TABLE II
OUR ATTACK COMPARED TO A GENERIC TIME/MEMORY/DATA TRADEOFF ATTACK

| $T$ | $D$ | $M$ | $P$ |
|---|---|---|---|
| $2^{0.50L}$ | $2^{0.50L}$ | — | — |
| $2^{0.52L}$ | $2^{0.32L}$ | — | — |
| $2^{0.54L}$ | $2^{0.21L}$ | $2^{0.52L}$ | $2^{0.79L}$ |
| $2^{0.56L}$ | $2^{0.14L}$ | $2^{0.58L}$ | $2^{0.86L}$ |
| $2^{0.58L}$ | $2^{0.09L}$ | $2^{0.62L}$ | $2^{0.91L}$ |
| $2^{0.60L}$ | $2^{0.05L}$ | $2^{0.65L}$ | $2^{0.95L}$ |

From the table it is clear that using the same time complexity and amount of keystream the generic time/memory/data tradeoff attack requires an infeasible amount of memory and precomputation. A typical point on the curve, mentioned in [9], is $P = T = 2^{0.66L}$ and $M = D = 2^{0.33L}$. This point will give more realistic values, and comparing it to our attack we see that it uses both more data and more computation than a typical point on our curve.

## VII. CONCLUSION

Since the introduction of the self-shrinking generator in 1994 several attacks have been proposed, some requiring only a small known keystream while others need longer sequence to succeed. In this correspondence, we presented two new attacks on the self-shrinking generator, one using a short keystream and one requiring a longer keystream. In the first attack, operating on a very short known keystream, we showed that the complexity is approximately the same as the best previously known attack (the BDD-based attack). However, our attack needs almost no memory whereas the BDD-based attack is unpractical due to the large memory required. In the second attack we assumed a longer known keystream. It was shown that the asymptotic computational complexity for this attack is significantly lower than in the previously best attack, for any amount of known keystream of length $2^{\alpha L}$ when $0 < \alpha < 0.5$.

## ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for useful suggestions and comments.

## REFERENCES

[1] M. Krause, , L. Knudsen, Ed., "Advances in Cryptology—EUROCRYPT 2002, ser. Lecture Notes in Computer Science," in *BDD-Based Cryptanalysis of Keystream Generators*. New York: Springer-Verlag, 2002, vol. 2332, pp. 222–237.

[2] D. Coppersmith, H. Krawczyk, and Y. Mansour, , D. Stinson, Ed., "Advances in Cryptology—CRYPTO'93, ser. Lecture Notes in Computer Science," in *The Shrinking Generator*. New York: Springer-Verlag, 1993, vol. 773, pp. 22–39.

[3] W. Meier and O. Staffelbach, , A. D. Santis, Ed., "Advances in Cryptology—EUROCRYPT'94 ser. Lecture Notes in Computer Science," in *The Self-Shrinking Generator*. New York: Springer-Verlag, 1994, vol. 905, pp. 205–214.

[4] E. Zenner, M. Krause, and S. Lucks, , V. Varadharajan and Y. Mu, Eds., "Australasian Conference on Information Security and Privacy ACISP'01, ser. Lecture Notes in Computer Science," in *Improved Cryptanalysis of the Self-Shrinking Generator*. New York: Springer-Verlag, 2001, vol. 2119, pp. 21–35.

[5] M. Mihaljevic, , J. Pieprzyk and J. Seberry, Eds., "First Australasian Conference on Information Security and Privacy ACISP'96, ser. Lecture Notes in Computer Science," in *A Faster Cryptanalysis of the Self-Shrinking Generator*. New York: Springer-Verlag, 1996, vol. 1172, pp. 182–189.

[6] S. Roman, *Coding and Information Theory, ser. Graduate Texts in Mathematics*. New York: Springer-Verlag, 1992.

[7] J. Golić, , W. Fumy, Ed., "Advances in Cryptology—EUROCRYPT'97, ser. Lecture Notes in Computer Science," in *Cryptanalysis of Alleged A5 Stream Cipher*. New York: Springer-Verlag, 1997, vol. 1233, pp. 239–255.

[8] S. Babbage, "A space/time tradeoff in exhaustive search attacks on stream ciphers," in *Proc. European Conv. Security Detection, ser. IEE Conf.*, 1995, no. 408.

[9] A. Biryukov and A. Shamir, , T. Okamoto, Ed., "Advances in Cryptology—ASIACRYPT 2000, ser. Lecture Notes in Computer Science," in *Cryptanalytic Time/Memory/Data Tradeoffs for Stream Ciphers*. New York: Springer-Verlag, 2000, vol. 1976, pp. 1–13.

[10] M. Hellman, "A cryptanalytic time-memory trade-off," *IEEE Trans. Inf. Theory*, vol. IT-26, no. 4, pp. 401–406, Jul. 1980.

# Synchronization of Pseudorandom Signals by Forward-Only Message Passing With Application to Electronic Circuits

Benjamin Vigoda, *Member, IEEE*, Justin Dauwels, *Member, IEEE*, Matthias Frey, *Student Member, IEEE*, Neil Gershenfeld, Tobias Koch, *Student Member, IEEE*, Hans-Andrea Loeliger, *Fellow, IEEE*, and Patrick Merkli, *Member, IEEE*

***Abstract*—It has been observed that a linear-feedback shift-register (LFSR) sequence can be synchronized by feeding the modulated sequence into a "soft" (or "analog") version of the LFSR. In this correspondence, the "soft LFSR" is derived as forward-only message passing in the corresponding factor graph. A continous-time analog (suitable for realization as a clockless electronic circuit) is then given of both the LFSR and the soft LFSR. A connection is thus established between statistical state estimation and the phenomenon of entrainment of dynamical systems, which opens the prospect of deriving dynamical systems (such as electronic circuits) with strong entrainment capabilities from more powerful message passing algorithms.**

***Index Terms*—Circuits, dynamical systems, entrainment, factor graphs, linear-feedback shift registers, message passing, nonlinear filtering, synchronization.**

## I. INTRODUCTION

Pseudorandom signals play an important role in spread-spectrum communications [1], [2] and in various measurement systems. In such systems, the synchronization of pseudorandom signals is a problem of significant interest. The standard solution to this problem is based on correlating the incoming signal with (a segment of) the pseudorandom

Manuscript received June 13, 2003; revised March 22, 2006. The work of B. Vigoda and N. Gershenfeld was supported by the NSF under Grant CCR-0122419. The work of J. Dauwels was supported by the Swiss NF under Grant 200021–101955. The material in this correspondence was presented in part at the 41st Annual Allerton Conference on Communication, Control, and Computing, Monticello, IL, October 2003.

B. Vigoda was with the MIT Media Lab, Massachusetts Institute of Technology, Cambridge, MA 02139 USA. He is now with the Mitsubishi Electric Research Lab, Cambridge, MA 02139 USA (e-mail: vigoda@merl.com).

J. Dauwels was with the Department of Information Technology and Electrical Engineering, ETH Zurich, CH-8092 Zurich, Switzerland. He is now with RIKEN Brain Science Institute, Saitama, Japan (e-mail: justin@dauwels.com).

M. Frey, T. Koch, and H.-A. Loeliger are with the Department of Information Technology and Electrical Engineering, ETH Zurich, CH-8092 Zurich, Switzerland (e-mail: frey@isi.ee.ethz.ch; tkoch@isi.ee.ethz.ch; loeliger@isi.ee.ethz.ch).

N. Gershenfeld is with the MIT Center for Bits and Atoms, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: neilg@cba.mit.edu).

P. Merkli was with the Department of Information Technology and Electrical Engineering, ETH Zurich, CH-8092 Zurich, Switzerland. He is now with Sensirion AG, Staefa, Switzerland (e-mail: patrick.merkli@sensirion.com).

Communicated by A. Kavčić, Associate Editor for Detection and Estimation.
Digital Object Identifier 10.1109/TIT.2006.878165

signal, which leads to a long acquisition time if the period of the signal is large.

