```
longlong enc(int k, int p[ ])
{int i, j, t; longlong c, r;

if (p[k – 1] < k) return 0;
i = 0;
/* skip null binomial coefficients */
while (p[i] < i + 1){i = i + 1;}
r = 1; t = p[i];
/* first nonzero binomial coefficient */
j = 1;
while (j <= i + 1)
      {r = (r * t)/j; t = t – 1; j = j + 1;}

c = r;
while (i < k – 1)
   {t = p[i] + 1; i = i + 1; r = (r * t)/(i + 1);
      while (t < p[i])
         {t = t + 1; r = (r * t)/(t – i –1);}
      c = c + r;}
   return c;
}

void dec(longlong c, int k, int p[ ])
{int i, digit; longlong r;

   if (c < 1){p[k – 1] = k – 1;}
   else
      {r = 1; digit = k;
      /* get rightmost digit counting up */
      while (r <= c){digit = digit + 1;
                     r = (r * digit)/(digit – k);}
      r = (r * (digit – k))/digit; c = c – r;
      digit = digit – 1; p[k – 1] = digit;}
   i = k – 2;
   /* get the other digits counting down from previous */
   while (i >= 0 && c > 0)
      {r = (r * (i + 2))/digit; digit = digit – 1;
       while (r > c)
         {r = (r * (digit – i – 1))/digit; digit = digit – 1;}
      c = c – r; p[i] = digit; i = i – 1;}
   /* set digits for null binomial coefficients, if any */
   while (i >= 0){p[i] = i; i = i – 1;}

return;
}
```

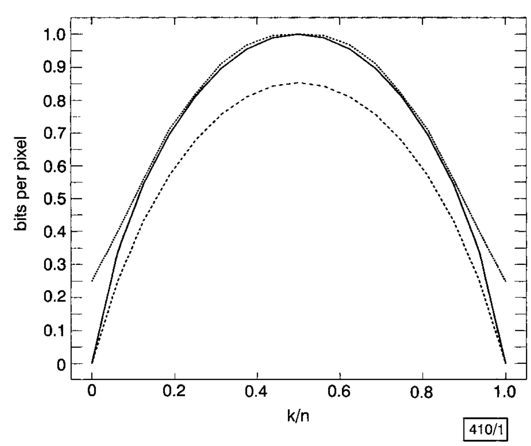

Fig. 1 *Graphs of functions L_{Huff}, L_{ent} and L_{comb} for $n = 16$ and $0 \le k/n \le 1$*

——— entropy
·········· Huffman code
– – – – combinatorial code

Coding performance: We now present a comparison of the bit rate of our method with bit rates that could be obtained using a classical lossless compression algorithm. A k-subset of an n-set can be described using a memoryless channel model that outputs a word of length n using two symbols, say a 1 if the element of the corresponding rank is in the subset and a 0 otherwise. The probabilities of the two symbols would be $p_1 = k/n$ and $p_0 = 1 - p_1 = (n - k)/n$, respectively. The noiseless coding theorem [4] assures us that the minimum average code length per symbol (bit) that could be achieved with a memoryless coding scheme, $L_{ent}(n, k)$ is given by the binary entropy

$$L_{ent}(n,k) = \frac{n-k}{n}\log_2\frac{n}{n-k} + \frac{k}{n}\log_2\frac{n}{k} \qquad (4)$$

As our coding scheme uses $\log_2\binom{n}{k}$ bits to encode any k-subset, the corresponding code length per bit would be

$$L_{comb}(n,k) = \frac{1}{n}\log_2\binom{n}{k} \qquad (5)$$

The combinatorial scheme outperforms any memoryless coding scheme because it assigns codes only to bit combinations that correspond to k-subsets; it can be shown that for any nontrivial pair (n, k) (that is, $k \neq 0$, $k \neq n$) $L_{comb}(n, k) < L_{ent}(n, k)$. Furthermore, $L_{ent}(n, k)$ is still a theoretical limit; any practical implementation of a memoryless coding scheme may give a slightly greater average code length. Fig. 1 shows the graphs of $L_{comb}(n, k)$, $L_{ent}(n, k)$ and $L_{Huff}(n, k)$, which is the average codelength per bit of a Huffman encoder using 4 bit words.

