^3} \\
G &= r_{i,L}\mu_i + \lambda_i r_{i,H}
\end{aligned}$$

3.2 The Process

Once we have obtained a set of sub-processes, we combine these to form the process. That will be matched to the measured data. The process is formed by multiplying the output from N sub-processes. The transition matrix is formed by using Kronecker addition:

$$\mathbf{M}_\mathrm{D} = \mathbf{M}_1 \oplus \mathbf{M}_2 \oplus \cdots \oplus \mathbf{M}_N \tag{8}$$

and the rate matrix as

$$\mathbf{R}_\mathrm{D} = \mathbf{R}_1 \odot \mathbf{R}_2 \odot \cdots \odot \mathbf{R}_N \tag{9}$$

See the Appendix in [Car03b] for a short description of Kronecker algebra and the $\odot$ operator.

3.3 Moments

Since the process is formed by multiplication, the moment analysis becomes very simple: We multiply the moments of the independent sub-processes in order to obtain the moments for the process.

$$\mathbf{E}\big[R^1(T)\big] = \prod_{i=1}^{N} \mathbf{E}\big[R_i^1(T)\big] \tag{10}$$

$$\mathbf{E}\big[R^2(T)\big] = \prod_{i=1}^{N} \mathbf{E}\big[R_i^2(T)\big] \tag{11}$$

$$\mathbf{E}\big[R^3(T)\big] = \prod_{i=1}^{N} \mathbf{E}\big[R_i^3(T)\big] \tag{12}$$

4 Process Matching

We have developed five matching methods. These are used to select the sub-process parameters, in such a manner as to minimize the difference (error) between the measured data and the process. One method is a manual method, i.e. the matching is done by manually selecting the parameter values. The second method uses a Genetic Algorithm (GA), the third one uses Simulated Annealing (SA), and the last two are heuristic methods. The first, HM1, iterates through each available timescale and tries to find suitable parameters. The second one, HM2, does an initial crude match using the edge timescales, and if necessary it applies a simulated annealing algorithm to this crude match. A detailed description and performance evaluation of the five methods is found in [Car03b].

Fitness Function

Each method creates a proposed solution, based on the measured data and other criteria (i.e. number of sub-processes, boundary values etc). The quality of a proposed solution is evaluated through a fitness function. The fitness function compares the proposed solutions' moments to the desired moments at the different timescales, where the desired moments are obtained from measurements. The fitness value is evaluated for each moment and timescale individually, and then combined to obtain the total fitness of the solution. The fitness of a solution is evaluated as follows:

1. Based on the proposed solutions parameter matrix, $\mathbf{P}$, calculate the moment matrix $\mathbf{S}$ using the analytical expressions for the moments. The $\mathbf{S}$ matrix rows are based on the measured moments at the observed time intervals, $\widehat{\gamma}_{i,k}$ is the i^{th} moment at the k^{th} timescale. $\mathbf{P}$ holds the sub-process parameters with one sub process per row. If the method is allowed to use K sub-processes, and the measured data contains N_m moments and N_T timescales, then the $\mathbf{P}$ and $\mathbf{S}$ matrices look like this :

$$\mathbf{P} = \begin{bmatrix} \widehat{\lambda}_1 & \widehat{\mu}_1 & \widehat{r}_{L,1} & \widehat{r}_{H,1} \\ \widehat{\lambda}_2 & \widehat{\mu}_2 & \widehat{r}_{L,2} & \widehat{r}_{H,2} \\ \cdots & \cdots & \cdots & \cdots \\ \widehat{\lambda}_K & \widehat{\mu}_K & \widehat{r}_{L,K} & \widehat{r}_{H,K} \end{bmatrix} \tag{13}$$

$$\mathcal{F}(\mathbf{P}) \rightarrow \mathbf{S} = \begin{bmatrix} \widehat{\gamma}_{1,0} & \widehat{\gamma}_{2,0} & \cdots & \widehat{\gamma}_{n,0} \\ \widehat{\gamma}_{1,1} & \widehat{\gamma}_{2,1} & \cdots & \widehat{\gamma}_{n,1} \\ \cdots & \cdots & \cdots & \cdots \\ \widehat{\gamma}_{1,N_T-1} & \widehat{\gamma}_{2,N_T-1} & \cdots & \widehat{\gamma}_{N_m,N_T-1} \end{bmatrix} \tag{14}$$

A matrix $\mathbf{D}$ containing the measured moments is constructed in the same way as $\mathbf{S}$. $\mathbf{D}_{i,j}$ identifies the desired value on the i:th timescale and j:th moment, similarly $\mathbf{S}_{i,j}$ identifies the proposed solution.