Perhaps the most popular class of pseudorandom signals are generated by linear-feedback shift registers (LFSRs). Both Gershenfeld and Grinstein [3] and Yang and Hanzo [4], [5] observed that LFSR sequences can be synchronized by means of a "soft" or "analog" LFSR. The approach of [3] is system theoretic: the soft LFSR is a dynamical system with entrainment capabilities (cf. [6], [7], [8]) obtained by embedding the discrete state space of the LFSR into a continuous state space. By constrast, the (better) soft LFSR of [4], [5], which was independently obtained also in [9], is derived from statistical estimation; it achieves quick synchronization—e.g., after 150 samples at 0 dB for an LFSR with a period of $2^{15}-1$ samples—at very low computational cost. Related algorithms, some of them more complex and more powerful, were presented in [9]–[13].

In this correspondence, we connect the dynamical systems view of [3] with the statistical view of [4], [5], both in discrete time and in continuous time. First, we derive the soft LFSR of [4], [5] as forward-only message passing in the corresponding factor graph. We then propose a new continuous-time analog of both the LFSR and the soft LFSR, both suitable for realization as electronic circuits. We actually implemented one such circuit, and we report some measurements. It is thus demonstrated that continuous-time dynamical systems (such as clockless electronic circuits) with good entrainment properties can be derived from message passing algorithms for statistical state estimation. Such systems/circuits may have substantial advantages in terms of speed and/or power consumption over digital implementations in some applications, and they may enable entirely new applications. However, such applications are outside the scope of this correspondence.

This correspondence is structured as follows. We begin by stating the discrete-time problem in Section II. In Section III, we review maximum-likelihood estimation and its interpretation as forward-only message passing in a cycle-free factor graph. In Section IV, we obtain the soft LFSR as forward-only message passing through another factor graph, and we present some simulation results. A continuous-time analog of the (discrete-time) LFSR is proposed in Section V. The corresponding continuous-time analog of the soft LFSR and its realization as an electronic circuit are described in Section VI. Some measurements of this circuit are reported in Section VII, and some conclusions are offered in Section VIII. Some details of alternative versions of the soft LFSR (sum-product, max-product, and Gershenfeld-Grinstein) are given in the Appendix.

## II. Noisy LFSR Sequences

For fixed integers $\ell$ and $m$ satisfying $1 \le \ell < m$, let

$$X \triangleq X_{-m+1}, \ldots, X_{-1}, X_0, X_1, X_2, \ldots \tag{1}$$

be a binary sequence satisfying the recursion

$$X_k = X_{k-\ell} \oplus X_{k-m} \tag{2}$$

for $k = 1, 2, 3, \ldots$, where "$\oplus$" denotes addition modulo 2. Any such sequence will be called a LFSR (linear-feedback shift register) sequence. For $k \ge 0$, the $m$-tuple $[X]_k \triangleq (X_{k-m+1}, \ldots, X_{k-1}, X_k)$ will be called the *state* of $X$ at time $k$. The sequence $X_1, X_2, \ldots$ is observed via a memoryless channel with transition probabilities $p(y_k | x_k)$. The situation is illustrated in Fig. 1 for $\ell = 1$ and $m = 3$; the boxes labeled "$D$" are unit-delay cells.

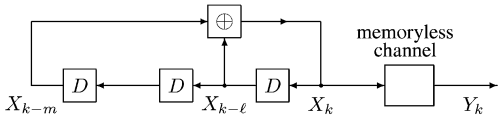


Fig. 1. LFSR sequence observed via a noisy channel.

Note that the restriction to two right-hand terms ("taps") in (2) is made only to keep the notation as simple as possible; all results of this correspondence are easily generalized to more taps. We also remark that, in most applications (and in our examples), LFSR sequences with the maximal period of $2^m - 1$ are preferred, but this condition plays no essential role in this correspondence.

From the received sequence $Y_1, Y_2, \ldots, Y_n$, we wish to estimate the state $[X]_n$ of the transmitted sequence. The computation of the maximum-likelihood (ML) estimate is straightforward and well known [1]; however, the complexity of this computation is proportional to $n2^m$, which makes it impractical unless $m$ is small.

In the examples, we will assume that the channel is defined by

$$Y_k = \tilde{X}_k + Z_k \tag{3}$$

with

$$\tilde{X}_k \triangleq \begin{cases} 1, & \text{if } X_k = 0 \\ -1, & \text{if } X_k = 1 \end{cases} \tag{4}$$

(i.e., binary antipodal signaling) and where $Z = Z_1, Z_2, \ldots$ is white Gaussian noise (i.e., independent zero-mean Gaussian random variables) with variance $\sigma^2$.

## III. ML Estimation, Trellis, and Factor Graphs

Let us recall some basic facts. First, we note that the mapping $x \mapsto [x]_k$ (from sequences to states) is invertible for any $k \ge 0$: from the forward recursion (2) and the backward recursion $X_{k-m} = X_k \oplus X_{k-\ell}$, the complete sequence $x$ is determined by its state at any time $k$.

Second, we consider the maximum-likelihood (ML) estimate of $[X]_n$. Using the notation $y^n \triangleq (y_1, \ldots, y_n)$ and $x^n \triangleq (x_{-m+1}, \ldots, x_n)$, the ML estimate of $[X]_n$ is the maximum (over all possible states $[x]_n$) of the likelihood function

$$p(y^n \mid [x]_n) = p(y^n | x^n) \tag{5}$$

$$= \prod_{k=1}^{n} p(y_k | x_k). \tag{6}$$

For the channel (3), maximizing (6) amounts to maximizing the correlation between $\tilde{x}^n$ and $y^n$.

Third, we note that the computation of (6) may be viewed as the forward recursion of the BCJR algorithm [14] through the trellis of the system or—equivalently—as forward-only message passing through the corresponding factor graph. Let us consider this more closely. Instead of factor graphs as in [15], we will use Forney-style factor graphs as in [16], where edges (or half-edges) represent variables and nodes (boxes) represent factors. A (Forney-style) factor graph of our system is shown in Fig. 2. (Add a circle on each edge to obtain a factor graph as in [15]). As in [16], we use capital letters for unknown variables and small letters for known (observed) variables. The nodes in the top row of Fig. 2. represent $\{0, 1\}$-valued functions $J(s_{k-1}, x_k, s_k)$ that indicate the allowed combinations of old state $s_{k-1} = [x]_{k-1}$, output symbol $x_k$, and new state $s_k = [x]_k$. The nodes in the bottom row

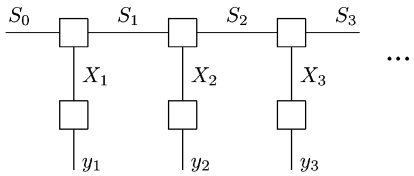


Fig. 2. Factor graph (Forney-style) corresponding to the trellis of the system in Fig. 1.

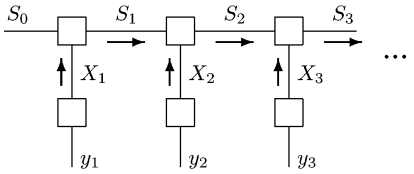


Fig. 3. Forward-only message passing through the factor graph of Fig. 2.

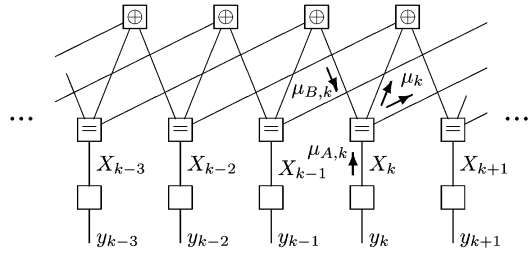


Fig. 4. Factor graph corresponding directly to Fig. 1.

