A Compressed Sensing Parameter Extraction Platform for Radar Pulse Signal Acquisition

Juhwan Yoo, Christopher Turnes, Eric Nakamura, Chi Le, Stephen Becker, Emilio Sovero, Michael Wakin, Michael Grant, Justin Romberg, Azita Emami-Neyestanak, and Emmanuel Candès

***Abstract*—In this paper we present a complete (hardware/software) sub-Nyquist rate ($\times$13) wideband signal acquisition chain capable of acquiring radar pulse *parameters* in an *instantaneous* bandwidth spanning 100 *MHz*–2.5 *GHz* with the equivalent of 8 ENOB digitizing performance. The approach is based on the alternative sensing-paradigm of Compressed-Sensing (CS). The hardware platform features a fully-integrated CS receiver architecture named the random-modulation pre-integrator (RMPI) fabricated in Northrop Grumman's 450 *nm* InP HBT bipolar technology. The software back-end consists of a novel CS parameter recovery algorithm which extracts *information* about the signal without performing full time-domain signal reconstruction. This approach significantly reduces the computational overhead involved in retrieving *desired information* which demonstrates an avenue toward employing CS techniques in power-constrained real-time applications. The developed techniques are validated on CS samples *physically measured* by the fabricated RMPI and measurement results are presented. The parameter estimation algorithms are described in detail and a complete description of the physical hardware is given.**

***Index Terms*—Compressed sensing, Indium-Phosphide, Parameter Estimation, Random-Modulation Pre-Integration**

I. Introduction

A principal goal in the design of modern electronic systems is to acquire large amounts of information quickly and with little expenditure of resources. In the wireless technology sector, the goal of maximizing information throughput is illustrated by the strong interest in RF sensing and spectral applications that require instantaneous bandwidths of many GHz. Such systems have applications ranging from scientific instrumentation to electronic intelligence. Although some solutions already exist, their large size, weight, and power consumption make more efficient solutions desirable.

At present, realizing high bandwidth systems poses two primary challenges. The first challenge comes from the amount of power required to operate back-end ADCs at the necessary digitization rate. This issue is so significant that the remaining elements of the signal chain (RF front-end, DSP core, etc.) are often chosen based upon an ADC that is selected to be compatible with the available power budget [1]. The second challenge comes from need to store, compress, and post-process the large volumes of data produced by such systems. For example, a system that acquires samples at a rate of 1 *Gsps* with 10 *bits* of resolution will fill 1 *Gb* of memory in less than 1 *s*. In light of the ever growing demand to capture higher bandwidths, it would seem that a solution at the fundamental system level is needed to address these challenges.

Some promise for addressing these challenges comes from the theory of *compressed sensing* (CS) [2]–[6]. CS has recently emerged as an alternative paradigm to the Shannon-Nyquist sampling theorem, which at present is used implicitly in the design of virtually all signal acquisition systems. In short, the CS theory states that signals with high overall bandwidth but comparatively low information level can be acquired very efficiently using randomized measurement protocols. The requisite sampling rate is merely proportional to the information level, and thus CS provides a framework for *sub-Nyquist rate* signal acquisition. As we discuss further in Sec. II, aliasing is avoided because of the random nature of the measurement protocol.

The emergence of the CS theory is inspiring a fundamental re-conception of many physical signal acquisition and processing platforms. The beginning of this renaissance has already seen the re-design of cameras [7], medical imaging devices [8], and RF transceivers [9]–[11]. However, the benefits of CS are not without their costs. In particular, the task of reconstructing Nyquist-rate samples from CS measurements requires solving an inverse problem that cannot be addressed with simple linear methods. Rather, a variety of nonlinear algorithms have been proposed (see, e.g., [12]–[14]). While the speed of these methods continues to improve, their computational cost can still be appreciably greater than many conventional algorithms for directly processing Nyquist-rate samples. This matter of computation, if not addressed, potentially limits the wide-spread application of CS architectures

J. Yoo and A. Emami-Neyestanak are with the Department of Electrical Engineering at the California Institute of Technology, Pasadena, CA, e-mail: juhwan@caltech.edu

C. Turnes and J. Romberg are with the School of Electrical and Computer Engineering at the Georgia Institute of Technology, Atlanta, GA, e-mail: cturnes@gatech.edu

M. Wakin is with the Department of Electrical Engineering and Computer Science, Colorado School of Mines, Golden, CO

S. Becker and M. Grant are with the Department of Applied and Computational Mathematics at the California Institute of Technology, Pasadena, CA; S. Becker is also with the Laboratoire Jacques-Louis Lions at Paris 6 University, Paris, France

C. Le, E. Nakamura, and E. Sovero are with the Northrop Grumman Corporation, Redondo Beach, CA, e-mail:eric.nakamura@ngc.com; Emilio Sovero is now at Waveconnex Inc., Westlake Village, CA, e-mail: emilio.sovero@waveconnex.com

E. Candès is with the Departments of Mathematics and Statistics at Stanford University, Stanford, CA

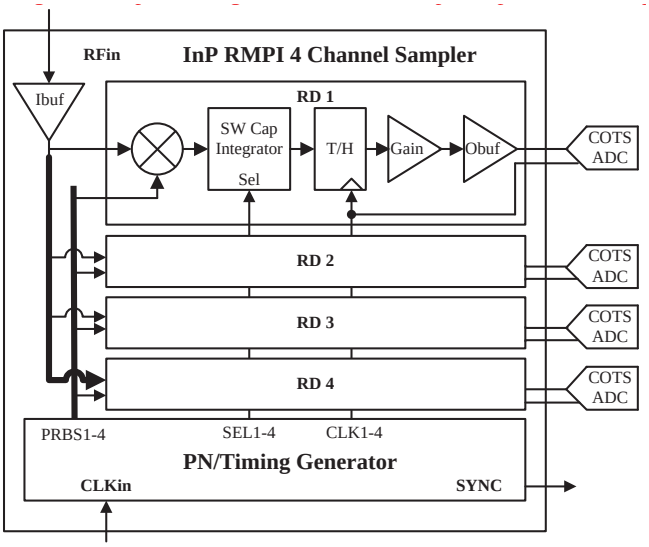
in power constrained real-time applications.

In this paper we address these issues by presenting a complete, novel signal acquisition platform (both hardware and software) that is capable—in certain applications—of estimating the desired signal parameters *directly* from CS measurements [15]. In the spirit of compressive signal processing [16], our approach takes the principal motivation of CS one step further and aims to eliminate the overhead of first reconstructing the Nyquist-rate signal samples before applying conventional DSP techniques for parameter extraction.

