SAMPLING JITTER CORRECTION USING FACTOR GRAPHS

Lukas Bolliger and Hans-Andrea Loeliger

ETH Zurich
Dept. of Information Technology & Electr. Eng.
8092 Zürich, Switzerland
Email: {bolliger, loeliger}@isi.ee.ethz.ch

ABSTRACT

Analog-to-digital converters are impaired by sampling clock jitter. The error induced by clock jitter depends on the slope of the analog signal at the sampling instant. In this paper, a continuous-time state space model allows to estimate the slope of the continuous-time signal, which is then used in an iterative algorithm for jitter correction.

1. INTRODUCTION

Sampling clock jitter in analog-to-digital (A/D) conversion is a well-known problem; it occurs when, due to hardware imperfections, the exact sampling instants are subject to random fluctuations. For bandlimited signals, approaches for clock jitter correction based on Linear Minimum Mean Square Error (LMMSE) estimation are proposed in [1] and [2]. In [1], a filter bank that converts a non-uniformly sampled signal to a uniformly sampled signal is extended for unknown delays in the sampling process. In [2], a Fourier series of the discrete-time signal is used to construct an LMMSE filter for clock jitter correction.

In this paper, we propose a different approach to clock jitter correction. Unlike [1] and [2], we do not assume the continuous-time analog signal to be strictly bandlimited. Instead, following [3], we assume that

- this signal is the output of a continuous-time linear system (or filter) that is driven by white noise and that
- the state space of this filter is finite dimensional.

A factor graph approach to Maximum a Posteriori (MAP) estimation (= LMMSE estimation) of such signals from noisy discrete-time observations (without clock jitter) was presented in [3]. In the present paper, we will exploit the fact that, with such a state space model, the slope of the continuous-time signal is computationally accessible.

Clearly, if the slope of the continuous-time signal is large at the sampling instant, even a small deviation of the sampling instant has a large impact on the observation. Thus, the noise variance under which the discrete-time signal is observed depends on the signal slopes at the sampling instants. We show how the algorithm proposed in [3] can be extended to estimate the slope of the continuous-time signal as well and we propose an iterative algorithm where the estimates of the slopes of the continuous-time signal are used to improve the quality of the discrete-time signal. Each iteration by itself calculates a regularized least squares estimate, where the first iteration turns out to be similar to the algorithm presented in [2]. We will use factor graphs and message passing algorithms as introduced in [4].

This paper is structured as follows: Sections 2 and 3 are about the system model and its factor graph representation, which are based on [3]. Section 4 is about some technicalities regarding the notion of signal-to-noise ratio (SNR). The proposed algorithm is presented in Section 5 and some simulation results are presented in Section 6.

2. SYSTEM MODEL

For $k = 0, \ldots, K-1$, let $\tilde{Y}_k$ be noisy samples of some continuous-time signal $Y(t)$ at the sampling instants T_k, which are random according to

$$T_k = t_k + D_k, \tag{1}$$

with known $t_0 < t_1 < t_2, \ldots$ and a random delay D_k with $\mathrm{E}[D_k] = 0$ and $\mathrm{E}\left[D_k^2\right] = \sigma_D^2$. We define the observed discrete-time (digital) signal as

$$\tilde{Y}_k \stackrel{\triangle}{=} Y(T_k) + Z_k, \tag{2}$$

where Z_k, which models quantization and general hardware imperfections, is additive white Gaussian noise with $\mathrm{E}\left[Z_k^2\right] = \sigma_Z^2$.

Some delay D_k of the sampling instant causes a larger error for $\tilde{Y}_k$ if the slope of $Y(t)$ at $t = t_k$ is large. This is expressed by a first order Taylor approximation of $Y(t)$:

$$\tilde{Y}_k = Y(t_k + D_k) + Z_k \tag{3}$$

$$\approx Y(t_k) + D_k \dot{Y}(t_k) + Z_k, \tag{4}$$

where $\dot{Y}$ denotes the derivative of Y with respect to time.

We define the approximate jitter error

$$J_k \stackrel{\triangle}{=} D_k \dot{Y}(t_k), \tag{5}$$

with

$$\mathrm{E}[J_k] = 0 \tag{6}$$

$$\mathrm{E}\left[J_k^2\right] = \sigma_D^2 \, \mathrm{E}\left[\dot{Y}(t_k)^2\right]. \tag{7}$$

Following [3] we will model $\tilde{Y}_k$ as the discrete-time observation of continuous-time filtered noise. This model is illustrated in Figure 1: A continuous-time filter is driven by continuous-time white Gaussian noise; its output is sampled and then corrupted by discrete-time additive white Gaussian noise.