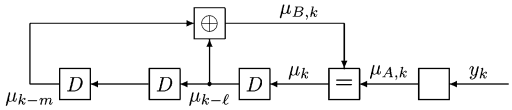


Fig. 5. Computation of messages in Fig. 4 by a "soft LFSR."

of Fig. 2 represent the channel transition probabilities $p(y_k|x_k)$. As a whole, the factor graph of Fig. 2 represents the function

$$p(y^n|x^n)J(x^n,s^n) = \prod_{k=1}^{n} J(s_{k-1},x_k,s_k)\,p(y_k|x_k) \tag{7}$$

(defined for arbitrary binary sequences $x^n$), where $J(x^n,s^n) \triangleq \prod_{k=1}^{n} J(s_{k-1},x_k,s_k)$ is the indicator function of valid LFSR sequences, which may also be viewed as a uniform prior over all valid $x^n$.

It then follows from basic factor graph theory [15], [16] that the a posteriori probability distribution over $S_n = [X]_n$ (and thus the MAP/ML estimate of $S_n$) is obtained from forward-only sum-product message passing as illustrated in Fig. 3. Since the trellis has no merging paths, the sum-product rule for the computation of messages reduces to a product-only rule and coincides with the max-product rule. By taking logarithms, the product-only rule becomes a sum-only rule; for the channel (3), this amounts to a recursive computation of the correlation between $\tilde{x}^n$ and $y^n$.

## IV. Soft LFSR

Another factor graph for our system is shown (for $\ell = 1$ and $m = 3$) in Fig. 4. This factor graph represents the function

$$p(y^n|x^n)J(x^n) = \prod_{k=1}^{n} \delta[x_k \oplus x_{k-\ell} \oplus x_{k-m}]\,p(y_k|x_k) \tag{8}$$

where $\delta[\cdot]$ is the Kronecker delta and where $J(x^n) = \prod_{k=1}^{n} \delta[x_k \oplus x_{k-\ell} \oplus x_{k-m}]$ is the indicator function for valid LFSR sequences according to (2).

As this factor graph has cycles, the standard sum-product and max-product algorithms become iterative algorithms. Such algorithms were investigated in [12] and [13]. In this correspondence, however, we stick to (noniterative) forward-only message passing. Since (full-state) forward-only message passing is optimal in Fig. 3, there is hope that (scalar) forward-only message passing in Fig. 4 might do well also. In any case, forward-only message passing in Fig. 4 amounts to a simple recursion, which may be interpreted as running the received sequence $Y$ through the "soft LFSR" circuit of Fig. 5. The quantities $\mu_{A,k}$, $\mu_{B,k}$, and $\mu_k$ in Fig. 5 are the messages indicated in Fig. 4. Note that the same message $\mu_k$ is sent along two edges out of the equality check node corresponding to $X_k$.

The computation of these messages (as indicated in Fig. 5) is a standard application of the sum-product or max-product rules [16]. Each message represents "pseudoprobabilities" $\tilde{p}(0)$ and $\tilde{p}(1)$, e.g., in the form $\tilde{p}(0)/\tilde{p}(1)$ or $\tilde{p}(0) - \tilde{p}(1)$. For the latter representation, the explicit sum-product update rules are as follows:

Initialization: $\mu_k = 0$ for $k = -m+1, -m+2, \ldots, 0$.

Recursion (for $k = 1, 2, 3, \ldots$):

$$\mu_{A,k} = \frac{p(y_k|x_k=0) - p(y_k|x_k=1)}{p(y_k|x_k=0) + p(y_k|x_k=1)} \tag{9}$$

$$\overset{\text{for AWGN}}{=} \frac{\exp(2y_k/\sigma^2) - 1}{\exp(2y_k/\sigma^2) + 1} \tag{10}$$

$$\mu_{B,k} = \mu_{k-\ell} \cdot \mu_{k-m} \tag{11}$$

$$\mu_k = \frac{\mu_{A,k} + \mu_{B,k}}{1 + \mu_{A,k} \cdot \mu_{B,k}} \tag{12}$$

Equation (9) holds for a general memoryless channel while (10) is the specialization to the channel specified at the end of Section I. At any given time $k$, an estimate of $X_k$ is obtained as

$$\hat{X}_k \triangleq \begin{cases} 0, & \text{if } \mu_k \geq 0 \\ 1, & \text{if } \mu_k < 0 \end{cases} \tag{13}$$

and $[\hat{X}]_k = (\hat{X}_{k-m+1}, \ldots, \hat{X}_{k-1}, \hat{X}_k)$ is an estimate of the state $[X]_k$.

The sum-product update rules for the case where the messages represent the ratio $\tilde{p}(0)/\tilde{p}(1)$ are given in the Appendix together with the max-product rules and the analog LFSR of [3].

Simulation results for maximum-length LFSR sequences with memory $m = 15$ and $m = 31$ are given in Figs. 6–8. All these figures show plots of the probability of synchronization

$$P_{\text{synch}}(k) \triangleq P([\hat{X}]_k = [X]_k) \tag{14}$$

either versus the time index $k$ or versus the signal-to-noise ratio $1/\sigma^2$ where $\sigma^2$ is the noise variance.

As is obvious from these plots (and from similar plots in [4], [5], [9]) the soft LFSR quickly achieves synchronization for sufficiently low noise power (up to about 0 dB) but fails for high noise power. It

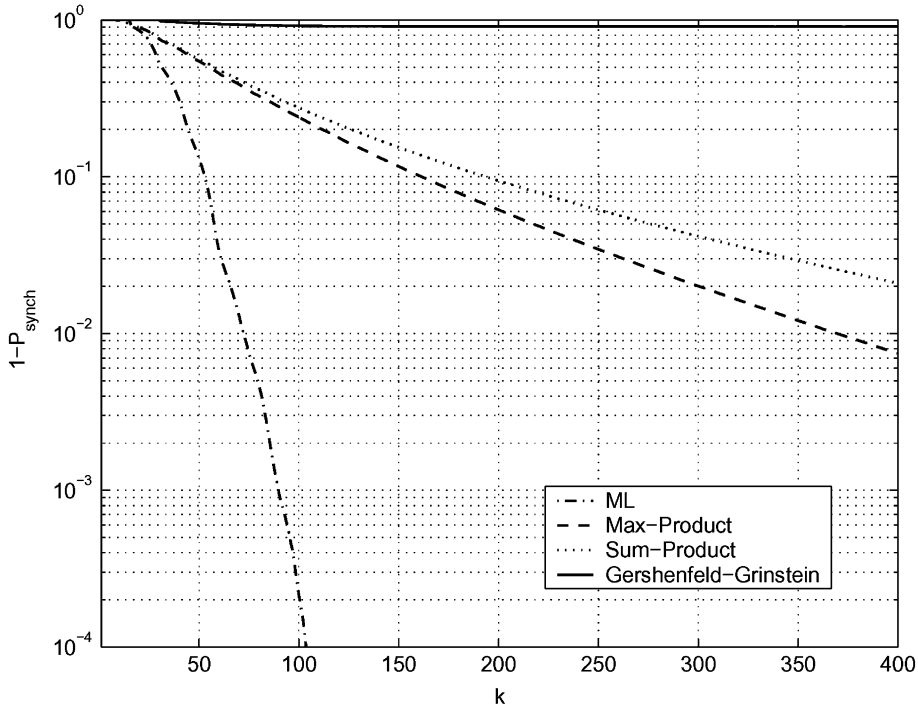


Fig. 6. $1-P_{\mathrm{sync}}(k)$ for the LFSR with $m=15$ ($\ell=1$) at SNR $=0$ dB. Algorithms (in the order of increasing performance): G.-G. soft LFSR [3]; sum-product soft LFSR; max-product soft LFSR; maximum likelihood (ML).

