A COMPRESSIVE PHASE-LOCKED LOOP

Stephen R. Schnelle,[1] *J. P. Slavinsky,*[1] *Petros T. Boufounos,*[2] *Mark A. Davenport,*[3] *Richard G. Baraniuk*[1]

[1] Department of Electrical and Computer Engineering, Rice University, Houston, TX 77005

[2] Mistubishi Electric Research Laboratories, Cambridge, MA 02139

[3] Department of Statistics, Stanford University, Stanford, CA 94305

ABSTRACT

We develop a new method for tracking narrowband signals acquired via compressive sensing. The compressive sensing phase-locked loop (CS-PLL) enables one to track oscillating signals in very large bandwidths using sub-Nyquist sampling. A key feature of the approach is the fact that we perform the frequency tracking directly on the compressive measurements without ever recovering the signal. The CS-PLL has a wide variety of potential applications, including communications, phase tracking, and robust control.

***Index Terms*—** Compressive sensing, phase-locked loop, FM demodulation

1. INTRODUCTION

Compressive sensing (CS) is a recently developed field within signal processing that enables the acquisition and recovery of sparse signals without loss of information at a sampling rate significantly below the Nyquist rate. CS uses a randomized measurement system, and typically recovers the signal via convex optimization or one of a fleet of greedy recovery algorithms. Several hardware architectures have applied this theory to wideband analog signals.

Unfortunately, sparse recovery algorithms are relatively computationally expensive. For streaming applications such as radio receivers, low computational complexity and real-time recovery is paramount. Furthermore, the finite-dimensional nature of existing recovery algorithms requires that streaming (infinite-length) signals must be processed in finite-length blocks, often introducing significant input-output delay and blocking artifacts at the boundaries. Thus, classical CS recovery algorithms are not appropriate for most real-time applications.

In this paper we develop a phase-locked loop (PLL) architecture that extracts phase and frequency information directly from compressive samples of modulated signals [1]. This has a variety of applications involving frequency and phase tracking, such as the demodulation of frequency modulated (FM) signals. Since the modulated signal is never fully recovered, the CS-PLL offers computational advantages over streaming CS recovery algorithms, such as [2], that would require the use of a conventional PLL on the recovered signal. Thanks to the implementation of other signal processing operations in the compressive domain, such as filtering and detection systems [3], the CS-PLL can be integrated smoothly into devices such as wideband compressive radio receivers, which can track and monitor a wide range of radio frequencies in real-time.

A special case of the CS-PLL includes PLLs that use random sampling, which have been explored in previous works. For instance, a simple FPGA implementation of a phase-locked loop for quantized additive random sampling (ARS) was supported with numerical simulations for varying loop filter parameters in a random sampling PLL for synchronization applications in [4]. This paper generalizes this approach to more generic CS sampling schemes.

In the next section we establish our notational lingo and provide the relevant background on PLLs, CS, and practical compressive samplers. Section 3 introduces the CS-PLL, while Section 4 provides an analysis of its key features. Section 5 provides experimental results of the system, while Section 6 concludes the paper.

2. BACKGROUND

2.1. Phase-locked loop (PLL)

The phase-locked loop (PLL) is a well-established method for tracking the frequency and phase of a signal $x[n]$ using a feedback loop to continuously update an estimate of the parameters of the signal. Figure 1 shows a typical discrete-time real-valued PLL architecture. The phase detector and loop filter estimate the phase difference between $x[n]$ and a reference signal $u[n]$ by multiplying the two signals and low-pass-filtering the product to obtain

$$\theta[n] = \sum_k x[k]\, u[k]\, h[n-k], \tag{1}$$

where $h[n-k]$ is the impulse response of the low-pass filter. The phase estimate is used by an oscillator to produce the reference signal (with sinusoidal carrier)

$$u[n] = \cos(\omega n + \theta[n]). \tag{2}$$

The PLL works by adjusting the estimate of $\theta[n]$ until $x[n]$ and $u[n]$ are approximately orthogonal.