The continuous-time system is known and given by its state-space representation. Hence, $X \in \mathbb{R}^n$ is the state of the system which evolves in time according to

$$\dot{X}(t) = AX(t) + bU(t), \tag{8}$$

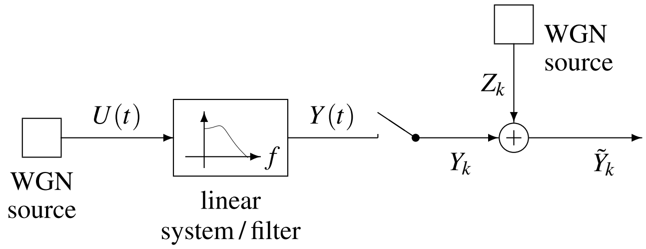


Figure 1: System model as in [3]

with $A \in \mathbb{R}^{n\times n}$ and $b \in \mathbb{R}^n$. The input $U(t) \in \mathbb{R}$ is assumed to have mean zero and an auto-correlation function

$$\mathrm{E}\left[U(t+\tau)U(t)\right] = \sigma_U^2\,\delta(\tau), \tag{9}$$

where $\delta(.)$ denotes the Dirac delta. The output $Y(t)$ of the filter is a linear combination of the state variables:

$$Y(t) = c^{\mathrm{T}}X(t) \quad \left(Y(t) \in \mathbb{R}, c \in \mathbb{R}^n\right). \tag{10}$$

Modeling $U(t)$ as continuous-time white Gaussian noise amounts to a power constraint on the input signal $U(t)$, which causes the spectrum of the estimate of $Y(t)$ to be shaped by the filter defined by A, b and c (cf. Section 3 and [3]).

$\dot{Y}(t)$ can be expressed as

$$\dot{Y}(t) = c\dot{X}(t) \tag{11}$$

$$= c^{\mathrm{T}}AX(t) + c^{\mathrm{T}}bU(t). \tag{12}$$

In the present paper, we will asssume $c^{\mathrm{T}}b = 0$, which amounts to insisting that the transfer functions from $U(t)$ to $Y(t)$ and $U(t)$ to $\dot{Y}(t)$ have low-pass characteristics. We thus have

$$\dot{Y}(t) = c^{\mathrm{T}}AX(t). \tag{13}$$

3. FACTOR GRAPH REPRESENTATION

We use Forney factor graphs (normal factor graphs) as in [4] where nodes represent factors and edges represent variables. For messages we use the same notation as in [4]: all edges are directed, the forward message along some edge X is denoted as $\overrightarrow{\mu}_X(x)$, and the backward message as $\overleftarrow{\mu}_X(x)$. The marginal density $f_X(x)$ of X equals $\overrightarrow{\mu}_X(x)\cdot\overleftarrow{\mu}_X(x)$, up to a scale factor.

In the present paper all messages are Gaussian and thus fully defined by a mean vector m and a covariance matrix V. We use $\mathcal{N}(m,V)$ to denote a Gaussian density function, thus

$$\overrightarrow{\mu}_X = \mathcal{N}(\overrightarrow{m}_X, \overrightarrow{V}_X) \tag{14}$$

$$\overleftarrow{\mu}_X = \mathcal{N}(\overleftarrow{m}_X, \overleftarrow{V}_X). \tag{15}$$

Gaussian message passing in factor graphs is discussed in detail in [4].

The factor graph described in [3] is easily extended for clock jitter correction as seen in Figure 2. To reduce calculation complexity, J_k is modeled as Gaussian noise; note that $\sigma_J^2(t_k) \triangleq \mathrm{E}\left[J_k^2\right]$ is not constant for all k.

The factor graph is cycle-free, thus all messages $\overrightarrow{\mu}_{X(t_k)}$ and $\overleftarrow{\mu}_{X(t_k)}$ can be calculated by performing one forward and one backward pass.