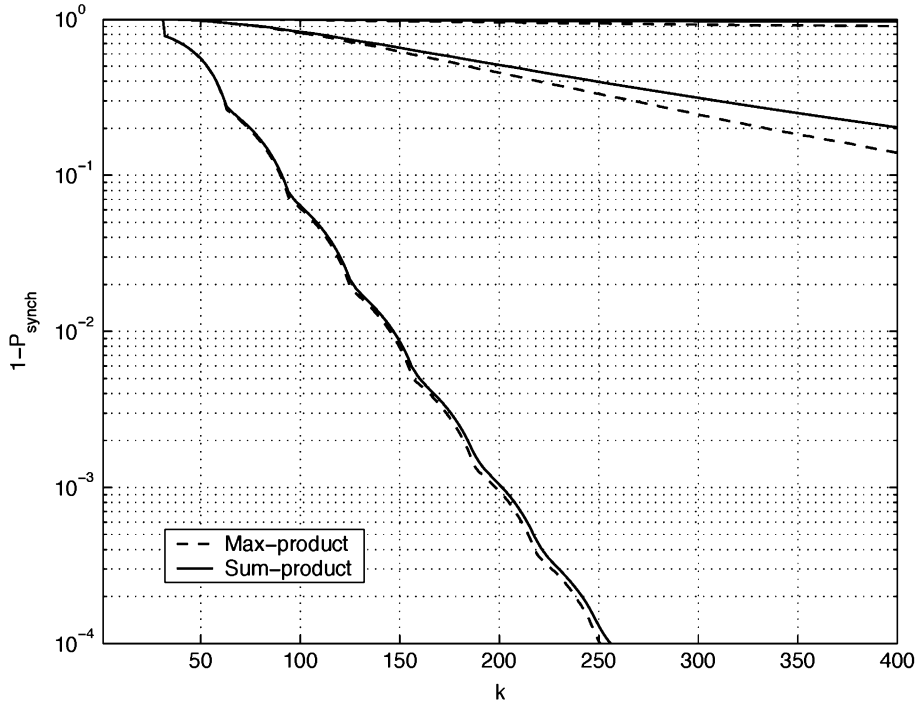


Fig. 7. $1-P_{\mathrm{sync}}(k)$ for the LFSR with $m=31$ ($\ell=3$) for three different signal-to-noise ratios: SNRs $=-2.92$ dB ($\sigma=1.4$), SNR $=0$ dB ($\sigma=0$), and SNR $=4.44$ dB ($\sigma=0.6$). Algorithms: max-product soft LFSR and sum-product soft LFSR.

is remarkable that the max-product algorithm gives better performance than the sum-product algorithm, but the difference is small.

We also note that better performance can be achieved both with more complex forward-only message passing [9], [10] and with iterative message passing, cf. [12], [13].

## V. A Continuous-Time Pseudorandom Generator

We now proceed to an analog of Figs. 1 and 5 in continuous time. Our proposal for a continuous-time analog of Fig. 1 is shown in Fig. 9. The signal $X(t)$ in Fig. 9 takes values in the set $\{+1,-1\}$. The multiplier in Fig. 9 corresponds to the $\mathrm{mod}$-2 addition in Fig. 1.

How should we translate the delay cells in Fig. 1 to continuous time? An obvious approach would be to simply translate them into continuous-time delay cells. However, ideal continuous-time delay cells cannot be realized by real circuits (except perhaps in optics); even a delay line (e.g., a piece of wire) has a low-pass characteristic.

We therefore choose to replace the discrete-time delay cells of Fig. 1 by low-pass filters with transfer functions $H_1(s)$ and $H_2(s)$ as shown

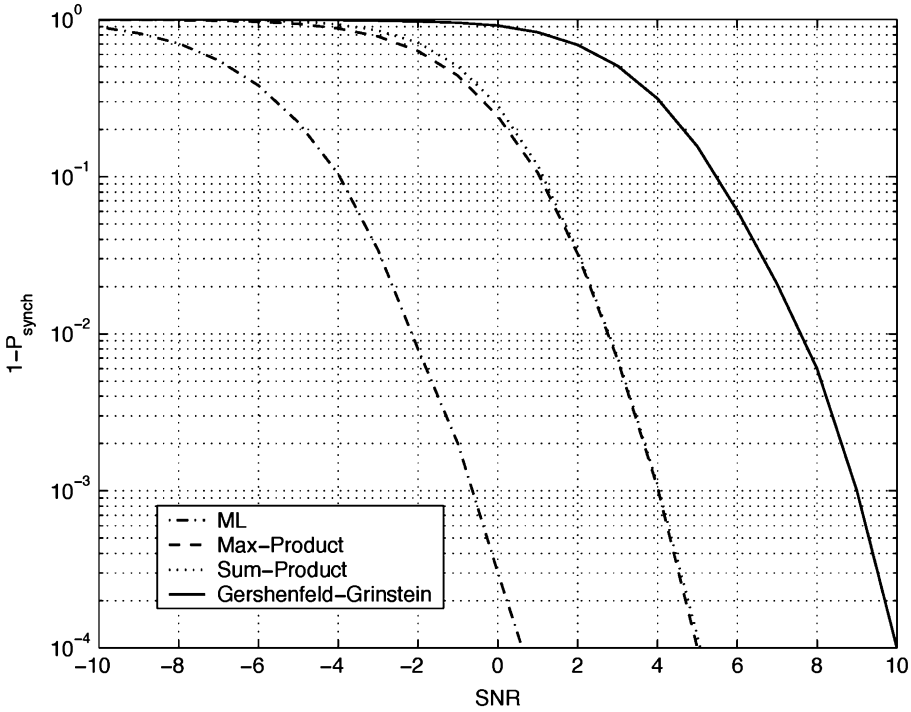


Fig. 8. $1 - P_{\text{sync}}(k = 100)$ versus SNR for the LFSR with $m = 15$ ($\ell = 1$). Algorithms (in the order of increasing performance): G.-G. soft LFSR [3]; sum-product soft LFSR; max-product soft LFSR; maximum likelihood (ML).

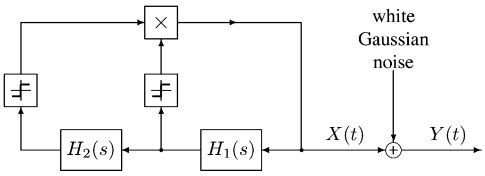


Fig. 9. Continuous-time analog to Fig. 1 with low-pass filters instead of delay cells.

in Fig. 9. Since the output signal of such filters is not restricted to $\{+1, -1\}$, we introduce threshold elements between the filter outputs and the multiplier, which reduce the filtered signals to their sign ($+1$ or $-1$). These threshold elements have no counterpart in Fig. 1 (and will create a small problem in the receiver).

The memoryless channel in Fig. 1 is translated into the additive white Gaussian channel shown in Fig. 9.

The type of signal $X(t)$ generated by the circuit of Fig. 9 is illustrated in Fig. 10 (top). From our simulations, it appears that the signal $X(t)$ is generically periodic. The actual signal depends, of course, on the two filters. In our examples, the first filter (with transfer function $H_1(s)$) is a fifth-order Butterworth filter with $-3$ dB frequency 1.6 kHz, and the second filter (with transfer function $H_2(s)$) is a cascade of six such filters. With these filters, the circuit of Fig. 9 is a dynamical system with a 35-dimensional state space. The resulting signal $X(t)$ is periodic with a period of 34 ms, 10 ms of which are shown in Fig. 10 (top).

It should be emphasized that, at present, we do not have a theory of such circuits and we cannot predict the period of the generated sequence $X(t)$. However, our simulation experiments (e.g., in [17]) suggest that a long period—"long" meaning many zero-crossings—requires a high-dimensional state space.

## VI. Circuit That Locks Onto the Pseudorandom Signal

A continuous-time analog to the soft LFSR of Fig. 5 matched to the pseudorandom generator of Fig. 9 is shown in Fig. 11. The linear filters $H_1(s)$ and $H_2(s)$ in Fig. 11 are identical to those in Fig. 9. All signals in Fig. 11 should be viewed as approximations of expectations of the corresponding signals in Fig. 9 (conditioned on the previous observations). Note that, for $\{+1, -1\}$ valued signals, the mean coincides with the difference $\tilde{p}(+1) - \tilde{p}(-1)$. It follows that the multiplier ③ in Fig. 11 computes (the continuous-time analog of) the message $\mu_{A,k}(t)$ according to (11); the box ④ in Fig. 11 computes (the continuous-time analog of) the message $\mu_k$ according to (12); and the box ⑤ computes (the continuous-time analog of) the message $\mu_{A,k}$ according to (10). All these computations can be done by simple transistor circuits as described in [18]–[20] (where the pseudoprobabilities $\tilde{p}(+1)$ and $\tilde{p}(-1)$ are represented by a pair of currents).