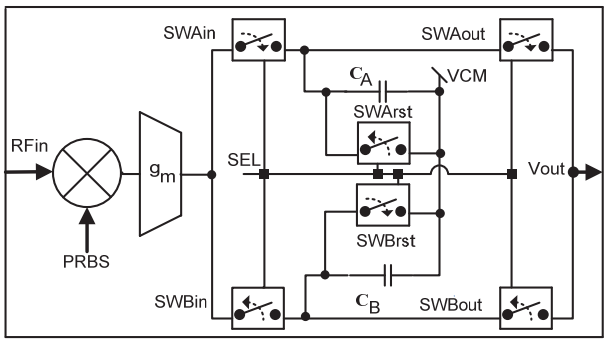
On the hardware side, we present a fully integrated wideband CS receiver called the random-modulation pre-integrator (RMPI) [9,10,12,17]. We fabricate this device with Northrop Grumman's 450 *nm* InP HBT bipolar process. On the software side, we focus on signal environments consisting of radar pulses and present a novel algorithm for extracting radar pulse parameters—carrier frequency (CF), phase θ_0, amplitude A_0, time-of-arrival (TOA), and time-of-departure (TOD)—directly from CS measurements. (The exact signal model is described in Sec. IV.)

Our complete system is capable of recovering radar pulse parameters within an effective instantaneous bandwidth (EIBW) spanning 100 *MHz*—2.5 *GHz* with a digitizing performance of 8 ENOB. We validate the system by feeding the fabricated RMPI with radar pulses and using the physically digitized CS measurements to recover the parameters of interest.

An outline of this paper is as follows: Sec. II provides a brief background on CS and a description of the high-level operation of the RMPI, Sec. III provides a complete description of the hardware platform used to encode the CS samples, Sec. IV provides details of the parameter estimation algorithms, and Sec. V presents measurement results.



(a) System Block Diagram



(b) RD Channel Block Diagram

Fig. 1: (a) Simplified block diagram of 4-channel RMPI. The analog-signal path of each RD channel is identical, however the timing signals they receive in operation are different. (b) Functional diagram of the mixer and integrator circuits.

II. The RMPI

A. Compressed Sensing

CS at its heart relies on two concepts: *sparsity* and *incoherence* [5]. Sparsity captures the idea that many high-dimensional signals can be represented using a relatively small set of coefficients when expressed in a properly chosen basis. Incoherence captures the idea of dissimilarity between any two representations; two bases are said to be incoherent if any signal having a sparse expansion in one of them must be dense in the other. An example of an incoherent pair comes from the classical time-frequency duality. A sparse signal time—e.g., a Dirac-delta function—has a dense spectrum. Similarly, a single tone is sparse in the Fourier domain but dense in time.

The key observation underlying CS is that when a signal is sparse in some basis, it can be acquired by taking a small number of measurements that are incoherent with its sparse basis [18]. Often this incoherence is achieved by incorporating randomness into the measurement process. There are many possibilities for implementing incoherent random measurements; a convenient and admissible choice for hardware implementation is to correlate the input signal (in our case, a time-windowed version of the input signal) with a pseudo-random binary sequence (PRBS) [4]. We refer the reader to [5] and references therein for additional information about the mathematical theory of CS.

B. A Brief History and Description of the RMPI

Almost simultaneously with the introduction of CS [2], a number of CS-based signal-acquisition architectures were proposed. Some of the more well-known proposals include: the Random Demodulator (RD) [17,19,20], the Random-Modulation Pre-Integrator (RMPI) [9,10,21], the Non-Uniform Sampler (NUS) [4,22], Random Convolution [23], the Modulated Wideband Converter (MWC) [24], and many others [25]—for a comprehensive overview see [12]. The basic function that all of these systems implement is to cor-

relate of the input signal $x(t)$ with an incoherent, randomly generated set of "basis" elements over a fixed time window.

The RMPI is one of the most direct physical implementations of the CS concept; it is composed of a parallel set of RDs driven by a common input. (See Fig. 1a, which is described more fully in Sec. III.) Each RD is driven by a distinct PRBS $p(t)$; it uses this PRBS to modulate the incoming signal $x(t)$, integrates the result over a time interval of length T_{int}, and then digitizes the output at a rate $f_{\text{ADC}} = 1/T_{\text{int}} \ll f_{\text{nyq}}$. In our RMPI, $T_{\text{int}} = 52 \cdot T_{\text{nyq}}$ with $f_{\text{nyq}} = 1/T_{\text{nyq}} = 5$ *GHz*. Thus $f_{\text{ADC}} = 1/(52 \cdot T_{\text{nyq}}) = 96.154$ *MHz*; the aggregate back-end sampling rate $f_s = 384.616\ Msps$, which corresponds to undersampling the Nyquist rate by a factor of 13. Aliasing is avoided in this measurement scheme because (i) the modulation with the PRBS will spread the spectrum of any tone (including high-frequency ones) across the entire band so that one can effectively subsample, and (ii) the input signal is again assumed to obey some model (aside from merely being bandlimited).

Letting x denote a time-windowed vector of Nyquist-rate samples of the input signal $x(t)$, we can implicitly model the RMPI measurement process as multiplication of x by a matrix Φ having $13\times$ fewer rows than columns. Each row of this matrix corresponds to a portion of the PRBS sequence used in a specific integration window from a specific channel. As an example, if we consider a sample vector x of length $N = 1040$, the matrix Φ will be block-diagonal, with each block having 4 rows (representing the parallel operation of the 4 channels) and $T_{\text{int}}/T_{\text{Nyq}} = 52$ columns (representing an integration window of 52 Nyquist bins). The rows of each block contain ± 1 entries, and the overall matrix will be composed of $NT_{\text{Nyq}}/T_{\text{int}} = 20$ blocks (one for each integration window). Denoting the vector of measurements as y, the RMPI mode of acquisition can be modeled as $y = \Phi\mathbf{x}$ where $\Phi \in \mathbb{R}^{80\times 1040}$.

We point out that high-fidelity recovery/extraction of information from CS measurements requires precise knowledge of the system transfer function Φ. Thus, practical deviations from the block-diagonal ± 1 model described above must be taken into account. For the measurements presented in this paper, we construct a model of our system's Φ matrix by feeding in sinusoidal tones and using the output measurements to characterize the system's impulse response.