Conclusions: We have presented a practical method for compactly representing subsets of coefficients for signal compression schemes. A given compression ratio is imposed and it is assumed that the retained coefficients can be located arbitrarily. The lack of a model about the distribution of the retained coefficients is both a strength and a limitation: it provides a convenient framework when no model can be established but may fail to fully take advantage of the existing correlations implied by a given model. Our method of coding provides the theoretically minimal codelength under the given assumptions, requires no book-keeping and uses only integer arithmetic. It can be easily extended to cope with large datasets that would not be representable within the available register range of a given computer.

22 February 2000
Electronics Letters Online No: 20000566
DOI: 10.1049/el:20000566

D.C. Popescu and D. Bone (*CSIRO Division of Mathematical and Information Sciences, GPO Box 664, Canberra, ACT 2601, Australia*)

References

1 SWELDENS, W.: 'The lifting scheme: A new philosophy in biorthogonal wavelet constructions', *Proc. SPIE*, 1995, **2569**, pp. 68-79 (in: 'Wavelet applications in signal and image processing')
2 STOLLNITZ, et al.: 'Wavelets for computer graphics' (Morgan Kaufmann, San Francisco, 1997)
3 LEHMER, D. H.: 'The machine tool of combinatorics' *in* 'Applied combinatorial mathematics' (John Wiley & Sons, New York, 1964)
4 ROMAN S.: 'Coding and information theory' (Springer-Verlag, 1992)

Superposition of multiplicative multifractal traffic processes

J. Gao and I. Rubin

Telecommunications network systems have been observed to contain processing and switching nodes that are loaded by highly bursty traffic streams. Such flows have been noted to exhibit long-range-dependent (LRD) properties. The design and sizing of electronic processing and switching systems critically depends on the features of such processes. LRD traffic streams are modelled as a multiplicative multifractal process. It is proven that superposition of a finite number of multiplicative multifractal traffic streams results asymptotically in another multifractal process. Noting that traffic processes driving network nodes result from the superposition of multiple traffic streams at the input ports of the node, this property makes the multiplicative multifractal traffic model a versatile tool for modelling LRD traffic flows at different locations and levels across a network.

Introduction: Recent analysis of high-quality traffic measurements have revealed the prevalence of long-range-dependent (LRD) features in traffic processes loading packet switching communications networks. Included are local area networks (LANs) [1], wide area networks (WANs) [2], variable-bit-rate (VBR) video traffic [3], and world wide web (WWW) traffic [4]. Hence, it is essential to have a relatively simple model for the effective design and sizing of telecommunications buffering, switching and processing systems.

It has been shown that a multiplicative multifractal traffic model can yield similar queueing performance results simultaneously for a network operating at low, medium and high utilisation levels when one compares the system-size tail performance of a single server queueing system driven on one hand by a multiplicative multifractal traffic model and on the other hand by the measured traffic streams such as flows across local and metropolitan area networks, involving a multitude of applications such as VBR video and WWW oriented services [5 – 7]. Quantitative understanding of why a multiplicative multifractal process makes a good LRD traffic model has also been obtained [8]. In communication networks, the traffic process loading nodal switching and transmission processors can be described as the superposition of multiple input traffic streams. Hence, a fundamental question that must be answered is the following: does superposition of multiplicative multifractal traffic streams result in another multifractal stream? In this Letter, we use the counting process model [6] to prove that the answer to the above question is positive. This result makes the multiplicative multifractal traffic model a versatile tool for the modeling of LRD traffic flows at different locations and levels across a network.