$$\mathbf{D} = \begin{bmatrix} \gamma_{1,0} & \gamma_{2,0} & \cdots & \gamma_{n,0} \\ \gamma_{1,1} & \gamma_{2,1} & \cdots & \gamma_{n,1} \\ \cdots & \cdots & \cdots & \cdots \\ \gamma_{1,N_T-1} & \gamma_{2,N_T-1} & \cdots & \gamma_{N_m,N_T-1} \end{bmatrix} \tag{15}$$

2. Calculate the error matrix as:

$$\mathbf{U} = |\mathbf{D} - \mathbf{S}|$$

From this the fitness for each timescale, i, and moment, j, is obtained as:

$$F_{i,j} = \begin{cases} \frac{1}{U_{i,j}} & U_{i,j} > 1 \\ \Delta \cdot U_{i,j} + a & 10^{-L} < U_{i,j} \leq 1 \\ F_{\text{max}} & U_{i,j} \leq 10^{-L} \end{cases} \tag{16}$$

where L is the accuracy requirement or fitness levels. The value 10^{-L} specifies how small the error has to be in order to have a fitness of F_{max}. Δ and a specify a line that goes from $(10^{-L}, F_{\text{max}})$ to $(1, 1)$, simply calculated as

$$\Delta = \frac{F_{\text{max}} - 1}{10^{-L} - 1}$$

$$a = F_{\text{max}} - \Delta \cdot 10^{-L}$$

F_{max} is the maximum fitness that a moment can obtain in a single timescale, currently set to 100, see Figure 3 for a visualization of the shape of the fitness function. The reason not to use a linear error-to-fitness conversion is the following: When an error goes towards zero the fitness will grow rapidly, particular once the error is smaller than one. Such a value will dominate the total fitness value, i.e. if we for one particular timescale and moment found an almost perfect match with an error $U_{i,j} < 10^{-9}$, the corresponding fitness value will be very large (10^9). Such a fitness value will dominate the total fitness, even if the other fitness values are very small. This is why we have introduced an upper fitness level F_{max} and not a linear error-to-fitness relation.

3. Calculate the fitness for the individual moments as the weighted sum of the fitness at the different timescales:

$$\vec{F} = [F_1, F_2, \ldots, F_{N_m}] \quad \text{where} \quad F_i = \sum_{j=0}^{N_T-1} h_j F_{i,j} \tag{17}$$

The weights h_j enable us to control the influence a particular timescale has on the total fitness of the moment. For instance, the larger timescales are formed from a smaller

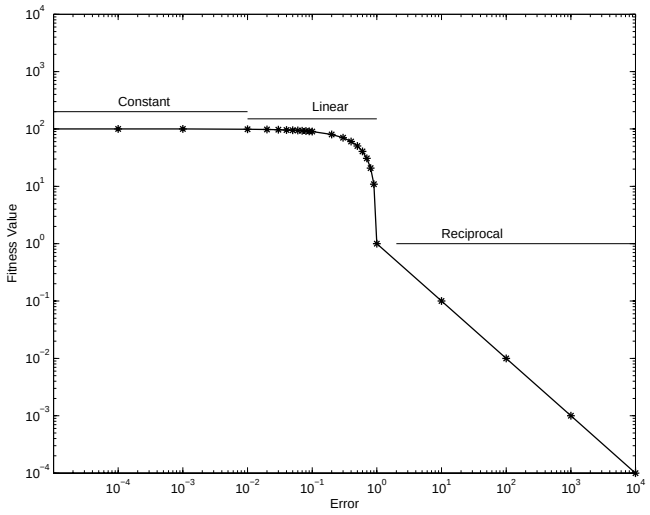

Figure 3: Shape of the fitness function, $L = 2$. (Note the logarithmic scales.)

number of samples than the small timescales. Hence, their values might be less accurate, and should be weighted accordingly when added to the others. $\vec{H}$ contains the weights for the individual timescales, $\sum h_j = 1$.

4. Calculate the total fitness F_t as the weighted sum of the fitness of the moments;

$$F_t = \sum_{j=1}^{N_m} g_j F_j \quad \text{where} \quad \vec{G} = [g_1, g_2, \ldots, g_{N_m}] \tag{18}$$

$\vec{G}$ is a vector containing the weights for each moment, $\sum g_j = 1$. This gives us the same capabilities as for the timescale weighting, hence we can prioritze the first moment over the second that has priority over the third.

5. Return the fitness value F_t.

By manipulating the fitness weights we can give preferential treatment to particular moments and timescales.