III. Hardware Implementation Description

A. Architecture and Operation

The RMPI presented in this work was realized with the proprietary Northrop Grumman (NG) 450 *nm* InP HBT bipolar process [26]. The process features a 4-layer metal stack with an f_T and $f_{\max} > 300$ *GHz*. Fig. 1a shows the block diagram of the IC containing the input buffer driving the common node of the four RD channels and the timing generator. The timing generator is responsible for generating the pseudo-random bit sequences (PRBS) and the clocking waveforms to coordinate the track-and-hold (T/H) and integration operations. All analog and digital signal paths are implemented differentially to improve common-mode noise rejection and increase linearity of the system. The analog path up to the integrator was designed for a 2.5 *GHz* bandwidth. The ensuing integration reduces the bandwidth containing significant energy content. The circuits following the integrator are designed to meet the settling requirements of the reduced bandwidth. A 5 *GHz* master clock reference (CLKin) is used to toggle the PRBS generators and is chosen to be the Nyquist-rate of the input bandwidth [12,17]. The T/H operate at $1/52$ the master clock frequency ($= 96.154$ *MHz*). A switched-capacitor interleaving integrator [27] is used so that one capacitor can be reset while the second integrates the mixer output. Finally, an output buffer is designed to drive the ADC with the correct swing and common-mode voltage. The chip was designed for a full-scale input amplitude of $0.5\ V_{pp}$ differential and $1\ V_{pp}$ differential output. In operation, the RMPI circuit takes the analog input signal, buffers it, and distributes the buffered signal to each of the 4 channels. In each channel, the signal is multiplied by one of 4 orthogonal PRBS—each of which is a 3276 *bit* long Gold code [28]. The resulting product is integrated by one of two sets of interleaved capacitors for exactly one frame (52 CLKin cycles). At the end of the integration period the signal is sampled and then held for 26 CLKin cycles to allow the external ADC to digitize the signal for post-processing. Immediately after the signal is sampled, the capacitor begins discharging and the second capacitor begins integrating the next frame (see Fig. 1b). The interleaved integration capacitors are used to avoid missing frames due to the reset operation. Additionally, the sampling instants for each channel are staggered to create more diversity in the windowed integrations obtained.

B. Analog Signal Path

The input buffer is a differential pair with emitter degeneration and $50\ \Omega$ termination at each single-ended input. It has a gain of 3 *dB*, a 2.5 *GHz* bandwidth, 70 *dB* SFDR, and a full-scale differential input amplitude of $0.5\ V_{pp}$. The random modulation is performed by a standard differential Gilbert mixer with the PRBS generator driving the top pair and the analog input driving the bottom differential pair. Emitter degeneration is used on the bottom differential pair to improve linearity. To reduce noise, the mixer was designed to have about 20 *dB* gain to offset the attenuation from the integrator. The output of the mixer is integrated using interleaved switched capacitors as shown in Fig. 1b and Fig. 2 and has a 12.5 *MHz* pole frequency. Input and output select switches are closed to route the mixer output current to the integration capacitor as well as to read out the capacitors with the T/H circuit. When the reset switch is on, the integration capacitor voltage is reset to zero. At the end of each integration cycle (one frame $= 10.4$ *ns*), the output of

the integrator is sampled by the T/H and held for 5.2 *ns*. This ensures that the external ADC has enough time to digitize the held voltage.

The T/H was implemented using the switched emitter follower topology with gain ≈ 1. To minimize the hold-mode feed-through, small feed-forward capacitors were inserted [29]. The switched emitter follower was chosen in favor over the more conventional diode bridge switch for its smaller footprint and comparatively low parasitic capacitance. The amplifier after the T/H has a gain of 2. In addition to emitter degeneration, diode connected transistors are used in the output load to cancel the input differential pair V_{be} modulation and improve linearity. The output driver was designed to be DC-coupled to the external ADC and have 70 *dB* SFDR and a 1 V_{pp} swing. In order to save power, 200 Ω on-chip termination resistors were used on each side to exploit the relatively long settling time.

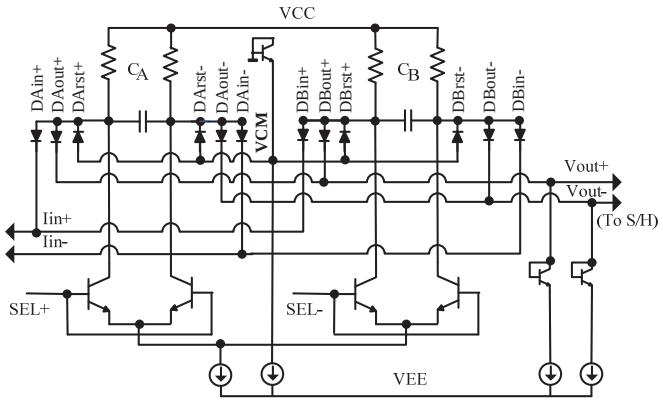


Fig. 2: Simplified schematic of interleaved switched capacitor integrator. The diodes act as switches to configure capacitors for integration or reset it based on the level of the control signal (SEL). When SEL is high, integrator A is resetting and integrator B is integrating. When SEL+ is low, integrator B is resetting and integrator A is integrating.

C. PRBS & Timing Generator

A master clock is applied to CLKin, from which all required timing signals are generated. The input clock buffer was biased with a relatively high power to reduce jitter. In addition, it has 4 separate output emitter followers to mitigate the deleterious effects of cross-talk on the clock jitter. Each emitter follower provides a low jitter signal to re-clock the PRBS input before it is mixed with the RF input in each channel.

The PRBS signals are generated with two 6 *bit* PRBS generating linear-feedback shift-registers (LFSR(s)) [30]. One PRBS generator (PN6A in Fig. 3) is programmed to cycle every 52 CLKin cycles while the second (PN6B in Fig. 3) is allowed to cycle through all 63 states. The 2 PRBS generator outputs are combined to generate 4 orthogonal $52 \times 63 = 3276$ *bit* long Gold code sequences. PN6A is also used to generate the T/H clocks (divide by 52) and select signal (divide by 104) for the switched capacitor integrator. Both PN generators also output a sync pulse used to synchronize the system. The output pulse from PN6B is re-clocked with the pulse from PN6A to produce a sync pulse that is 52 CLKin cycles long once every 3276 cycles. The synchronization pulse is essential to provide precise knowledge of the chipping sequence used in each integration window. This relative alignment information is crucial for signal-recovery and parameter estimation.

CLKin and the RF input are located on opposite sides of the chip to minimize coupling. Special attention was paid to the routing of the PRBS, T/H clocks, and select signals to minimize clock/data coupling among the four channels. A simplified block diagram of the PRBS/timing generator is shown in Fig. 3. The timing generator block was designed to operate at speeds in excess of 5 *GHz* and consumes 2.8 *W* when operated at the designed rate.

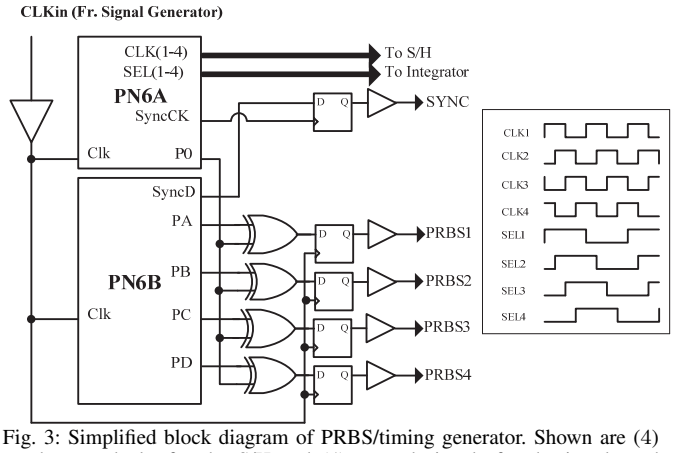


Fig. 3: Simplified block diagram of PRBS/timing generator. Shown are (4) quadrature clocks for the S/H and (4) control signals for the interleaved capacitors. Combinational logic is used to prevent the illegal start up condition of the PN generators and to generate divide by 52 and divide by 104 used as control signals in the S/H and in the interleaved integrators respectively.