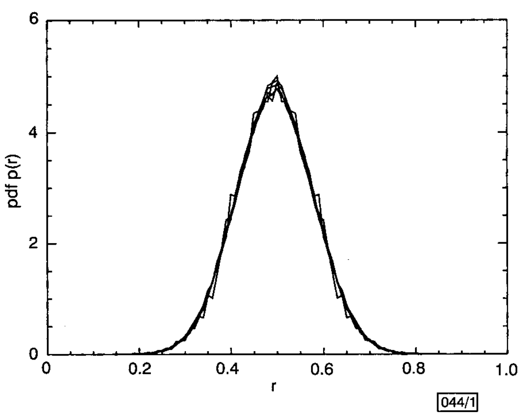

Fig. 1 *Multiplier distributions P(r) against r curves for stage numbers N = 13, ... 17*

Multiplicative multifractal: We consider a unit interval and associate it with a unit mass. We divide the unit interval into two (e.g. left and right) segments of equal length. We also partition the mass into two fractions, r and $1 - r$, and assign them to the left and right segments, respectively. The parameter r, called the multiplier, is in general a random variable, governed by a probability density function (pdf) $P(r)$, $0 \leq r \leq 1$. Each new subinterval and its associated weight (or mass) are further divided into two parts following the same rule. Note the scale (i.e. the interval length) associated with stage i is 2^{-i}. We assume that $P(r)$ is symmetric about $r = 1/2$, and has successive moments $\mu_1, \mu_2, \ldots$. Hence, the weights at the stage N, $\{w_n(N), n = 1, \ldots, 2^N\}$, can be expressed as $w_n(N) = u_1 u_2 \ldots u_N$, where u_l, $l = 1, \ldots, N$ are independent identically distributed (i.i.d) random variables having the pdf $P(r)$. When $w_n(N)$ is interpreted as the loading to a network (representing the total count of message units) in a time slot of length $2^{-N}T$, where T is the total time period concerned, then this process becomes a counting traffic process model. The multifractality of the multiplicative process refers to the fact that $M_q(\varepsilon) = E(\Sigma_{n=1}^{2N} (w_n(N))^q) \sim \varepsilon^{\tau(q)}$ with $\varepsilon = 2^{-N}$, $\tau(q) = -\ln(2\mu_q)/\ln 2$ [8].

Superposition of multiplicative multifractals: We consider the superposition of (an arbitrary) k independent multiplicative multifractal traffic streams. We let these multifractal traffic streams be denoted as $MF_1, \ldots, MF_k$. Their multiplier distributions are $P^{(1)}(r), \ldots, P^{(k)}(r)$. These distributions are assumed to be symmetric about 1/2, and have successive moments $\mu_q^{(i)}$, $i = 1, \ldots, k$, $q = 1, 2, \ldots$. The superimposed traffic stream is denoted by $MF^{(sk)}$, $MF^{(sk)} = \Sigma_{i=1}^{k} \lambda_i \cdot MF_i$, with $0 < \lambda_1, \ldots, \lambda_k < 1$, $\Sigma_{i=1}^{k} \lambda_i = 1$. A weight $w^{(sk)}(N)$ of $MF^{(sk)}$ at the stage N can then be expressed as $w^{(sk)}(N) = \Sigma_{i=1}^{k} \lambda_i u_1^{(i)} \ldots u_N^{(i)}$, where $u_j^{(i)}, j = 1, \ldots, N$ are i.i.d random variables governed by the pdf $P^{(i)}(r)$, for $i = 1, \ldots, k$.

Lemma 1: $E[(w^{(sk)}(N))^q] = E[(\Sigma_{i=1}^{k} \lambda_i u_1^{(i)} \ldots u_N^{(i)})^q] = l[\mu_q^{(sk)}]^N[1 + \Sigma_{j=0}^{q'} b_j y_j^N]$, where l, $\mu_q^{(sk)}$, $\{b_j, y_j, |y_j| < 1, j = 0, \ldots, q'\}$, and q' are suitable constants.