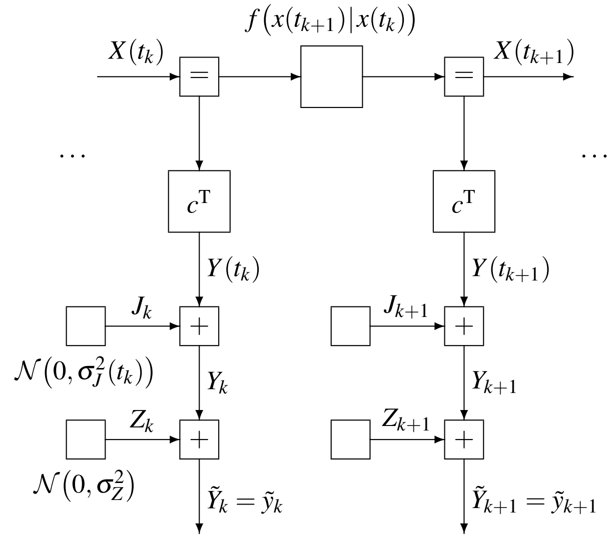
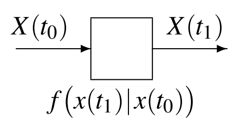


Figure 2: A Forney factor graph of the system in Figure 1.

Table 1: Computation rules as in [3] for Gaussian messages through node / factor $f\left(x(t_1)|x(t_0)\right)$ with $t_1 > t_0$.



$$\overrightarrow{m}_{X(t_1)} = e^{A(t_1-t_0)}\overrightarrow{m}_{X(t_0)} \tag{i.1}$$

$$\overrightarrow{V}_{X(t_1)} = e^{A(t_1-t_0)}\overrightarrow{V}_{X(t_0)}e^{A^{\mathrm{T}}(t_1-t_0)} + \sigma_U^2 \underbrace{\int_0^{t_1-t_0} e^{A\tau}bb^{\mathrm{T}}e^{A^{\mathrm{T}}\tau}d\tau}_{Q\overrightarrow{\Theta}(t_1-t_0)Q^{\mathrm{H}} \quad \text{see (17)}} \tag{i.2}$$

$$\overleftarrow{m}_{X(t_0)} = e^{-A(t_1-t_0)}\overleftarrow{m}_{X(t_1)} \tag{i.3}$$

$$\overleftarrow{V}_{X(t_0)} = e^{-A(t_1-t_0)}\overleftarrow{V}_{X(t_1)}e^{-A^{\mathrm{T}}(t_1-t_0)} + \sigma_U^2 \underbrace{\int_0^{t_1-t_0} e^{-A\tau}bb^{\mathrm{T}}e^{-A^{\mathrm{T}}\tau}d\tau}_{Q\overleftarrow{\Theta}(t_1-t_0)Q^{\mathrm{H}} \quad \text{see (19)}} \tag{i.4}$$

The calculation of the mean and the covariance matrix for Gaussian messages through $f\big(x(t_{k+1})|x(t_k)\big)$ are given in [3] and are repeated in Table 1. If the matrix A is diagonalizable, then the integrals in (i.2) and (i.4) can easily be expressed in closed forms. If

$$A = Q \begin{pmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{pmatrix} Q^{-1} \tag{16}$$

for some complex square matrix Q, then

$$\int_0^t e^{A\tau} b b^{\mathrm{T}} e^{A^{\mathrm{T}}\tau} d\tau = Q \overrightarrow{\Theta}(t) Q^{\mathrm{H}} \tag{17}$$

where the square matrix $\overrightarrow{\Theta}(t)$ is given by

$$\overrightarrow{\Theta}(t)_{k,\ell} \triangleq \frac{(Q^{-1}b)_k \overline{(Q^{-1}b)_\ell}}{\lambda_k + \overline{\lambda_\ell}} \left(e^{(\lambda_k + \overline{\lambda_\ell})t} - 1 \right), \tag{18}$$

and

$$\int_0^t e^{-A\tau} b b^{\mathrm{T}} e^{-A^{\mathrm{T}}\tau} d\tau = Q \overleftarrow{\Theta}(t) Q^{\mathrm{H}} \tag{19}$$

with

$$\overleftarrow{\Theta}(t)_{k,\ell} \triangleq \frac{(Q^{-1}b)_k \overline{(Q^{-1}b)_\ell}}{\lambda_k + \overline{\lambda_\ell}} \left(1 - e^{-(\lambda_k + \overline{\lambda_\ell})t} \right). \tag{20}$$

Note that, in (18) and (20), $(Q^{-1}b)_k$ denotes the k-th component of the vector $Q^{-1}b$. For the message computation rules through all other nodes/factors the reader is referred to [4].