D. Performance Analysis

Simulation validation was done by performing transient-based two-tone inter-modulation distortion simulations in the Cadence design environment. Noise simulations were performed using the periodic steady-state (PSS) mode of spectre. The RMPI sampling system, including the off-chip ADCs consumed 6.1 *W* of power. We point out that this system was designed as a proof-of-concept and was not optimized for power. Thus, caution should be used when comparing the CS system in this work to conventional counterparts. For example, the use of an InP process in this work leads to power penalties compared to the CMOS RMPI (which consumes ≈ 0.5 *W*) reported in [9,10], which is also similarly unoptimized, due to the availability of static logic. A die photo of the fabricated chip is shown in Fig. 4.

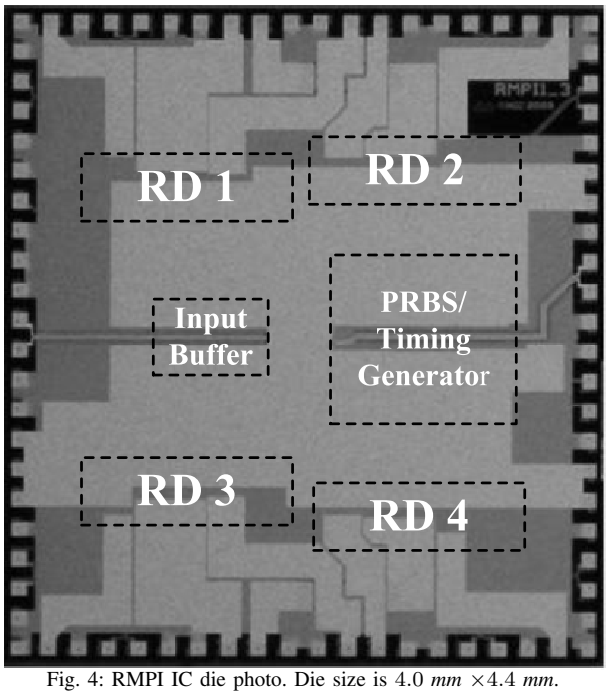


Fig. 4: RMPI IC die photo. Die size is 4.0 *mm* ×4.4 *mm*.

IV. Pulse-Descriptor Word (PDW) Extraction

Having described the acquisition system, we now present algorithms for detecting radar pulses and estimating their parameters, referred to as pulse-descriptor words (PDW), from randomly modulated pre-integrated (RMPI) samples. The detection process is based on familiar principles employed by detectors that operate on Nyquist samples. Our algorithms use a combination of template matching, energy thresholding, and consistency estimation to determine the presence of pulses. By using all three of these methods, we gain robust detection at the cost of a number of tunable parameters that must be set to appropriate levels depending on the application and sensing equipment. The general procedure consists of three steps: first, we estimate the carrier frequency and energy of a potential pulse segment at various time shifts; second, based on consistency in frequency estimates and large enough pulse energies, we apply criteria to determine if a pulse is present; finally, for detected pulses we use our parameter estimation methods to refine our carrier frequency, amplitude, phase, time-of-arrival, and time-of-departure estimates.

The remainder of the section elaborates on the procedure and is arranged as follows. First, we describe our methods of parametric estimation, focusing in particular on carrier frequency estimation. After describing how we can reliably estimate the carrier frequency of a signal from compressive measurements, we then explain how we use such estimations to form a detection algorithm that jointly uses energy detection and consistency of our frequency measurements. We then describe how we perform parametric estimation on compressive samples while simultaneously removing a band in which a known interfering signal is present. Finally, we combine the detection algorithm we have formulated with the cancellation technique to present an algorithm capable of detecting multiple overlapping pulses.

A. General Parametric Estimation

Our general parameter estimation problem can be stated as follows. We consider signals $x_0(t)$ drawn from one of a collection of (low-dimensional) subspaces $\{\mathcal{S}_\alpha\}$ indexed by a parameter set $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_K)$. Given the measurements $y = \Phi[x_0] + \text{noise}$, we search for the set of parameters corresponding to the subspace which contains a signal which comes closest to explaining the measurements y. We solve

$$\hat{\alpha} = \arg\min_{\alpha} \left(\min_{x \in \mathcal{S}_\alpha} \|y - \Phi[x]\|_2^2 \right). \tag{IV.1}$$

The inner optimization finds the signal in $\mathcal{S}_\alpha$ that is most consistent with the measurements for a fixed α; the outer optimization compares these best fits for different α.

The inner optimization program, which is the classical "closest point in a subspace" problem, has a well-known closed form solution as it is easily formulated as a least-squares problem.

Let $u_{\alpha,1}(t), u_{\alpha,2}(t), \ldots, u_{\alpha,d}(t)$ be a basis for the space $\mathcal{S}_\alpha$, meaning that

$$\begin{aligned} &x_0(t) \in \mathcal{S}_\alpha \\ &\Rightarrow x_0(t) = a_1 u_{\alpha,1}(t) + a_2 u_{\alpha,2}(t) + \cdots + a_p u_{\alpha,d}(t), \end{aligned}$$

for some unique $a_1, a_2, \ldots, a_d \in \mathbb{R}$. If we define V_α to be the $M \times d$ matrix containing the inner products between each pair of RMPI test functions $\phi_m(t)$ and basis functions $u_{\alpha,i}(t)$,

$$V_\alpha = \begin{bmatrix} \langle \phi_1, u_{\alpha,1} \rangle & \langle \phi_1, u_{\alpha,2} \rangle & \cdots & \langle \phi_1, u_{\alpha,d} \rangle \\ \langle \phi_2, u_{\alpha,1} \rangle & \langle \phi_2, u_{\alpha,2} \rangle & \cdots & \langle \phi_2, u_{\alpha,d} \rangle \\ \vdots & \vdots & \cdots & \vdots \\ \langle \phi_M, u_{\alpha,1} \rangle & \langle \phi_1, u_{\alpha,2} \rangle & \cdots & \langle \phi_M, u_{\alpha,d} \rangle \end{bmatrix} \tag{IV.2}$$

then we can re-write (IV.1) as

$$\begin{aligned} \hat{\alpha} &= \arg\min_{\alpha} \ \|y - V_\alpha (V_\alpha^T V_\alpha)^{-1} V_\alpha^T y\|_2^2 \\ &= \arg\min_{\alpha} \ \|(I - P_\alpha) y\|_2^2, \end{aligned} \tag{IV.3}$$

where $P_\alpha = V_\alpha (V_\alpha^T V_\alpha)^{-1} V_\alpha^T$ is the *orthogonal projector* onto the column space of V_α. It is worth mentioning that when the measurement noise consists of independent and identically distributed Gaussian random variables, the result $\hat{\alpha}$ in (IV.3) is the *maximum likelihood estimate* (MLE). When the noise is correlated, we may instead pose the optimization in terms of a weighted least squares problem.