Proof of lemma 1:

$$E\left[\left(\sum_{i=1}^{k} \lambda_i u_1^{(i)} \cdots u_N^{(i)}\right)^q\right] =$$

$$E\left[\sum \frac{q!}{q_1! \cdots q_k!} \left(\lambda_1 u_1^{(i)} \cdots u_N^{(1)}\right)^{q_1} \cdots \left(\lambda_k u_1^{(k)} \cdots u_N^{(k)}\right)^{q_k}\right]$$

$$= \sum \frac{q!}{q_1! \cdots q_k!} \left[\lambda_1^{q_1} \left(\mu_{q_1}^{(1)}\right)^N\right] \cdots \left[\lambda_k^{q_k} \left(\mu_{q_k}^{(k)}\right)^N\right]$$

where $\Sigma_{i=1}^{k} q_i = q$. We let

$$\mu_q^{(sk)} = \max_{q_1, \ldots, q_k} \left(\mu_{q_1}^{(1)} \cdots \mu_{q_k}^{(k)}\right)$$

We assume that, among all the terms $\mu_{q_1}^{(1)} \ldots \mu_{q_k}^{(k)}$, there are m terms that attain the above maximal value, and group those terms together. We can then write $E[(w^{(sk)}(N))^q] = l[(\mu_q^{(sk)}]^N[1 + \Sigma_{j=0}^{q'} b_j y_j^N]$ with $l = l(m)$, $\mu_q^{(sk)}$, $\{b_j = b_j(m), y_j, |y_j| < 1, j = 0, \ldots, q'\}$, and $q' = q'(m)$ are suitable constants. This concludes the proof.

It is interesting to note that, because of the non-negativeness of variances, we have $\mu_2^{(sk)} = \max(\mu_2^{(1)}, \ldots, \mu_2^{(k)})$. Since the Hurst parameter for an ideal multiplicative multifractal process is given by $-\frac{1}{2} \log_2 \mu_2$ [8], we thus observe that the Hurst parameter for $MF^{(sk)}$ is the same as the source traffic stream with the largest second moment for the multiplier distribution.

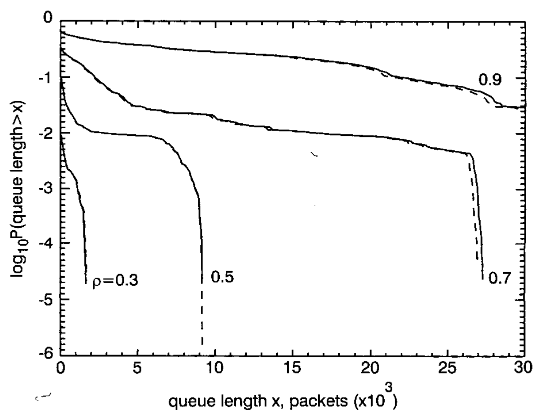

Fig. 2 *System-size tail distributions obtained when $MF^{(s2)}$ (dashed lines) and MF^e (solid lines) are used to drive identical single server queueing systems*

Utilisation levels are indicated on the Figure
– – – – $MF^{(s2)}$
——— MF^e

Theorem 1: $MF^{(sk)}$ is (asymptotically in N) a multifractal.

The proof is quite straightforward if it is noted from lemma 1 that $M_q(\varepsilon) = 2^N E[(w^{(sk)}(N))^q] \sim \varepsilon^{\tau(q)}$, with $\varepsilon = 2^{-N}$, $\tau(q) = -\ln(2\mu_q^{(sk)})/\ln 2$, when $N \to \infty$.

We can prove that the moments of the multiplier distribution of $MF^{(sk)}$ are (asymptotically in N) stage-independent. Hence, it is to be expected that $MF^{(sk)}$ will for most practical situations be a multiplicative multifractal. In fact, we have observed in our numerical simulations that $MF^{(sk)}$ is asymptotically a multiplicative process. To illustrate this feature, we present an example involving the superposition of two multiplicative multifractal traffic streams, MF_1 and MF_2, with the largest stage number being 18. Let $\lambda_1 = \lambda_2 = 1/2$, and the multiplier distributions for MF_1 and MF_2 be truncated Gaussian: $P(r) \sim e^{-\alpha(r-1/2)^2}$, $0 \leq r \leq 1$, with $\alpha = 50$ and 100 for MF_1 and MF_2, respectively. To estimate the transformation between $w^{(s2)}(N+1)$ and $w^{(s2)}(N)$, we use