5 Results

Here we apply the model to three different links. The first is one of the classical Bellcore traces [BCT]. The second is a link that functions as Internet access for approximately 300 users. The third provides Internet access to a single user via an ASDL modem. The Bellcore trace has an timestamp accuracy of about 10 μs. Hence we use a base scale of 1 ms in order to keep the bit rate estimation error within one percent. The IAL and ADSL traces were obtained using MPs

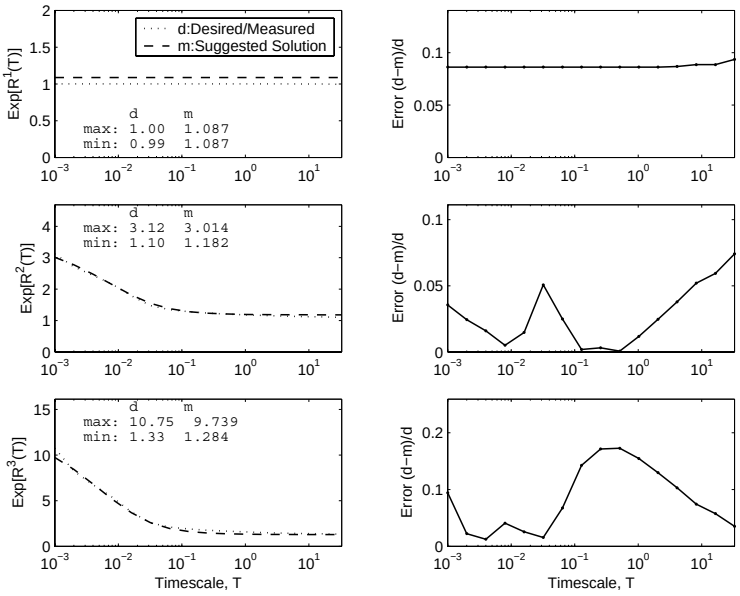

Figure 4: BCpOct89 Model with a fitness of 88.7681 obtained by the HM2 algorithm.

with DAG3.5E cards. We can use a base scale of 10 μs while maintaining the same level of error. For all three traces we use a scale factor of two, $S = 2$.

Out of the four Bellcore traces, we have selected the BCpOct89 trace, this contains primarily LAN traffic. The BCpOct89 trace started at 11:00 October 5 1989 EDT and ended approximately 30 minutes later. The Internet access link (IAL) connects to a student dormitory network with approximately 300 users. The link is a full-duplex 100Base-T, and interconnects a media converter (100Base-X to 100Base-T) with a 100Base-T switch port. We refer to this link as IAL. We performed several measurements at this location, but here we present the measurement we called IAL-1 since it exhibits an interesting shape. The IAL-1 started on 17:10:49 September 17, 2003 and it ran for one hour. Here we focus on the traffic from the network to the Internet. The ADSL link is located in an apartment and connects a broadband router to an ADSL modem using 10Base-T. The router has four switch ports and a wireless access point (IEEE 803.11b). In normal operations, there are between two and three hosts attached to the switch, but no wireless hosts. We ran one measurement on the link, spanning eight hours. The measurement called ADSL8, ran from 09:30 to 17:30 September 14, 2003. During this period two hosts were active, one of which received streaming audio for the majority of the measurement. Here we focus on the traffic going from the router to the modem.

The best model for the Bellcore trace was obtained by the HM2 algorithm, see Figure 4. It obtained a fitness of 88.77, using only two sub-processes. For the IAL-1 the best solution was obtained by the SA method, it reached a fitness of 75.5 using three sub-processes, see Figure 5. Finally the manual method performed best ADSL link, obtaining a fitness of 34.4 using twelve sub-processes, see Figure 6.

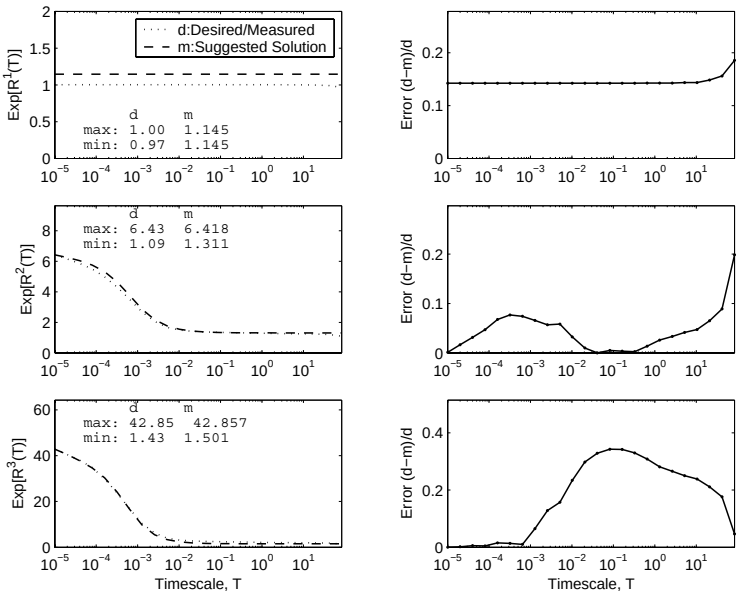

Figure 5: IAL-1-NI Model, this solution was obtained by SA with a fitness of 75.5058.