In Sec. IV-B and Sec. IV-C below, we will discuss the particular cases of frequency estimation for an unknown tone, and time-of-arrival estimation for a square pulse modulated to a known frequency. In both of these cases, we are trying to estimate one parameter and the underlying subspaces $\mathcal{S}_\alpha$ have dimension $d = 2$. Moreover, the functional $\|(I - P_\alpha) y\|_2^2$

can be efficiently computed over a fine grid of values for α using the Fast Fourier Transform (FFT).

When there are multiple parameters, performing the joint minimization over $\alpha = (\alpha_1, \alpha_2, \ldots)$ can be computationally prohibitive. In Sec. IV-D, we present a heuristic algorithm for estimating the key parameters of a Doppler pulse (carrier frequency, time-of-arrival, and pulse length) that operates by looking for consistent frequency estimates over consecutive windows of time. The frequency and time-of-arrival estimators play a central role in this algorithm.

In Sec. IV-F, we show how the algorithm can be modified to accommodate overlapping pulses and strong narrowband interferers.

B. Carrier Frequency, Amplitude, and Phase

In this section, we consider the task of estimating the frequency of a pure (Doppler) tone from the observed RMPI measurements. The algorithms developed here play a central role in the detection process as well, as the frequency estimation procedure is fundamental to determining the presence of pulses. We also describe how to estimate the amplitude and phase once the carrier frequency (CF) is known. A similar technique can be used to estimate the amplitude and phase while estimating the time of arrival (TOA) and time of departure (TOD) as well.

For our CF estimation task, we suppose that we observe an M-vector y (consisting of a concatenation of all samples from the RMPI device over a certain interval) given by $y = \Phi[x_0] + \text{noise}$, where x_0 consists of N Nyquist samples of a sinusoid

$$\begin{aligned} x_0[n] &= A_0 \cos(2\pi f_0 t_n + \theta_0), \\ t_n &= n/f_{\text{nyq}}, \ n = 1, \ldots, N, \end{aligned}$$

and f_{nyq} is the Nyquist frequency. The amplitude A_0, phase θ_0, and frequency f_0 of the sinusoid are *a priori* unknown, and we make the implicit assumption that the TOA corresponds to $n = 1$ and the TOD corresponds to $n = N$.

It is convenient to rewrite x_0 as a weighted sum of a cosine and sine with zero phase,

$$x_0[n] = a_1 \cos(2\pi f_0 t_n) + a_2 \sin(2\pi f_0 t_n), \qquad \text{(IV.4)}$$

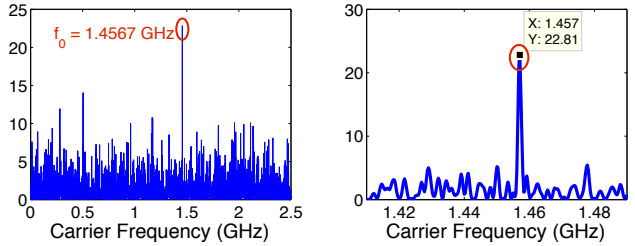
where we can relate a_1, a_2 to A, θ by realizing that they are the real and complex parts of the phasor $Ae^{j\theta}$: $a_1 + ja_2 = Ae^{j\theta}$. The subspaces $\mathcal{S}_f$ we search are thus spanned by two vectors u_1 and u_2 with $u_1[n] = \cos(2\pi f_0 t_n)$ and $u_2[n] = \sin(2\pi f_0 t_n)$. Since we have discretized the signal $x(t)$ through its Nyquist samples, our measurement process is modeled through a matrix Φ, the rows of which are the basis elements ϕ_k. For a given frequency f, we define V_f as in (IV.2) and solve (IV.3) to obtain the MLE estimate of the basis expansion coefficients in the subspace $\mathcal{S}_f$.

Rather than dealing with continuously variable frequency, we define a fine grid of frequencies between 0 and $f_{\text{nyq}}/2$. As an equivalent alternative to minimizing the outer optimization in (IV.1), for each f_k in the grid (and corresponding subspace $\mathcal{S}_k$) we may instead compute the quantity

$$W_k = \frac{\|V_{f_k} \alpha_{f_k}\|^2}{\|y\|^2}$$

and choose the frequency f_k that maximizes W_k.

In practice, since we obtain values for W_k on a grid of frequencies, we can realize tangible gains by treating these values as samples of a continuous function $W(f)$ and interpolating in between the samples of $W(f)$ once the maximum has been localized; that is, we can "super-resolve" $W(f)$ using cubic interpolation. Fig. 5 illustrates the function $W(f)$ for an example Doppler tone.



(a) (b)

Fig. 5: Plot of the energy function $W(f)$ for measurements derived from a noisy Doppler tone at 1.4567 *GHz* over (a) the entire range of allowable frequencies and (b) frequencies close to the true CF. In (a) we see that the energy functional is clearly maximized in an area near the true CF, and (b) shows that the maximum of the energy function occurs at 1.4571 *GHz*. For this example, we used a total of 3315 Nyquist samples with a sampling rate of 5 *GHz*, so our intrinsic frequency resolution is on the order of 5 *GHz*/3315 $\approx$ 1.5 *MHz*. The estimate of the carrier frequency is well within this resolution.

Once the carrier frequency is estimated as $\hat{f}$, we estimate the tone amplitude as $\hat{A} = \sqrt{(\alpha_{\hat{f}})_1^2 + (\alpha_{\hat{f}})_2^2}$ and the tone phase as $\hat{\theta} = \tan^{-1}((\alpha_{\hat{f}})_2/(\alpha_{\hat{f}})_1)$.

C. Time of Arrival/Departure

Next we describe how the TOA of a pulse can be estimated. We now assume that we have RMPI samples of a pure tone at a *known* frequency windowed by a step function. That is, we observe an M-vector $y = \Phi[x_0] + \text{noise}$, where $x_0[n]$ has the form

$$\begin{aligned} x_0[n] &= \mu(t_n - \tau_0) \cdot A_0 \cos(2\pi f_0 t_n + \theta_0), \qquad \text{(IV.5)} \\ t_n &= n/f_{\text{nyq}}, \ n = 1, \ldots, N, \end{aligned}$$

and $\mu(\cdot)$ is the unit step, $\mu(t) = 1$ for $t \geq 0$ and is zero for $t < 0$. While we assume that the frequency f_0 is known (or has been estimated as in Sec. IV-B), the amplitude, phase, and TOA τ_0 are unknown. The processes we describe are perfectly analogous for TOD estimation if we consider a "flipped" version of $x_0[n]$.