$$r_i(N) = \frac{w_{2i-1}^{(s2)}(N+1)}{w_i^{(s2)}(N)} \qquad i = 1, ..., 2^N$$

where $w_i^{(s2)}(N)$ and $w_{2i-1}^{(s2)}(N+1)$, $i = 1, ..., 2^N$, are the weights of $MF^{(s2)}$ at stages N and $N+1$, respectively. We compute the distribution $P_N(r)$ from its histogram based on $\{r_i(N), i = 1, ..., 2^N\}$. We then plot $P_N(r)$ against r for different stages N. Fig. 1 shows $P_N(r)$ against r curves for $N = 13, ..., 17$. We observe that those curves neatly collapse together, indicating that $P_N(r)$ is quite independent of the stage number N for reasonably large values of N.

We then model $MF^{(s2)}$ using a single ideal multiplicative multifractal, $MF^{(e)}$. For this purpose, we use the distribution $P_N(r)$ as calculated for reasonably large values of N as the multiplier distribution for $MF^{(e)}$. From Fig. 1 we find that $P_N(r)$ can be well fitted by a truncated Gaussian distribution with $\alpha = 67$. We then generate $MF^{(e)}$ until stage $N = 18$. We drive a single server queueing system, on the one hand by $MF^{(e)}$, and on the other hand by $MF^{(s2)}$, and compare the system size tail distributions. Fig. 2 shows this comparison. We observe that the system-size tail distributions are almost identical for the two queueing systems for all four utilisation levels, $\rho = 0.3, 0.5, 0.7$ and 0.9. Hence, $MF^{(s2)}$ is indeed equivalent to $MF^{(e)}$ in terms of queueing performance.

Summary: We have proven that the superposition of a finite number of multiplicative multifractal traffic streams results asymptotically in another multifractal process. The Hurst parameter for the superimposed process is the same as the corresponding one for the source traffic stream that has the largest second moment of the multiplier distribution. Furthermore, we find in numerical simulations that the superimposed process is typically asymptotically a multiplicative process, and can be modelled using a single ideal multiplicative multifractal. These results also shed light on why aggregated LAN and WAN traffic streams can be effectively represented as an ideal multiplicative multifractal traffic stream [5, 6].

Acknowledgment: The authors thank Z.-S. She for some useful discussions. This work was supported by UC MICRO/SBC Pacific Bell research grant 98-131 and by ARO grant DAAGIJ-98-1-0338.

16 February 2000

Electronics Letters Online No: 20000486

DOI: 10.1049/el:20000486

J. Gao and I. Rubin (*Department of Electrical Engineering, University of California, Los Angeles, Los Angeles, CA 90095, USA*)

E-mail: jbgao@ee.ucla.edu

References

1 LELAND, W.E., TAQQU, M.S., WILLINGER, W., and WILSON, D.V.: 'On the self-similar nature of Ethernet traffic (extended version)', *IEEE/ACM Trans. Netw.*, 1994, **2**, pp. 1–15

2 PAXSON, V., and FLOYD, S.: 'Wide area traffic – the failure of Poisson modeling', *IEEE/ACM Trans. Netw.*, 1995, **3**, pp. 226–244

3 BERAN, J., SHERMAN, R., TAQQU, M.S., and WILLINGER, W.: 'Long-range-dependence in variable-bit-rate video traffic', *IEEE Trans. Commun.*, 1995, **43**, pp. 1566–1579

4 CROVELLA, M.E., and BESTAVROS, A.: 'Self-similarity in world wide web traffic: evidence and possible causes', *IEEE/ACM Trans. Netw.*, 1997, **5**, pp. 835–846

5 GAO, J.B., and RUBIN, I.: 'Multiplicative multifractal modeling of long-range-dependent traffic'. Proc. ICC'99, Vancouver, Canada, June 1999