Consider next the filtered signals. Let $S_1(t)$ denote the output signal of the filter $H_1(s)$ in Fig. 9 and let $h_1(t)$ be the impulse response of that filter (i. e., the inverse Laplace transform of $H_1(s)$). We thus have

$$S_1(t) = \int_{-\infty}^{\infty} h_1(\tau)\, X(t-\tau)\, d\tau \tag{15}$$

and

$$\mathrm{E}[S_1(t)] = \int_{-\infty}^{\infty} h_1(\tau)\, \mathrm{E}[X(t-\tau)]\, d\tau \tag{16}$$

where the expectation is a (time dependent) ensemble average based on the (time dependent) pseudoprobabilities $\tilde{p}(+1)$ and $\tilde{p}(-1)$. It follows that the output of the filter $H_1(s)$ in Fig. 11—which is given by the right-hand side of (16)—is the expected value of $S_1(t)$. In other words, *all* signals in Fig. 11 may be viewed as (approximations of) expectations of the corresponding signals in Fig. 9.

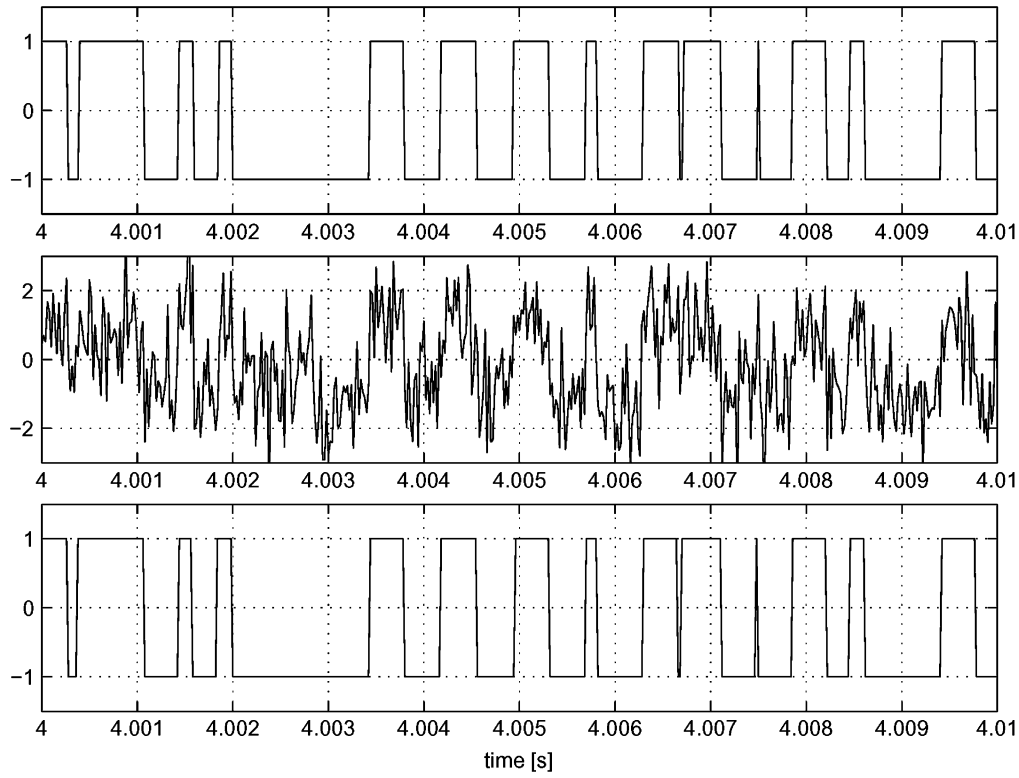


Fig. 10. Top: example of pseudorandom signal $X(t)$ generated by the circuit of Fig. 9. Middle: noisy signal $Y(t)$ as in Fig. 9 at SNR $= 0$ dB. Bottom: measured output signal $\hat{X}(t)$ of the circuit of Fig. 11 fed with $Y(t)$.

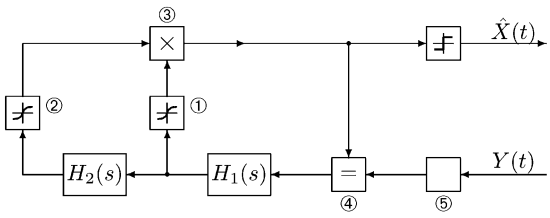


Fig. 11. Continuous-time analog of Fig. 5.

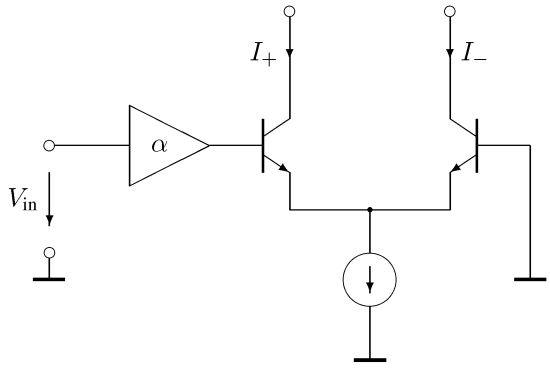


Fig. 12. Differential transistor pair used for blocks ①, ②, and ⑤.

So far, all computations have been locally exact in the same sense as in the discrete-time case (i.e., ignoring cycles in the factor graph). This fails, however, for the threshold elements in Fig. 9: the (instantaneous) expectation of the output signal of such a threshold element is not determined by the (instantaneous) expectation of its input signal. At this point, however, practical considerations strongly suggest to implement the boxes ① and ② by the circuit of Fig. 12. This circuit accepts as input a voltage and produces as output two currents $I_+$ and $I_-$ proportional to $\tilde{p}(+1)$ and $\tilde{p}(-1)$, respectively.

This same circuit is also used to implement the box ⑤ *exactly* (where the amplification $\alpha$ depends on the SNR and on the temperature). As an implementation of ① and ②, the circuit is an approximation; it would be exact (for the correct choice of $\alpha$) if the distribution of the filtered signals—more precisely, the full sum-product message at the input of the soft-threshold elements—would be the logistic distribution

$$f(x) = \frac{1}{\beta\left(e^{\frac{x-\mu}{2\beta}} + e^{-\frac{x-\mu}{2\beta}}\right)^2} \tag{17}$$

with mean $\mu$ and variance $\pi\beta/\sqrt{3}$ [21, Appendix E]. In our experiments, the amplification $\alpha$ of these circuits was manually adjusted for the best performance.

## VII. Some Measurements

Simulation results of analog circuits are often subject to doubt concerning their robustness with respect to nonidealities. We therefore built the system of Fig. 11 as an actual (clockless) electronic circuit with discrete components. The filters were realized as active RC filters with integrated operational amplifiers.