A second-order loop enables the PLL to track both a signal's phase and frequency, while higher-order systems can be useful for handling Doppler effects. A standard baseline model of a PLL has a loop filter described by the transfer function

$$H_l(z) = C_2 \frac{(z-1) + \frac{C_1}{C_2}}{(z-1)}, \tag{3}$$

where $C_1 = \omega_n^2$, and $C_2 = 2\zeta\omega_n$. For more information on PLLs, see [5].

This material is based upon work supported by the following grants: DARPA N66001-08-1-2065, ONR N00014-08-1-1112, ONR N00014-11-1-0714, NSF IIS-1124535, ARL W911NF-09-1-0383, NSF CCF-0926127, ONR N00014-10-1-0989, DARPA/ONR N66001-11-1-4090, NSF OCI-1041396,AFOSR FA9550-09-1-0432, NSF CCF-0431150, NSF CCF-0728867, ONR N00014-08-1-1067, ARO W911NF-07-1-0185, DARPA N66001-11-C-4092, NSF DMS-1004718, NSF Graduate Research Fellowship Program (NSF 0940902), NDSEG Fellowship Program, and the Texas Instruments Leadership University Fellowship Program. PTB performed part of this work while at Rice University; PTB is currently supported in full by Mitsubishi Electric Research Laboratories.

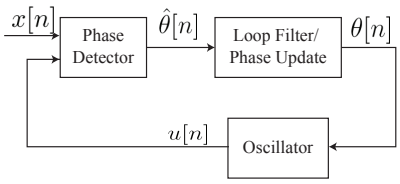

Fig. 1. Basic discrete time PLL design.

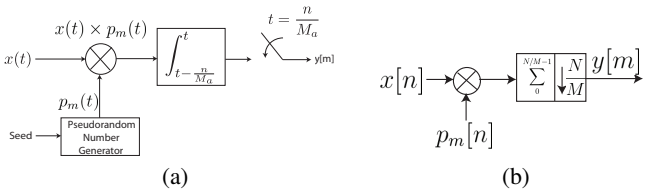

Fig. 2. Random demodulator (RD) in (a) continuous-time and (b) discrete-time.

2.2. Compressive sensing

In the ecumenical CS framework [6], we acquire a signal $x \in \mathbb{R}^N$ that is sparse or compressible in some basis, via the measurements $y = \Phi x$, with Φ an $M \times N$ matrix with $M \ll N$ which represents the sampling system. The reduction in measurements is enabled by the properties of Φ, in particular the *restricted isometry property* (RIP). First we define $\Sigma_K = \{x \in \mathbb{R}^N : \|x\|_0 \leq K\}$ where $\|x\|_0 := |\text{supp}(x)|$ counts the number of non-zero entries of x, i.e., Σ_K is the set of all K-sparse signals in $\mathbb{R}^N$. The RIP of order K of a matrix Φ implies that

$$(1-\delta)\|x\|_2^2 \leq \|\Phi x\|_2^2 \leq (1+\delta)\|x\|_2^2 \tag{4}$$

holds for some constant $\delta \in (0,1)$ over all $x \in \Sigma_K$, (i.e., Φ acts as an approximate isometry on the set of vectors that are K-sparse).

2.3. Practical compressive samplers

While CS theory often focuses on random matrix constructions of Φ, there are also a variety of practical CS sampling methods, including *random demodulation* (RD) [7], *random sampling* [8], and the *compressive multiplexer* (CMUX) [9]. The random demodulator, which we use in our simulations yonder is described here for reference. An analog input $x(t)$ is modulated with a pseudo-random square wave with amplitude ± 1s, called the *chipping sequence* $p_m(t)$, with transition frequency at or above the Nyquist rate N_aHz of the input signal. Next, integration over a time period $1/M_a$ is performed on the mixed signal, and lastly this result is sampled at M_aHz $< N_a$Hz. The architecture of the RD is depicted in Figure 2(a), whereas a discrete-time model (which will be used inside the CS-PLL) is shown in Figure 2(b).