6 GAO, J.B., and RUBIN, I.: 'Multifractal modeling of counting processes of long-range dependent network traffic'. Proc. SCS Advanced Simulation Technologies Conf., San Diego, CA, 1999

7 GAO, J.B., and RUBIN, I.: 'Multifractal analysis and modelling of VBR video traffic', *Electron. Lett.*, 2000, **36**, (3), pp. 278–279

8 GAO, J.B., and RUBIN, I.: 'Statistical properties of multiplicative multifractal processes in modelling telecommunications traffic streams', *Electron. Lett.*, 2000, **36**, (1), pp. 101–102

Time-varying filtering of speech signals using linear prediction

S. Stanković and J. Tilp

Time-varying filtering of noisy speech signals based on linear prediction (LP) is presented. The approach is tested on noisy speech signals apparent in hands-free telephone systems.

Introduction: Basic difficulties in the filtering of noisy speech signals arise from their highly nonstationary and multicomponent nature [1]. When dealing with short-time speech spectra, the dynamic range of those components is usually as high as 40dB. This means that the energies of speech and noise vary significantly at different frequencies. By using linear prediction (LP) [2], we can simultaneously consider the excitation part of speech signals (with approximately constant spectral amplitudes) and a remaining part obtained by subtraction of the excitation signal from the speech signal itself. This remaining part of the speech signal will be referred to as the pseudo-envelope. The aim of this Letter is to separately apply time-varying filtering [3, 4] to the excitation signal and to the pseudo-envelope of speech signals. An efficient method for time-varying filtering of speech signals is thus obtained. The basic concept is illustrated by an experimental example of filtering a noisy speech signal in a hands-free telephone system.

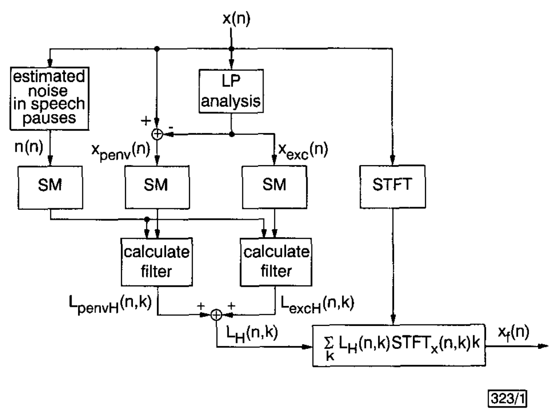

Fig. 1 *Block diagram of filtering approach*

Linear predictive analysis: A linear predictive analysis of a speech signal is based on an all-pole model [2], where the speech signal is defined by

$$s(n) = \sum_{i=1}^{p} a_i(n)s(n-i) + e(n) \qquad (1)$$

The error or LP residual (i.e. the excitation) is denoted by $e(n)$, and $a_i(n)$ are the weights applied to the previous speech samples, for any time n. The weights correspond to the direct-form coefficients of a non-recursive filter, the transfer function of which is given as

$$A(n,z) = 1 - \sum_{i=1}^{p} a_i(n)z^{-i} \qquad (2)$$

Passing the speech signal through the filter $A(n, z)$ results in the removal of the near-sample correlations and produces the excitation $e(n)$. The LP analysis has both time- and frequency-domain interpretations. The key aspect for applying LP in our approach is that it is possible to extract the excitation signal and, on the basis of the excitation signal, the pseudo-envelope as well. To keep the signal delay short, it is possible to determine the coefficients $a_i(n)$ sequentially using an adaptive LMS-type algorithm.

Time-varying filtering: Time-varying filtering of the noisy signal $x(t) = s(t) + n(t)$ can be performed as follows [3, 4]:

$$(Hx)(t) = \int_{-\infty}^{\infty} h\Big(t + \frac{\tau}{2}, t - \frac{\tau}{2}\Big)w(\tau)x(t+\tau)d\tau \qquad (3)$$