For the measurements, the clean signal $X(t)$ as well as the noisy signal $Y(t)$ were created by *simulating* the circuit of Fig. 9 on a (digital) computer; the noisy signal $Y(t)$ was then passed as input to the

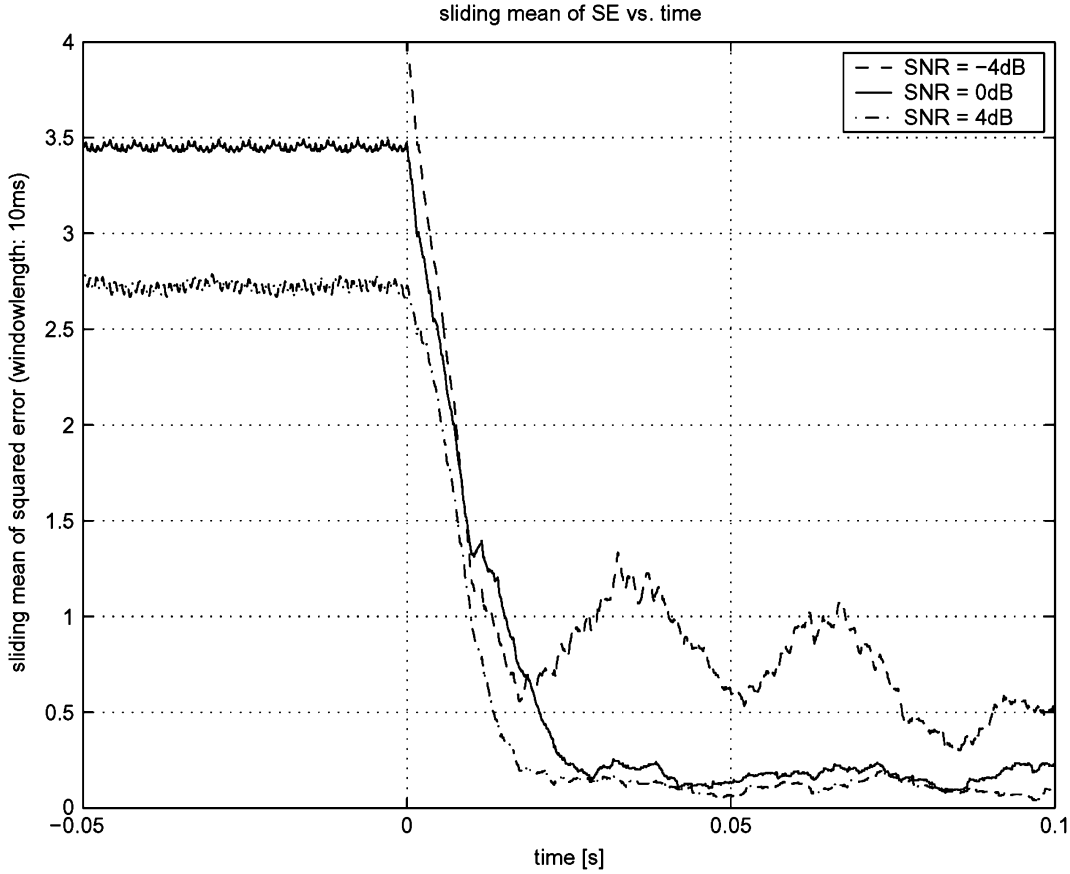


Fig. 13. Average squared error versus time after switching the transmission on.

electronic realization of Fig. 11. A typical measured output signal $\hat{X}(t)$ is shown in Fig. 10 (bottom).

Some measurements of this system are given in Figs. 13–15. For the measurements of Figs. 13 and 14, the signal $Y(t)$ is replaced by a constant signal with value $-1$ for $t < 0$. Both figures show the squared error (SE) $(\hat{X}(t) - X(t))^2$ averaged, first, over a sliding window and then, over a number of experiments. Fig. 13 shows the SE (averaged over 10 ms and over five experiments) versus the time $t$; Fig. 14 shows the SE (averaged over 1 s and over five experiments) versus the SNR at time $t = 4$ s (which is the steady state). Note that the receiver achieves good synchronization for an SNR down to about 0 dB. Not surprisingly, a signal with a longer period (top in Fig. 14) is more difficult to synchronize than a signal with a shorter period (bottom in Fig. 14).

It is instructive to observe what happens when the input to the receiving circuit is switched off for a while as illustrated in Fig. 15. Before the interruption, the receiver is synchronized. The signal $Y(t)$ is then masked (i.e., overwritten by zero) for 20 ms. During the interruption, $X(t)$ and $\hat{X}(t)$ drift apart and the averaged SE increases. The figure shows the signals $X(t)$ and $\hat{X}(t)$ around the critical moment when $Y(t)$ is switched on again.

## VIII. Concluding Remarks

Gershenfeld and Grinstein demonstrated the synchronization of LFSR sequences (both in discrete time and in continuous time) by an "analog LFSR," which was obtained by embedding the discrete state space of the LFSR into a larger continuous state space. In this correspondence, we derived such dynamical systems from message passing algorithms for statistical state estimation. First, we noted that the soft LFSR proposed by Yang and Hanzo may be obtained by forward-only message passing through a factor graph. Second, we proposed a new continuous-time analog of both the LFSR and the soft LFSR that can be realized as a practical electronic circuit. We have thus established aconnection between statistical state estimation and the phenomenon of entrainment of dynamical systems. It follows that dynamical systems (e.g., electronic circuits) with better entrainment capabilities may be obtained from more powerful (more complex) message passing algorithms.

## Appendix I
## Alternative Message Update Rules for the Soft LFSR

For the convenience of the reader, we explicitly state all computations in the soft LFSR for an alternative (more standard) version of the sum-product algorithm, for the max-product (min-sum) algorithm, as well as for the analog LFSR of Gershenfeld and Grinstein.

### A. Sum-Product LFSR for Likelihood Ratio Representation

If the messages represent the *ratio* $\tilde{p}(0)/\tilde{p}(1)$ of the pseudoprobabilities, the sum-product update rules of the soft LFSR are as follows.

Initialization: $\mu_k = 1$ for $k = -m+1, -m+2, \ldots, 0$.

Recursion (for $k = 1, 2, 3, \ldots$):

$$\mu_{A,k} = \frac{p(y_k|x_k = 0)}{p(y_k|x_k = 1)} \tag{18}$$

$$\overset{\text{for AWGN}}{=} \exp(2y_k/\sigma^2) \tag{19}$$

$$\mu_{B,k} = \frac{1 + \mu_{k-\ell} \cdot \mu_{k-m}}{\mu_{k-\ell} + \mu_{k-m}} \tag{20}$$

$$\mu_k = \mu_{A,k} \cdot \mu_{B,k} \tag{21}$$

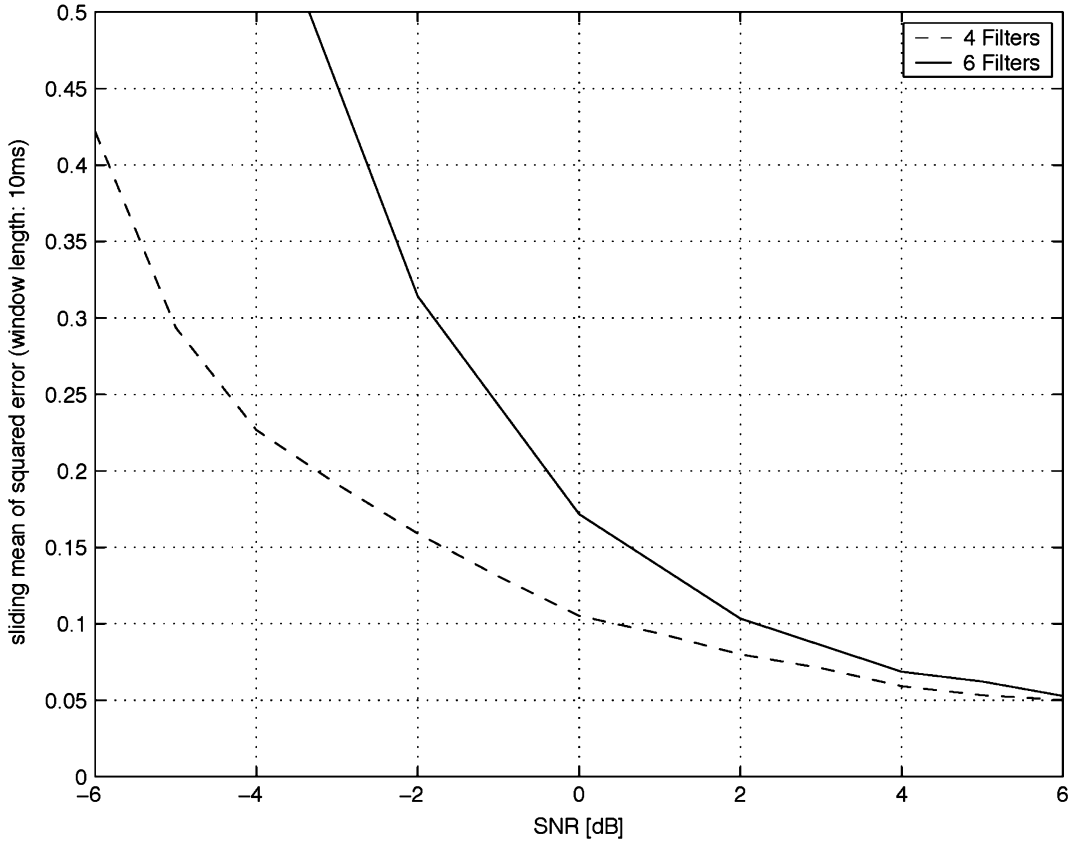


Fig. 14. Average squared error in steady state versus SNR. Dashed curve: pseudorandom signal with shorter period (7 ms instead of 34 ms, achieved with $H_2(s) = H_1(s)^4$ instead of $H_2(s) = H_1(s)^6$).