Calculating all the messages and maximizing the marginal of $X(t_k)$ yields the Maximum a Posteriori (MAP) estimate $\hat{x}(t_k)$ of $X(t_k)$:

$$\hat{x}(t_k) \triangleq \operatorname*{argmax}_x f_{X(t_k)}(x|\tilde{y}_0, \ldots, \sigma_J^2(t_0), \ldots) \tag{21}$$

$$= \operatorname*{argmax}_x \overrightarrow{\mu}_{X(t_k)}(x) \cdot \overleftarrow{\mu}_{X(t_k)}(x). \tag{22}$$

We define $\hat{u}(t)$ as the estimate of the input signal $U(t)$ of the filter. In this paper we are not interested in $\hat{u}(t)$, cf. [3] on the calculation and the properties of $\hat{u}(t)$. Since all messages are Gaussian, the maximization in (21) is equivalent to a least squares estimation (cf. [4]). Thus, $\hat{x}(t_k)$ is a regularized least squares estimate of $X(t_k)$ for given $\sigma_J^2(t_0), \ldots, \sigma_J^2(t_{K-1})$ and $\tilde{y}_0, \ldots, \tilde{y}_{K-1}$, which minimizes the cost function

$$\frac{1}{\sigma_U^2} \int_{t_0}^{t_K} \hat{u}(t)^2 dt + \sum_{k=0}^{K-1} \frac{\big(\tilde{y}_k - c\hat{x}(t_k)\big)^2}{\sigma_Z^2 + \sigma_J^2(t_k)}. \tag{23}$$

This cost function illustrates how the parameter σ_U^2 serves as a power constraint on $\hat{u}(t)$. The ratio between the input power σ_U^2 and the average output noise power of $J_k + Z_k$ is a regularization parameter used to force the spectrum of $\hat{u}(t)$ and $\hat{y}(t) = c^{\mathrm{T}}\hat{x}(t)$ to the passband of the continuous-time filter (cf. [3]).

4. DEFINITIONS OF SNR

The following definitions for signal-to-noise ratios (SNR) are used in Sections 5 and 6. We measure SNR in dB (i.e. $10 \cdot \log_{10}(\mathrm{SNR})$). For the signal $\tilde{Y}_k$ we define

$$\mathrm{SNR}_{\mathrm{tot}} \triangleq \frac{\mathrm{E}\left[Y(t)^2\right]}{\sigma_Z^2 + \mathrm{E}\left[J_k^2\right]} \tag{24}$$

$$\mathrm{SNR}_Z \triangleq \frac{\mathrm{E}\left[Y(t)^2\right]}{\sigma_Z^2} \tag{25}$$

$$\mathrm{SNR}_J \triangleq \frac{\mathrm{E}\left[Y(t)^2\right]}{\mathrm{E}\left[J_k^2\right]}, \tag{26}$$

with $\mathrm{E}\left[J_k^2\right] = \sigma_D^2 \mathrm{E}\left[\dot{Y}(t)^2\right]$ (7).

For the SNR of the estimate $\hat{y}(t_k) \triangleq c^{\mathrm{T}}\hat{x}(t_k)$ of $Y(t_k)$ we define

$$\mathrm{SNR}_{\mathrm{out}} \triangleq \frac{\mathrm{E}\left[Y(t_k)^2\right]}{\mathrm{E}\left[\big(\hat{Y}(t_k) - Y(t_k)\big)^2\right]}. \tag{27}$$

Assuming $\mathrm{Re}(\lambda_i) < 0$ for all λ_i in (16) (i.e., a stable system), the mean vector and the covariance matrix of the message

$$\overrightarrow{\mu}_{X(\infty)} \triangleq \lim_{t\to\infty} \overrightarrow{\mu}_{X(t)} \tag{28}$$

converge to

$$\overrightarrow{m}_{X(\infty)} \triangleq \lim_{t\to\infty} e^{At} = 0 \tag{29}$$

$$\overrightarrow{V}_{X(\infty)} \triangleq \lim_{t\to\infty} \sigma_U^2 Q \overrightarrow{\Theta}(t) Q^{\mathrm{H}}. \tag{30}$$