If desired, multiple random demodulators can be offset and their outputs interleaved to obtain a higher rate of compressive samples without increasing the complexity of ADC hardware (as explained in [2]). We define N/M as the overall compression ratio in this scenerio, whereas L is the window size of an individual random demodulator. In the case of a single random demodulator RD, we have a window of length $L = N/M$ where the $p_m[k]$ are ± 1. Alternative chipping sequences, such as one consisting of Gaussian random coefficients can also be used.

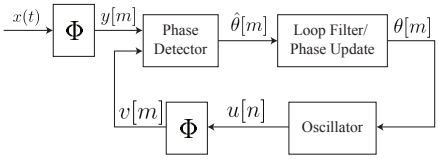

Fig. 3. Block diagram of CS-PLL.

3. CS-PLL SYSTEM DESCRIPTION

We now introduce the CS-PLL ,a new family of digital and mixed analog/digital PLLs based on CS. Recall that the phase estimate update calculation in the basic PLL is the (weighted) inner product between the Nyquist-rate samples $x[n]$ of the signal we wish to estimate/track and the estimated signal $u[n]$ generated by the oscillator. If both $x[n]$ and $u[n]$ can be represented by not only their Nyquist rate samples but also their (lower rate) compressive samples (with $x[n]$ producing $y[m]$ and $u[n]$ producing $v[m]$), then the RIP guarantees that the inner product between their compressive samples $y[m]$ and $v[m]$ will be very close to the inner product between their Nyquist rate samples $x[n]$ and $y[n]$ [10]. Hence the angle is maintained between the vectors.

Leveraging this insight, we introduce two compressive samplers into the basic PLL system to create the CS-PLL shown in Figure 3. The first sampler acquires compressive samples $y[m]$ of the continuous-time input signal $x(t)$. (Note that the architecture could also easily accommodate a discrete-time input signal $x[n]$. For simplicity we focus in this paper only on the continuous-time case.) The second converts the oscillator"s Nyquist rate samples $u[n]$ into the compressive samples $v[m]$. The sampling operation in the loop—which is the discrete-time model for the compressive sampling used to acquire the signal—is causal and efficient to implement.

As with the classical PLL, the CS-PLL computes the phase estimate inner product

$$\theta[m] = \sum_k y[k]v[k]h[m-k], \tag{5}$$

with index m denoting the lower sampling rate and $h[n]$ mimicing the response of the higher rate filter in the Nyquist-rate PLL. A nonlinear and/or time-varying filter could also be used here, for example if we had a signal with varying message bandwidth, just as in the traditional case.

4. ANALYSIS AND MODELLING OF THE CS-PLL

In this section, we analyze the properties of the CS-PLL and use a simple model to study its stability properties.

4.1. Estimation

We first show that the CS-PLL is a maximum likelihood estimator of a modulated signal's phase and frequency. To begin, suppose that the CS-PLL is frequency locked, and thus we are only trying to estimate the signal's phase [5]. This is a safe assumption in typical PLL analysis; large frequency offsets prevent the loop from locking, whereas small offsets are handled with other structures such as second-order loops. Our signal of interest

$$x(t) = \cos(\omega_c t + \theta) \tag{6}$$

(with ω_c the continuous time frequency), is compressively sampled and corrupted with additive white Gaussian noise (AWGN) $w_i[m]$. Thus, we obtain

$$y[m] = \sum_{k=-\infty}^{\infty} p_m[k]\cos(\omega k + \theta) + w_i[m] \tag{7}$$

where $p_m[k]$ denotes the pseudo-random coefficients for sample m (i.e., element (m,k) of the sampling matrix Φ corresponding to the compressive sampler) and ω is the Nyquist-rate discrete-time frequency corresponding to ω_c. Although $p_m[k]$ may be a pseudorandom sequence, the sequence is known to the system, and thus our sole source of randomness is the input noise $w_i[m]$ over a set of M measurements. AWGN added to $x(t)$ is simply scaled AWGN on $y[m]$.