At any given time $k$, an estimate of $X_k$ is obtained as

$$\hat{X}_k \triangleq \begin{cases} 0, & \text{if } \mu_k \geq 1 \\ 1, & \text{if } \mu_k < 1 \end{cases} \tag{22}$$

and $[\hat{X}_k] = (\hat{X}_{k-m+1}, \ldots, \hat{X}_{k-1}, \hat{X}_k)$ is an estimate of the state $[X_k]$.

### B. Max-Product Soft LFSR

We state the max-product soft LFSR [15], [16] for the case where the messages represent $\ln(\tilde{p}(0)/\tilde{p}(1))$.

Initialization: $\mu_k = 0$ for $k = -m+1, -m+2, \ldots, 0$.

Recursion (for $k = 1, 2, 3, \ldots$)

$$\mu_{A,k} = \ln \frac{p(y_k|x_k = 0)}{p(y_k|x_k = 1)} \tag{23}$$

$$\stackrel{\text{for AWGN}}{=} 2y_k/\sigma^2 \tag{24}$$

$$|\mu_{B,k}| = \min\{|\mu_{k-\ell}|, |\mu_{k-m}|\} \tag{25}$$

$$\operatorname{sgn}(\mu_{B,k}) = \operatorname{sgn}(\mu_{k-\ell}) \cdot \operatorname{sgn}(\mu_{k-m}) \tag{26}$$

$$\mu_k = \mu_{A,k} + \mu_{B,k} \tag{27}$$

where $\operatorname{sgn}(x)$ denotes the sign of $x$. Finally, we have

$$\hat{X}_k \triangleq \begin{cases} 0, & \text{if } \mu_k \geq 0 \\ 1, & \text{if } \mu_k < 0 \end{cases} \tag{28}$$

In fact, (24) may be replaced by

$$\mu_{A,k} = y_k \tag{29}$$

which amounts to multiplying all messages by $\sigma^2/2$ and does not change the estimate (28).

### C. Analog LFSR by Gershenfeld and Grinstein

In [3], Gershenfeld and Grinstein obtained a discrete-time "analog" LFSR by embedding the discrete dynamics of the LFSR into a continuous state space. They showed that such an analog LFSR entrains to a LFSR sequence even if the latter is modulated by a weak data signal. An extension of this approach to continous time (using ideal continuous-time delay cells) is also given in [3]. In the setup of this correspondence, the analog LFSR of [3] can be described as follows.

Initialization: $\mu_k = 0$ for $k = -m+1, -m+2, \ldots, 0$.

Recursion (for $k = 1, 2, 3, \ldots$):

$$\mu_{A,k} = y_k \tag{30}$$

$$\mu_{B,k} = \cos\left[\pi\left(\frac{1-\mu_{k-\ell}}{2} + \frac{1-\mu_{k-m}}{2}\right)\right] \tag{31}$$

$$\mu_k = (1-\epsilon)\,\mu_{B,k} + \epsilon\,\mu_{A,k} \tag{32}$$

or, alternatively

$$\mu_k = \begin{cases} \mu_{B,k} & \text{if } \|\mu_{A,k}| - 1| > \delta \\ (1-\epsilon)\mu_{B,k} + \epsilon \operatorname{sgn}(\mu_{A,k}) & \text{otherwise} \end{cases} \tag{33}$$

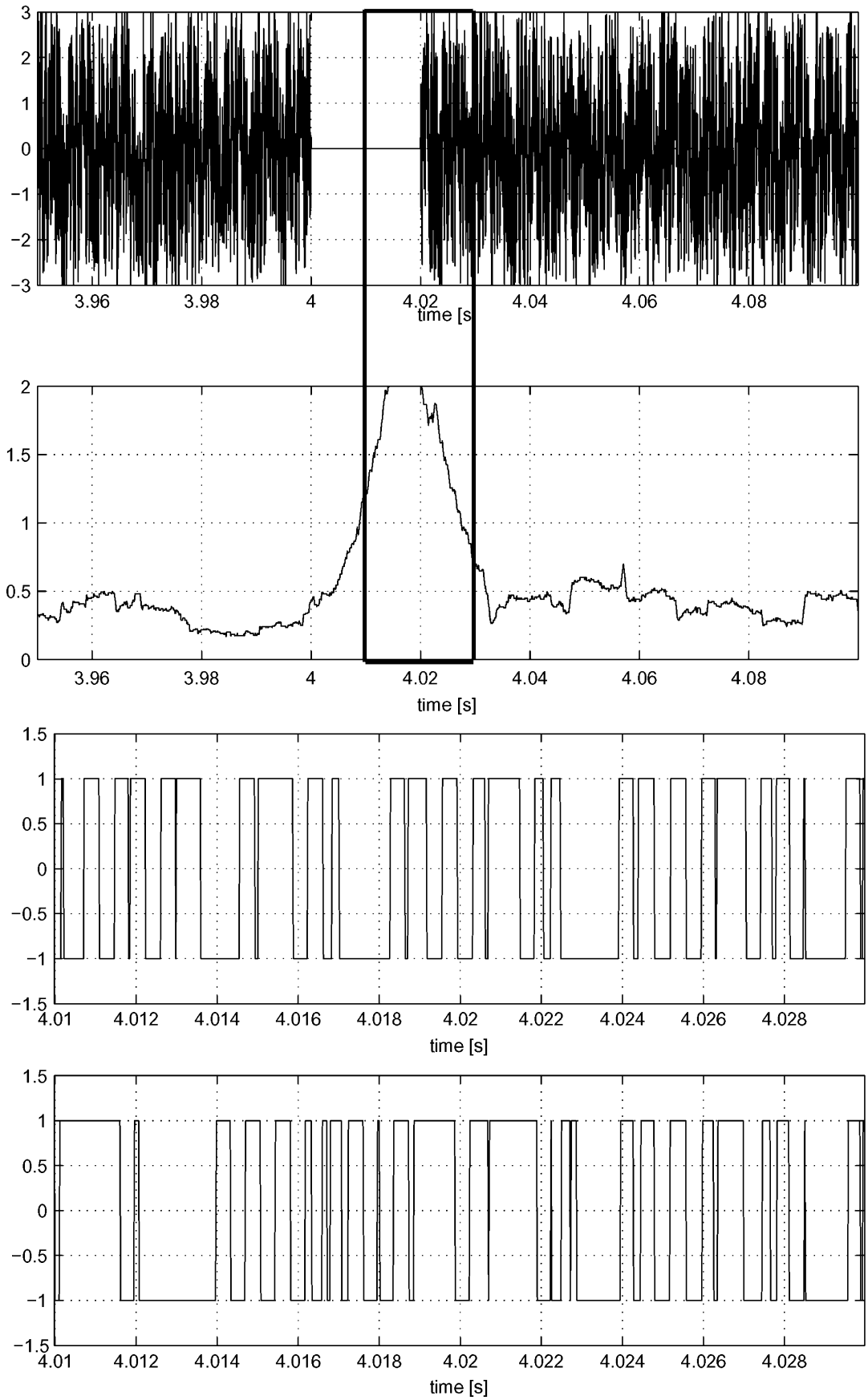


Fig. 15. Resynchronization example with modified $Y(t)$ (top), sliding-window squared error (2nd from top), $X(t)$ (2nd from bottom), and $\hat{X}(t)$ (bottom) at SNR $= 0$ dB. The plots of $X(t)$ and $\hat{X}(t)$ are zoomed to the marked interval around $t = 4.02$ s.

and

$$\hat{X}_k \triangleq \begin{cases} 0, & \text{if } \mu_k \geq 0 \\ 1, & \text{if } \mu_k < 0 \end{cases} \tag{34}$$

In this formulation (and differing from [3]), the "hard" logical values 0 and 1 are represented as $+1$ and $-1$, respectively. It should be noted that [3] does not explicitly consider noise at all.