As in (IV.4), we can write x_0 as a weighted sum of a windowed sine and cosine:

$$\begin{aligned} x_0[n] &= a_1 u_1[n] + a_2 u_2[n], \\ u_1[n] &= \mu(t_n - \tau_0)\cos(2\pi f_0 t_n), \\ u_2[n] &= \mu(t_n - \tau_0)\sin(2\pi f_0 t_n). \end{aligned}$$

Then for any given candidate TOA τ, the subspace $\mathcal{S}_\tau$ is spanned by the vectors

$$u_{\tau,1}[n] = \mu(t_n - \tau)\cos(2\pi f_0 t_n)$$

and

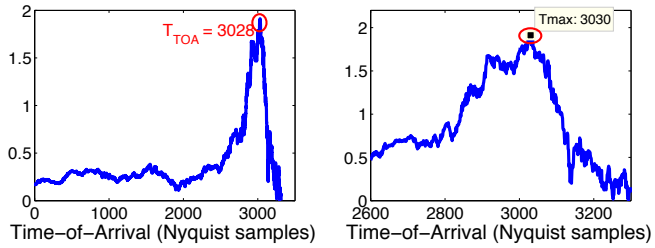
$$u_{\tau,2}[n] = \mu(t_n - \tau)\sin(2\pi f_0 t_n).$$

We can again construct the matrix V_τ of (IV.2) and complete the MLE estimate of α_τ by solving (IV.3) and subsequently (IV.1)

Again, rather than deal with continuously variable τ we define a grid of times τ_k. In practice, using the grid of Nyquist sample locations $\tau_k = t_k$ is sufficient. As in the case of carrier frequency, rather than choose the subspace with the smallest value of the norm in (IV.1), we instead solve (IV.3) for each grid point and then select the subspace $\mathcal{S}_k$ yielding the largest value of

$$E(\tau_k) = \frac{\|V_{\tau_k}\hat{\alpha}_{\tau_k}\|^2}{\|y\|^2},$$

which is an equivalent solution. We may once again super-resolve by treating these values as samples of a continuous function. Fig. 6 illustrates the function $E(\tau)$ for an example Doppler tone.



(a) (b)

Fig. 6: Plot of the energy function $E(\tau)$ for measurements derived from a noisy Doppler tone at 1.4567 *GHz* arriving at Nyquist sample $n = 3028$ over (a) the entire range of sample indices and (b) sample indices close to the true TOA. In (a) we see that the energy functional is clearly maximized in an area near the true TOA, and (b) shows that the maximum of the energy function occurs at $n = 3030$. Since the sampling rate for this example is 5 *GHz*, this corresponds to an error of 400 *ps*.

D. Pulse Detection

With our estimation techniques explained we next describe a pulse detection algorithm that takes a stream of RMPI samples and classifies each as either having a "pulse present" or "no pulse present." We start by assuming that only one pulse is present at any given time. In the next section we will describe an extension of this algorithm that can account for multiple simultaneous pulses.

We model the pulses we are trying to detect as continuous functions of the form

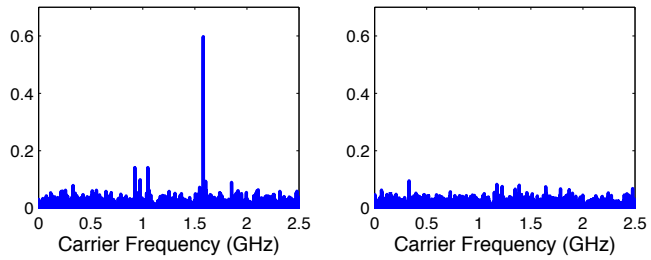
$$p_k(t) = A_k w_R(t, \tau_{k0}, \tau_{k1})\cos(2\pi f_{k0} t + \theta_k),$$

where A_k is the pulse amplitude, τ_{k0} the TOA, τ_{k1} the TOD, $w_R(t, \tau_{k0}, \tau_{k1})$ is the rectangular window $w_R(t, \tau_{k0}, \tau_{k1}) = u(t - \tau_{k0}) - u(t - \tau_{k1})$, f_{k0} the CF, and θ_k the phase. We assume that the input to our acquisition system consists of the Nyquist samples of some number of these pulses; that is,

$$x[n] = \sum_k p_k(nT_s),$$

where T_s is the sampling period. The vector of RMPI measurements we receive is the result of a linear matrix Φ applied to this input together with additive noise: $y = \Phi x + \text{noise}$. Our task is to determine how many pulses are present and to estimate the parameters of each pulse we find.

We use the CF estimation method of Sec. IV-B as a building block. The general approach of the detection algorithm is to divide the RMPI samples up into overlapping blocks and estimate how well the measurements corresponding to each block can be explained by the presence of a single tone. This is done by assuming a pulse is present in each block, estimating its CF, and determining how well the observed measurements agree with measurements generated by this pulse. As is shown in Fig. 7, if a pulse is indeed present at the estimated CF, it should account for a reasonably large portion of the measurement energy.



(a) (b)

Fig. 7: Fraction of measurement energies that are explained by frequencies up to 2.5 *GHz* for the case where (a) there is a 1.581 *GHz* tone and noise present and (b) there is only noise present. The noise energy is equally spread out over the band, where the tone energy is concentrated at one frequency.

Once we obtain the CF estimates, we look for consistency from block to block. If neighboring blocks have CF estimates that are consistent in value and tones at these frequencies account for a considerable portion of their measurement energies, then we are confident that a pulse is indeed present. If the CF estimates vary across neighboring blocks or contain insufficient energy in a single frequency, we consider there

to be insufficient evidence to indicate that there is a pulse present. Fig. 8 illustrates consistency in frequency estimation and energy proportion when pulses are present. In practice, the blocks near the end of a pulse may account for a smaller percentage of their measurement energies with a single tone, since they may contain some RMPI samples that correspond only to noise. To counteract this effect we use a weighted average across blocks consistent in their CF estimates.