We now show that the CS-PLL can be viewed as a maximum likelihood (ML) estimator (and, assuming independent measurements, an MMSE estimator as well) by considering a signal with slowly varying phase over M measurements. The tracking error variance of the unbiased estimator $\sum_{k=-\infty}^{\infty} p_m[k]\cos(\omega k+\tilde{\theta})$ assuming each measurement is independent is

$$\sigma^2 = \sum_{m=1}^{M}\left(y[m] - \sum_{k=-\infty}^{\infty} p_m[k]\cos(\omega k+\tilde{\theta})\right)^2 \tag{8}$$

where $\tilde{\theta}$ represents the quantity we are estimating. Taking the derivative of σ^2 with respect to $\tilde{\theta}$

$$\frac{\partial\sigma^2}{\partial\tilde{\theta}} = 2\sum_{m=1}^{M}\left(y[m]\sum_{k=-\infty}^{\infty} p_m[k]\sin(\omega k+\tilde{\theta}) - \sum_{k=-\infty}^{\infty} p_m[k]\cos(\omega k+\tilde{\theta}) \times \sum_{k=-\infty}^{\infty} p_m[k]\sin(\omega k+\tilde{\theta})\right) \tag{9}$$

Two terms are present in each measurement m: the correlation of $y[m]$ with the compressive measurements of a 90-degree out-of-phase reference output and a offset term independent of the output. The correlation term confirms that the compressive sampler model in the loop should match that used on the input. For the offset term, we note that if the $p_m[k]$ terms are generated independently of each other with mean zero, then when we treat $p_m[k]$ as random and consider the average case of M samples by finding the expected value over $p_m[k]$. The second terms in the summation reduce to $\sum_{k=-\infty}^{\infty}(p_m[k])^2\cos(2(\omega k+\tilde{\theta}))$, which would be filtered out by the low-pass filter in the CS-PLL. Ignoring this offset simplifies our system's complexity and corresponding analysis model and does not drastically affect performance. The CS-PLL naturally acts to achieve average case performance over time with its loop filters and adaptive nature.

4.2. Linear model for the CS-PLL

Next we develop a simplified linear model for the PLL and determine its transfer function and other characteristics. Given an RD with chipping sequence $p_m\left[\frac{N}{M}m+k\right]$ and Nyquist rate samples $x[k] = \sin(\omega k + \theta_1[k])$ and $u[l] = \cos(\omega l + \theta_2[l])$ with ω the discrete-time Nyquist frequency, the multiplier output

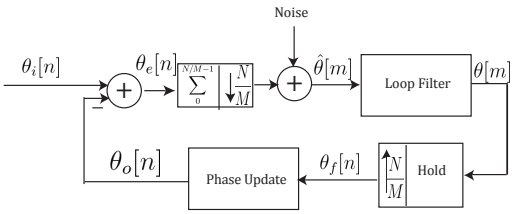

Fig. 4. Linear sample-and-hold model for CS-PLL stability analysis using RDs.

$x[m]u[m]$ consists of two terms $\sum_{k=0}^{L-1}\left(p_m\left[r_k\right]\right)^2 x\left[r_k\right] u\left[r_k\right]$ and $\sum_{\substack{k=0\\k\neq l}}^{L-1}\sum_{l=0}^{L-1} p_m\left[r_k\right] x\left[r_k\right] p_m\left[r_l\right] u\left[r_l\right]$ where L is the length of the RD band, and $r_k = \frac{N}{M}m + k$. The first sum accumulates Nyquist-rate samples, and the second accumulates cross-term noise $w_c[m]$. $w_c[m]$ which is zero-mean and uncorrelated with the input (and feedback) due to the randomness introduced by the RD. Using ±1 modulation in the RD implies that $\left(p_m\left[r_k\right]\right)^2 = 1$.