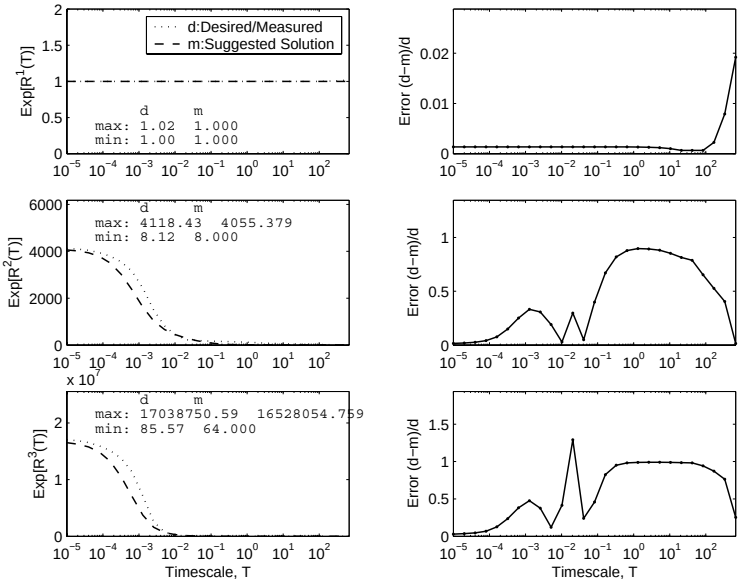

Figure 6: Manual methods suggested solution to the ADSL8RM trace, the fitness is 34.3949.

6 Conclusions

In this paper we have described a model for modelling on Ethernet on multiple timescales. We applied this model to three different Ethernet links. The conclusion with regards to these links, is that the more sources that are present on the link, the better and less complex the model becomes.

In general, the model has two constraints: the number of sub-processes needed, and the number of moments to match. The number of sub-processes is a constraint set by the computer running the fluid flow analysis, about 10 for a 512 MB RAM system. Some traces are less suitable for modelling all three moments, for instance traces that indicate the possible presences of a long range-dependent process, or traces that only contain a small number of sources. How to model this is left for further study.

To model the small timescales found on an Ethernet, TCPDUMP is insufficient. In order to do measurements at the these timescales, the equipment needs to have a timestamp accuracy that is in the same order as the bit time on the measured link.

References

[BCT] Bellcore traces. Available from World Wide Web: `http://ita.ee.lbl.gov/html/contrib/BC.html` [checked 2 June, 2004].

[Car03a] P. Carlsson. Measurement point: Operational description and specification, 2003. Work in progress, please contact the author.

[Car03b] P. Carlsson. Multi-timescale modelling of ethernet traffic, 2003. Available from World Wide Web: `http://www.bth.se/fou` [checked 1 June, 2004].

[CF00] P. Carlsson and M. Fiedler. Multifractal products of stochastic processes: Fluid flow analysis. In *Proceedings of the 15th Nordic Teletraffic Seminar (NTS-15), Lund*, 2000.

[CFT+02] P. Carlsson, M. Fiedler, K. Tutschku, S. Chevul, and A. Nilsson. Obtaining reliable bit rate measurements in SNMP-managed networks. In *Proceedings of the 15th ITC Specialist Seminar, Würzburg*, 2002.

[Don02] S. Donnelly. *High Precision Timeing in Passive Measurements of Data Networks*. PhD thesis, The University of Waikato, 2002.

[END] Dag card specifications. Available from World Wide Web: `http://www.endace.com/products.htm` [checked 2 June, 2004].

[LW91] W. E. Leland and D. V. Wilson. High time-resolution measurement and analysis of lan traffic: Implications for lan interconnection. In *Proceedings of IEEE Infocom*, pages 1360–1366, 1991.

[MDG01] J. Micheel, S. Donnelly, and I. Graham. Precision timestamping of network packets. In *Proceedings of ACM Sigcomm Internet Measurement Workshop*, 2001.

[MNR02] P. Mannersalo, I. Norros, and R. Riedi. Multifractal products of stochastic processes: Construction and some basic properties. *Applied Probability Trust*, November 2002.

[TC] Tcpdump public repository. Available from World Wide Web: `http://www.tcpdump.org` [checked 2 June, 2004].