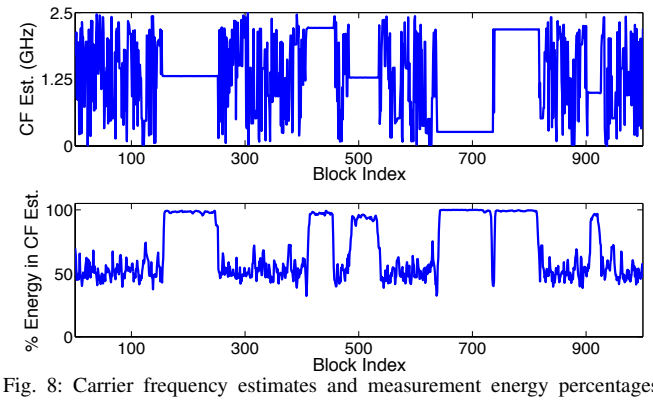


Fig. 8: Carrier frequency estimates and measurement energy percentages for block shifts of RMPI measurements. In this simulation, several pure-tone pulses are present at various times. The blocks corresponding to RMPI samples that cover the time support of the pulse contain consistent CF estimates which account for a reasonably large portion of the RMPI measurement energy. When pulses are absent, the CF estimates are erratic and account for considerably less energy in the measurements.

Once we have obtained pulse detections, we can further resolve their parameters. For example, suppose we have detected a pulse that starts at RMPI sample index k and has length P. We create a vector with the following RMPI samples:

$$y_s = \begin{bmatrix} y_{k-P/2} & y_{k-P/2+1} & \cdots & y_{k+3P/2-1} \end{bmatrix}^T.$$

While we are uncertain of the exact TOA and TOD of the pulse, the middle portion of the detected pulse (corresponding to samples $k + P/4$ to $k + 3P/4$) almost certainly contains only samples where the pulse is active. We use these samples to refine and super-resolve our carrier frequency estimate. Empirically, we find that our initial estimates for the TOA and TOD can often dramatically under/overestimate the correct values. Accordingly, we use the samples $k - P/2$ to $k + P/2 - 1$ to refine and super-resolve our TOA estimate and $k + P/2$ to $k + 3P/2 - 1$ to refine and super-resolve our TOD estimate. For longer pulses (large P), this means that we check a larger number of potential locations for our TOA and TOD estimates. We could instead fix a certain number of samples before and after each pulse detection to check, invariant of the detected pulse length, but we have found experimentally that making the search length dependent on the detected pulse length results in more accurate estimation. Once the TOA, TOD, and CF estimates have been refined, we calculate our amplitude and phase estimates.

Supposing we have a stream of RMPI measurements $y_1, y_2, \cdots, y_B$ over some time period, where each y_i is a vector of m RMPI samples at a given sample time, the complete detection algorithm is outlined in Alg. 1.

Algorithm 1 RMPI Pulse Parameter Extraction Algorithm

1: Choose a set of block lengths, each of which corresponds to a certain number of RMPI samples per channel.
2: For fixed block length L and for each sample time $k = 1, \ldots, B$, find the measurement vector $\tilde{y}_k$ containing the samples from time $k-(L-1)/2, \ldots, k+(L-1)/2$ and the corresponding portion of the measurement matrix $\tilde{\Phi}_k$.
3: Estimate the CF for each block by following the procedure in §IV-B, calling the frequency estimate F_k and the corresponding fraction of the energy it accounts for $P_k = P(F_k)/\|\tilde{y}_k\|_2^2$.
4: Determine for which blocks $|F_k - F_{k-1}|$ is below some threshold. Call each sequence of blocks with consistent frequency estimates a segment.
5: In each segment, take a weighted average of the P_k values. Keep all segments for which this average exceeds a certain threshold.
6: Repeat steps 2-5 for different values of L, creating a list of potential pulse segments for each block length.
7: Merge any segments that are close together in time and have similar CF estimate.
8: Keep only the remaining segments in the merged list that are longer than a pre-determined minimum signal length.
9: Super-resolve the amplitude, phase, TOA, and TOD can then using the techniques in Sec. IV-B and Sec. IV-C.

E. Complexity

For the measurement system we consider, Φ consists of shifted repetitions of a matrix Φ_0 that produces 63 separate 4-channel RMPI samples (for a total of 252 rows). For CF estimation, we can precompute the product $V_f = \Phi U_f$ for each frequency f in our grid by taking the real and imaginary parts of the FFT of each row of Φ. In fact, since Φ contains repetitions of Φ_0 we need only take the FFT of the 252 rows of Φ_0, which we may then shift through complex modulation. This calculation does not depend on the measurement data, and therefore can be done offline as a precomputation.

For each window length L (typically between 5-11 RMPI samples) and each window shift, we have to compute $P_f \tilde{y}_k$ for each f in the grid. For a given frequency f, we have precomputed the matrix V_f. We have $4L$ samples (since there are 4 channels per measurement), so V_f is $4L \times 2$ and $V_f^T V_f$ is a 2×2 system. We can explicitly invert $V_f^T V_f$ using the 2×2 matrix inverse formula, and all other calculations involve a small number of $4L$-point inner products. The number of frequencies we test is proportional to the number of Nyquist

samples N, and thus the cost of the frequency estimation for each sliding window shift is $\mathcal{O}(NL)$. Since the number of window shifts is equal to the number of RMPI samples, this is our per-sample computational cost. If we use multiple window lengths, our complexity is $\mathcal{O}(N\sum_i L_i)$.

F. Cancellation and multiple pulses

In this section, we focus on how to remove contributions from certain frequency bands in the RMPI measurements. This allows us to remove the effects of interferers that occupy a known, fixed frequency band. It will also allow us to remove the contributions of pulses that we have already detected in order to detect additional pulses that may occur at simultaneous points in time.

We assume that we have a signal interfering with the underlying signal whose parameters we wish to estimate, and that this interferer has energy only in a specific, fixed, and known frequency band. Our procedure for nulling the interfering signal involves the computation of discrete prolate spheroidal sequences (DPSS) [31,32][1]. A DPSS is essentially a set of functions that can best express signals of specified duration whose frequency content is restricted to a certain bandwidth. These can be modulated to be centered at any frequency, and thus can serve as a basis for compact signals whose energy occupies a specific band.

Suppose we have a DPSS with R elements that serve as a basis for signals of length N. We can express the DPSS as an $N \times R$ matrix V, whose R columns are the elements of the sequence. Then the interfering signal z can be written as a linear combination of these elements $z = Va$ for some $a \in \mathbb{R}^R$. The contribution from the interfering band in a set of measurements can be modeled as

$$y_i = \Phi V a.$$

Then we can estimate the portion of the measurements y that correspond to frequency content outside the interfering band as

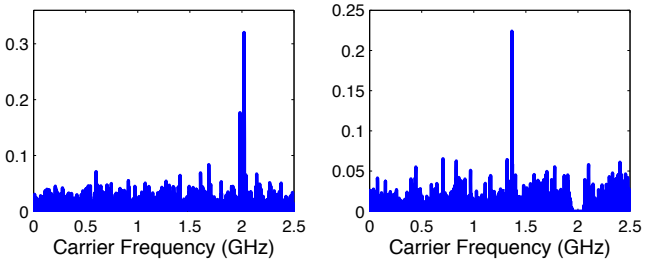
$$\tilde{y} \quad = \quad (I - \Phi V(V^T\Phi^T\Phi V)^{-1}V^T\Phi^T)y = \Psi y.$$

The operator Ψ can be used to remove the contributions of the interfering band in the measurements y when we run our estimation methods. This allows us to effectively operate as if the interferer is absent. The matrix Ψ does not depend on the measurements y but merely the RMPI matrix Φ, and therefore can be precomputed. Fig. 9 shows how the use of the nulling operator aids in removing interfering bands.