$\overrightarrow{\mu}_{X(\infty)}$ represents the probability distribution of $X(t)$ in the absence of observations, therefore

$$\mathrm{E}[X(t)] = \overrightarrow{m}_{X(\infty)} = 0 \tag{31}$$

$$\mathrm{E}\left[X(t)^2\right] = \overrightarrow{V}_{X(\infty)}, \tag{32}$$

and thus

$$\mathrm{E}\left[Y(t)^2\right] = c^{\mathrm{T}} \mathrm{E}\left[X(t)^2\right] c \tag{33}$$

$$= c^{\mathrm{T}} \overrightarrow{V}_{X(\infty)} c \tag{34}$$

$$\mathrm{E}\left[\dot{Y}(t)^2\right] = c^{\mathrm{T}} \mathrm{E}\left[\dot{X}(t)^2\right] c \tag{35}$$

$$= c^{\mathrm{T}} A \overrightarrow{V}_{X(\infty)} \left(c^{\mathrm{T}} A\right)^{\mathrm{T}}. \tag{36}$$

5. THE ALGORITHM

We now describe an iterative algorithm to estimate the signal $Y(t_k)$ for all k. We assume the special case of uniform sampling with rate f_s, i.e., $t_k = k/f_s$, although the algorithm can easily be adjusted for non-uniform sampling.

Some remarks on notation: $x^{(\ell)}$ denotes the value of some variable x at the ℓ-th iteration and $\hat{x}$ denotes the estimate of some random variable X.

At each iteration ℓ, $\hat{x}(t_k)^{(\ell)}$ is calculated as described in Section 3. The estimates of $Y(t_k)$ and its derivatives are

$$\hat{y}(t_k)^{(\ell)} = c^{\mathrm{T}} \hat{x}(t_k)^{(\ell)} \tag{37}$$

$$\hat{\dot{y}}(t_k)^{(\ell)} = c^{\mathrm{T}} A \hat{x}(t_k)^{(\ell)}. \tag{38}$$

At the first iteration no estimates of the derivatives are available. Thus, $\sigma_J^2(t_k)^{(0)}$ is initialized using (36) to the expected average power of J_k which is constant for all k:

$$\sigma_J^2(t_k)^{(0)} = \sigma_D^2 c^{\mathrm{T}} A \overrightarrow{V}_{X(\infty)} \left(c^{\mathrm{T}} A\right)^{\mathrm{T}}. \tag{39}$$

This first iteration is similar to the algorithm described in [2]; the main difference is that we are not assuming strictly bandlimited signals.

In iteration $\ell = 1, 2, \ldots$, the value for $\sigma_J^2(t_k)^{(\ell)}$ is set based on the estimate of $\dot{Y}(t_k)$ of iteration $\ell - 1$ according to (7):

$$\sigma_J^2(t_k)^{(\ell)} = \sigma_D^2 \left(\hat{\dot{y}}(t_k)^{(\ell-1)}\right)^2 \tag{40}$$

$$= \sigma_D^2 \left(c^{\mathrm{T}} A \hat{x}(t_k)^{(\ell-1)}\right)^2 \quad \ell = 1, 2, \ldots. \tag{41}$$

The ratio between the input power and the average output power

$$\frac{\sigma_U^{2(\ell)}}{\sigma_Z^2 + \sigma_D^2 \frac{1}{N} \sum_{k=1}^{N} \left(c^{\mathrm{T}} A \hat{x}(t_k)^{(\ell-1)}\right)^2} \tag{42}$$

influences the spectrum of $\hat{x}(t)$ as mentioned in the end of Section 3. Thus, $\sigma_U^{2(\ell)}$ is chosen to keep (42) constant over all iterations.

For the numerical results given in Section 6, the values for σ_D^2, σ_Z^2 and $\sigma_U^{2(0)}$ are set equal to the values used for signal generation.

6. NUMERICAL RESULTS

To test the algorithm, sample signals $\tilde{y}_k$ have been generated according to the model in Figure 1 (cf. [3] on how to generate sample signals). For the continuous-time system a low-pass Butterworth filter of order 8 was used. Recall that $Y(t)$ is not strictly bandlimited but its spectrum is shaped by the spectrum of the continuous-time filter. We characterize the "bandwidth" of $Y(t)$ by the -3 dB frequency f_c of the Butterworth filter.