In our simulations, we used (33) with $\delta = \infty$ and optimized $\epsilon$ ($\approx 0.4$ for large SNR).

## ACKNOWLEDGMENT

The authors are indebted to Thomas Schärer of ETH Zurich who built the hardware.

## REFERENCES

[1] M. K. Simon, J. K. Omura, R. A. Scholtz, and B. K. Levitt, *Spread Spectrum Communications*. Rockville, MD, USA: Computer Science Press, 1985, vol. 3.

[2] A. F. Molisch, *Wideband Wireless Digital Communications*. Upper Saddle River, NJ: Prentice Hall, 2001.

[3] N. Gershenfeld and G. Grinstein, "Entrainment and communication with dissipative pseudorandom dynamics," *Phys. Rev. Lett.*, vol. 74, pp. 5024–5027, June 1995.

[4] L.-L. Yang and L. Hanzo, "Iterative soft sequential estimation assisted acquisition of m-sequences," *Electron. Lett.*, vol. 38, pp. 1550–1551, Nov. 2002.

[5] L.-L. Yang and L. Hanzo, "Acquisition of m-sequences using recursive soft sequential estimation," *IEEE Trans. Commun.*, vol. 52, pp. 199–204, Feb. 2004.

[6] L. M. Pecora and T. L. Carroll, "Synchronization in chaotic systems," *Physical Review Letters*, vol. 64, pp. 821–824, Feb. 1990.

[7] K. M. Cuomo and A. V. Oppenheim, "Circuit implementation of synchronized chaos with applications to communications," *Phys. Rev. Lett.*, vol. 71, Jul. 1993.

[8] A. Abel and W. Schwarz, "Chaos communications—Principles, schemes, and system analysis," *Proc. IEEE*, vol. 90, pp. 691–710, May 2002.

[9] J. Dauwels, H.-A. Loeliger, P. Merkli, and M. Ostojic, "On structured-summary propagation, LFSR synchronization, and low-complexity trellis decoding," in *Proc. 41st Allerton Conf. Commun., Contr., Comput.*, Monticello, Illinois, Oct. 1–3, 2003, Allerton House, pp. 459–467.

[10] J. Dauwels, H.-A. Loeliger, P. Merkli, and M. Ostojic, "On Markov structured summary propagation and LFSR synchronization," in *Proc. 42nd Allerton Conf. Commun., Contr., Comput.*, Monticello, Illinois, Sep.–Oct. 29–1, 2004, Allerton House, pp. 451–460.

[11] D. Megnet, H. Mathis, P. Flammant, and A. Thiel, "C/A-code synchronization using analog feedback shift registers (AFSR)," in *Proc. ION GNSS 2004*, Long Beach, CA, Sept. 21–24, 2004, pp. 32–42.

[12] K. M. Chugg and M. Zhu, "A new approach to rapid PN code acquisition using iterative message passing techniques," *IEEE J. Sel. Areas Commun.*, vol. 23, pp. 884–897, May 2005.

[13] O. W. Yeung and K. M. Chugg, "An iterative algorithm and low complexity hardware architecture for fast acquisition of long PN codes in UWB systems," *J. VLSI Signal Process.*, vol. 43, no. 1, Apr. 2006.

[14] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inf. Theory*, vol. 20, pp. 284–287, Mar. 1974.

[15] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, pp. 498–519, Feb. 2001.

[16] H.-A. Loeliger, "An introduction to factor graphs," *IEEE Signal Proc. Mag.*, pp. 28–41, Jan. 2004.

[17] T. Koch, "Continuous-time synchronization," Tech. Rep., Signal and Information Processing Lab, ETH Zurish, Zurich, Switzerland, Jul. 2003.

[18] H.-A. Loeliger, F. Lustenberger, M. Helfenstein, and F. Tarköy, "Probability propagation and decoding in analog VLSI," *IEEE Trans. Inf. Theory*, vol. 47, pp. 837–843, Feb. 2001.

[19] F. Lustenberger, "On the Design of Analog VLSI Iterative Decoders," Ph.D. dissertation, ETH Zurich, Zurich, Switzerland, Nov. 2000.

[20] H.-A. Loeliger, "Analog decoding and beyond," in *Proc. 2001 IEEE Inf. Theory Workshop*, Cairns, Australia, Sep. 2–7, 2001, pp. 126–127.

[21] P. Merkli, "Message-Passing Algorithms and Analog Electronic Circuits," Ph.D. dissertation, ETH Zurich, Zurich, Switzerland, Apr. 2005.

# Explicit Loss Inference in Multicast Tomography

Nicholas G. Duffield, *Fellow, IEEE*, Joseph Horowitz, Francesco Lo Presti, and Don Towsley, *Fellow, IEEE*

***Abstract*—Network performance tomography involves correlating end-to-end performance measures over different network paths to infer the performance characteristics on their intersection. Multicast based inference of link-loss rates is the first paradigm for the approach. Existing algorithms generally require numerical solution of polynomial equations for a maximum-likelihood estimator (MLE), or iteration when applying the expectation maximization (EM) algorithm. The purpose of this note is to demonstrate a new estimator for link-loss rates that is computationally simple, being an explicit function of the measurements, and that has the same asymptotic variance as the MLE, to first order in the link-loss rates.**

***Index Terms*—End-to-end measurement, link-loss rates, statistical inference.**

## I. INTRODUCTION

### A. Summary

Network tomography is becoming a rapidly established discipline. One branch of this focuses on the development of statistical techniques for inferring internal network properties, such as link-loss rates [2], [4], [1], link delay statistics [7], [14] and topology [6], [9], based on end-to-end packet measurements. In this correspondence, we focus on the loss inference problem where loss observations are taken at the leaves of a tree over which packets are multicast. The current solution to this inference problem relies on obtaining maximum-likelihood estimates (MLEs) of link-level loss rates. In general, this requires finding the roots of polynomial equations associated with the internal nodes in the tree where the polynomial degree corresponds to the branching factor of the associated node. Iterative solutions via the EM algorithm can also be obtained.

In this correspondence, we derive a simple *explicit formula* for the link loss rate estimates. Although they do not correspond to the MLEs, the estimators are consistent, i.e., they converge to the true loss rates as the number of measurements grows. Furthermore, the asymptotic variance of the explicit estimator equals that of the MLE to at least first order in the loss rates.

In the remainder of this section we describe some related work. Section II defines the underlying model for multicast loss inference. The

Manuscript received April 1, 2004; revised April 6, 2006.

N. G. Duffield is with the AT&T Labs-Research, 180 Park Avenue, Florham Park, NJ 07932 USA (e-mail: duffield@research.att.com).

J. Horowitz is with the Department of Mathematics and Statistics, University of Massachusetts, Amherst, MA 01003 USA (e-mail: joeh@math.umass.edu).

F. Lo Presti is with the Computer Engineering Department, Università di Roma "Tor Vergata," 00133 Rome, Italy (e-mail: lopresti@info.uniroma2.it).

D. Towsley is with the Department of Computer Science, University of Massachusetts, Amherst, MA 01003 USA (e-mail: towsley@cs.umass.edu).

Communicated by G. Sasaki, Associate Editor for Communication Networks.

Digital Object Identifier 10.1109/TIT.2006.878228