Since $w_c[m]$ is zero-mean and uncorrelated with the input and the feedback term, we can formulate a linear model for the phase as well. As before, our model includes an FIR filter with L random taps equal to the $\left(p_m\left[\frac{N}{M}m+k\right]\right)^2$ and downsampling by $\frac{N}{M}$ following the phase detector. For a sampler that contains only a single RD, $L = N/M$. This FIR filter model has sharp nulls at multiples of the aliasing frequency, indicating that we continue to remove narrowband noise that CS is designed to prevent. The loop filter operates at the lower sampling rate and a sample and hold element is added to this input to the oscillator to return the sampling rate to the original high rate. The linearized model is shown in Fig. 4(b). Similar to [11], we can write the system equations.

The open loop transfer function is

$$H_{op} = \frac{z^{-1}}{\frac{N}{M}(1-z^{-1})}\left(\frac{1-z^{-N/M}}{1-z^{-1}}\right)\left(\frac{1-z^{-L}}{1-z^{-1}}\right)H_l(z^{N/M})$$

with $H_l(z^{N/M})$ the loop filter. The closed loop transfer function is

$$H_{cl}(z) = \frac{F(z)}{\frac{N}{M}(1-z^{-1})^2 + F(z)} \tag{10}$$

where $F(z) = C_2 z^{-1}(1 - z^{-N/M}(1-\frac{C_1}{C_2}))(1+\cdots+z^{-(L-1)})$. Using the loop filter described by (3), we choose filter parameters C_1 and C_2 such that the poles of the transfer function are within the unit circle and the loop filter has the appropriate characteristics for the error signal. Although solving this high-order polynomial analytically for bounds is difficult, we find heuristically that reducing C_1 and C_2 by the compression ratio N/M makes the system stable in the case of a single random demodulator. Assuming the loop filter is designed properly to ensure stability, a zero steady state phase error for an initial frequency offset will result, found by solving $e_{ss} = \lim_{z\to1}(z-1)(1-H_{cl}(z))$. Because of the accumulate-and-dump component in this model, the loop order is $L + N/M$, verifying the potential for increasing instability as the compression grows. If we were to interleave multiple random demodulators (with L a multiple of N/M), the stability of the system becomes a much greater concern.

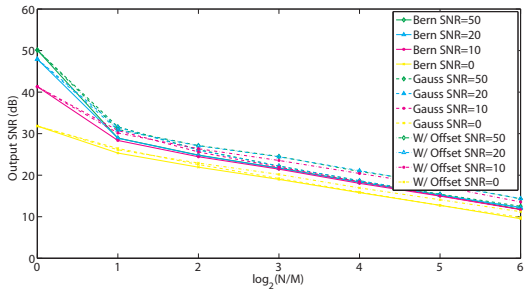

Fig. 5. Output SNR for a random demodulator when using Bernoulli coefficients, normalized Gaussian coefficients, and when including the offset in the phase detector, each for varying input SNRs

5. SIMULATIONS

To simulate the performance of the CS-PLL, we use a sampling rate of 2.048MHz, and a oscillator frequency of 120kHz, and simple second-order filter with loop bandwidth of 10kHz (adjusted in the discrete domain for the appropriate compressive sampling rate). Our first input signal is a compressively sampled FM-modulated 2.5kHz signal with frequency deviation of 1.6kHz. Data is sampled using a random demodulator with ± 1 taps.

We quantify the CS-PLL's performance in terms of its output SNR and compare with a more conventional Nyquist-rate PLL for various and compression factors averaged over 25 trials. Note that the performance of the conventional PLL corresponds to a logarithmic compression factor of N/M of 0, shown at the left edge of the plots. We do not use an MMSE error criterion due to the inherent delay in discrete-time PLLs. Instead, the output SNR is measured by dividing the signal power by the noise power in the frequency spectrum, computed over an average 250 Hz band around the signal of interest (the noise is relatively flat over the spectral region of interest, and traditionally out-of-band noise is filtered at the output of a PLL).