For signal generation, we chose the random delay D_k of the sampling instants to be uniformly distributed in the interval $\left(-\frac{0.25}{f_s}, \frac{0.25}{f_s}\right)$. Note that the estimation algorithm uses a Gaussian distribution for the time delay with

$$\mathrm{E}[D_k] = 0 \tag{43}$$

$$\mathrm{E}\left[D_k^2\right] = \frac{(0.25/f_s)^2}{3}. \tag{44}$$

The algorithm stops after the ℓ-th iteration if

$$\frac{\mathrm{SNR}_{\mathrm{out}}^{(\ell)}[\mathrm{dB}] - \mathrm{SNR}_{\mathrm{out}}^{(\ell-1)}[\mathrm{dB}]}{\mathrm{SNR}_{\mathrm{out}}^{(\ell)}[\mathrm{dB}] - \mathrm{SNR}_{\mathrm{out}}^{(0)}[\mathrm{dB}]} < 0.01. \tag{45}$$

For the data shown in Figure 3, each point in the plot was generated by 100 signals with $K = 100\ 000$ samples. The algorithm always stopped after three or four iterations.

Simulations were performed for three different values of the -3 dB frequency f_c (thus, different "bandwiths" of $Y(t)$). The results are shown in Figure 3, where $\mathrm{SNR}_{\mathrm{out}}$ of the estimates after the first and the last iteration are plotted for different values for SNR_Z. To show the overall improvement of the SNR, $\mathrm{SNR}_{\mathrm{tot}}$ of the observed, noisy samples $\tilde{y}_k$ is plotted as well.

Not surprisingly, the numerical results of the first iteration resemble the results in [2]. As mentioned in the introduction, the first iteration of our proposed algorithm is similar to the algorithm proposed in [2]. Since [2] assumes strictly bandlimited signals an exact comparison does not make sense.

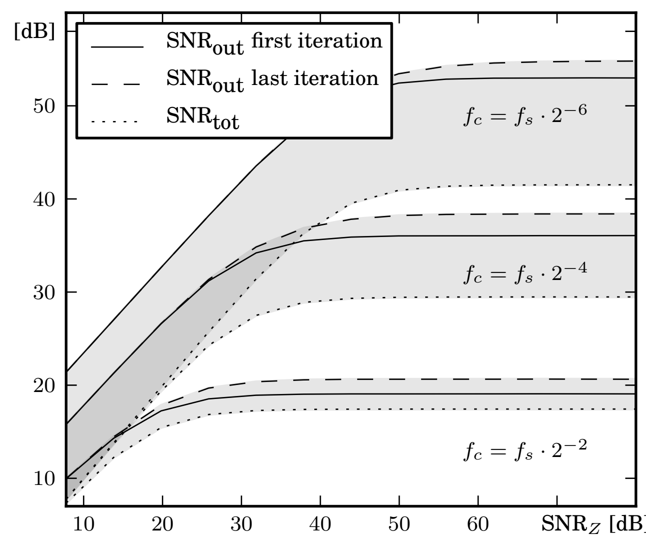


Figure 3: Empirical $\mathrm{SNR}_{\mathrm{out}}$ and $\mathrm{SNR}_{\mathrm{tot}}$ for different bandwith of $Y(t)$.

7. CONCLUSION

We have proposed an iterative algorithm to estimate a continuous-time signal from noisy discrete-time observations subject to clock jitter. The noise induced by the clock jitter was modeled to be dependent on the slope of the continuous-time signal at the sampling instant. We showed how to improve the estimate by iterative processing.

The algorithm is easily extended to non-uniform sampling. Also the estimation of the input signal $U(t)$ of the continuous-time filter is straightforward (cf. [3]).

ACKNOWLEDGEMENT

The authors would like to thank Mr. Daniel Baumann for the initial implementation of the algorithms of this paper and for providing first simulation results.

REFERENCES

[1] R.S. Prendergast and T.Q. Nguyen, "Optimal reconstruction of periodically sampled signals with probabilistic

timing delays," *Thirty-Eighth Asilomar Conference on Signals, Systems and Computers*, 2004, Vol. 2, pp. 2141–2145.

[2] A. Nordio, C.-F. Chiasserini, and E. Viterbo, "Signal reconstruction errors in jittered sampling," *IEEE Trans. Signal Proc.*, Vol. 57, No. 12, pp. 4711–4718, Dec. 2009.

[3] L. Bolliger, H.-A. Loeliger, and C. Vogel, "Simulation, MMSE estimation, and interpolation of sampled continuous-time signals using factor graphs," *Information Theory & Applications Workshop, UCSD, La Jolla, CA, USA*, 2010

[4] H.-A. Loeliger, J. Dauwels, Junli Hu, S. Korl, Li Ping, and F.R. Kschischang, "The factor graph approach to model-based signal processing," *Proceedings of the IEEE*, Vol. 95, No. 6, pp. 1295–1322, June 2007.