From Figure 5, we see that at high input SNR, there is an initial sharp drop in the SNR when compression is used due to the introduced cross-term noise, but then it smooths to a more gradual 3 dB per factor of 2 compression due to noise-folding. Wideband noise folding continues to be an inevitable consequence of using CS, just as it appears in traditional CS recovery algorithms [12].

The performance of the CS-PLL improves slightly when an additional offset factor is added to the phase detector based on the current phase estimate derived with our maximum likelihood estimator in (9), though in many practical scenerios it can be ignored.

Compression rates in the wideband receiver are not limited to powers of 2. In Figure 6, we apply the CS-PLL to a simulated high-power cordless phone input signal of 25 dB input SNR with a compression ratio of 20. The output of the CS-PLL closely resembles the result of FM demodulation using a Hilbert transform and phase unwrapping.

6. CONCLUSIONS

This paper has developed a framework for constructing phase-locked loops that perform phase and frequency tracking directly on compressive measurements. The ideas could be extended relatively easily to any variants of the PLL that use a multiplier phase detector. A variety of applications could utilize the CS-PLL, ranging from communication schemes using phase or amplitude modulation, GPS, and industrial control applications. Future work includes a more thorough probabilistic non-linear analysis of the transient characteristics of the CS-PLL, such as lock time, cycle slipping, and instantaneous phase error.

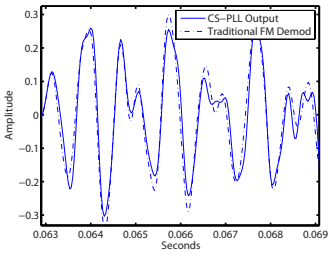

Fig. 6. CS-PLL output compared to traditional FM demodulation of the Nyquist-rate samples.

7. REFERENCES

[1] S. Schnelle, "A compressive phase-locked loop," M.S. thesis, Rice University, 2011.

[2] P. Boufounos and M. Asif, "Compressive sampling for streaming signals with sparse frequency content," in *Proc. IEEE Conf. Inform. Science and Systems (CISS)*, Princeton, NJ, Mar. 2010.

[3] M. Davenport, S. Schnelle, J. P. Slavinsky, R. Baraniuk, M. Wakin, and P. Boufounos, "A wideband compressive radio receiver," in *Proc. Military Comm. Conf. (MILCOM)*, San Jose, CA, Oct. 2010.

[4] M. Sonnaillon, R. Urteaga, and F. Bonetto, "Software PLL based on random sampling," *IEEE Trans. Inst. Meas.*, vol. 59, no. 10, pp. 2621–2629, 2010.

[5] J. Crawford, *Advanced Phase-Lock Techniques*, Artech House, 2008.

[6] E. Candès and T. Tao, "Decoding by linear programming," *IEEE Trans. Inform. Theory*, vol. 51, no. 12, pp. 4203–4215, 2005.

[7] S. Kirolos, J. Laska, M. Wakin, M. Duarte, D. Baron, T. Ragheb, Y. Massoud, and R. Baraniuk, "Analog-to-information conversion via random demodulation," in *Proc. IEEE Dallas Circuits and Systems Work. (DCAS)*, Dallas, TX, Oct. 2006.

[8] A. Gilbert, S. Muthukrishnan, and M. Strauss, "Improved time bounds for near-optimal sparse Fourier representations," in *Proc. SPIE Optics Photonics: Wavelets*, San Diego, CA, Aug. 2005.

[9] J. P. Slavinsky, J. Laska, M. Davenport, and R. Baraniuk, "The compressive mutliplexer for multi-channel compressive sensing," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing (ICASSP)*, Prague, Czech Republic, May 2011.

[10] M. Davenport, P. Boufounos, M. Wakin, and R. Baraniuk, "Signal processing with compressive measurements," *J. Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 445–460, 2010.

[11] F. Gardner, *Phaselock Techniques*, Wiley, 2005.

[12] M. Davenport, J. Laska, J. Treichler, and R. Baraniuk, "The pros and cons of compressive sensing for wideband signal acquisition: Noise folding vs. dynamic range," *Preprint*, 2011.