We can use similar concepts to detect multiple overlapping pulses. After the detection algorithm of Section IV-D, we can run a second pass of the algorithm. During this second pass, when we calculate the CF estimates for each block we do so by first constructing a nulling operator Ψ_k that cancels the contributions of any of the previously detected pulse segments that are within the block and then perform CF estimation using the modified measurements $\Psi_k\tilde{y}_k$. When we compute the metric $P(F_k)$, we use $\Psi_k\tilde{y}_k$ in place of $\tilde{y}_k$ so that the quantity expresses the fraction of the energy of the potential pulse relative to the *nulled measurements*.

The nulling operators Ψ_k that we construct will cancel a band, rather than a single frequency. The bandwidth and block length affect the number of elements in the DPSS that we require to cancel within a certain accuracy. Since the DPSS is designed to cancel a band rather than a single frequency, we must make an assumption as to how close in CF two simultaneously overlapping pulses are allowed to be; for our 5 *GHz* system we assume that overlapping pulses are at least 10 *MHz* apart in their carrier frequencies. Fig. 10 shows an example of the two-stage detector for overlapping pulse data.

The additional cost introduced by nulling detected pulses is dominated by the computation of the DPSS functions. This computation is dependent on the detected pulse length. However, if it is known that the pulses we are to detect are of a specific length, it may be possible to precompute the DPSS functions. In this case, the nulling produces introduces few extra calculations.



(a) (b)

Fig. 9: Fraction of measurement energies that are explained by frequencies up to 2.5 *GHz* for the case where (a) no nulling is used and there is strong interfering signal between 1.976 and 2.026 *GHz* and (b) the nulling operator is used to cancel out the interfering band. The nulling operator allows us to estimate the CF of the underlying tone, which is 1.367 *GHz*. In (b) the energy in the interfering band has virtually disappeared.

V. Verification of Hardware

A. Measurement Test Setup Description

In order to test the performance of the radar-pulse parameter estimation system-composed of the RMPI sampling hardware and the PDW extraction algorithm §IV, we ran a set of over 686 test radar pulses composed of permutations of A_0, θ_0, CF, TOA and TOD through the RMPI and estimated the varied parameters from the compressed-samples digitized by the RMPI.

[1]These are also known as *Slepian sequences.*

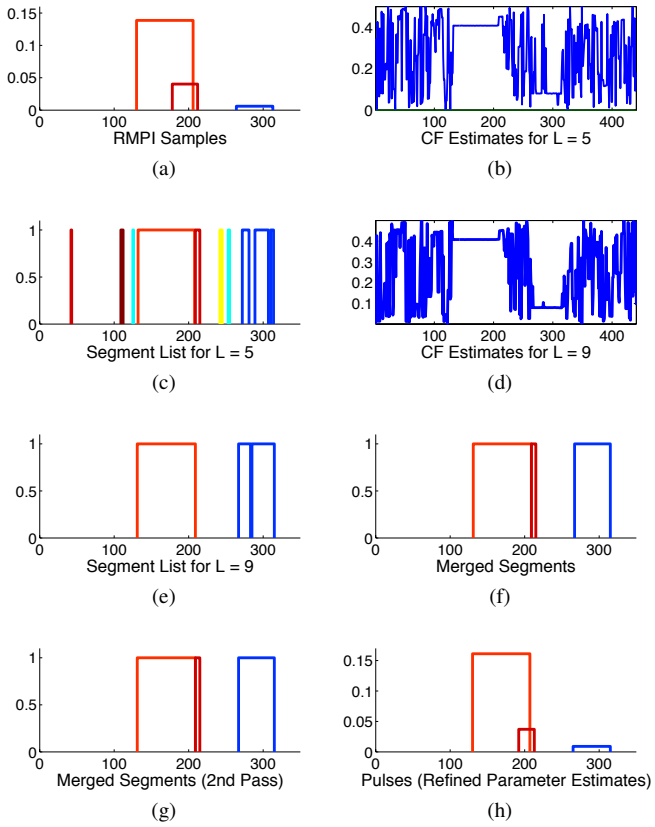
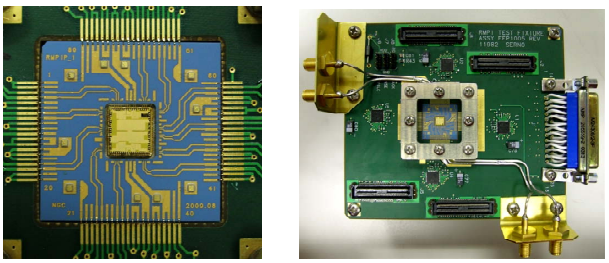


Fig. 10: Stages of the pulse detection and parameter estimation algorithm for overlapping pulse data: (a) source pulses; the pulse heights indicate their amplitude and the color of the pulse is reflective of the CF of the pulse, with red closer to 2.5 *GHz* and blue closer to 0 *Hz*; (b) CF estimates for blocks using 5 RMPI samples; (c) segments detected based on consistent CF estimates for blocks of size 5 with no phase or amplitude estimation; (d) CF estimates for blocks of size 9; (e) segments detected based on consistent CF estimates for blocks of size 9 with no phase or amplitude estimation; (f) the merged segments; (g) the merged segments after the second pass is completed; (h) the detected pulses with refined estimates of their parameters.



(a) RMPI IC Mount (b) RMPI Digitizer Board

Fig. 11: Assembled RMPI IC/Digitizer Interface. The board is 5 *inches* × 5 *inches*. The ADC board has 4 12 *bit* ADCs with output bits routed to 4 data-connectors that are acquired with a Logic Analyzer

Fig. 12 shows a block diagram for the test setup used for the RMPI. The input clock and data to the RMPI were driven differentially and AC-coupled. An Arbitrary Waveform Generator AWG with an output sampling rate of 10 *Gsps* was used to output the pulses of interest. The stimulus was input into the RMPI whose outputs were then sampled by external ADCs located on a custom digitizing PCB shown in Fig. 11b: the RMPI IC is mounted on a a low-temperature co-fired ceramic (LTCC) substrate shown in Fig. 11a which is placed in the center of the digitizing board. The digitized samples were then transferred to a PC where the PDW extraction algorithm was used to estimate the signal parameters.

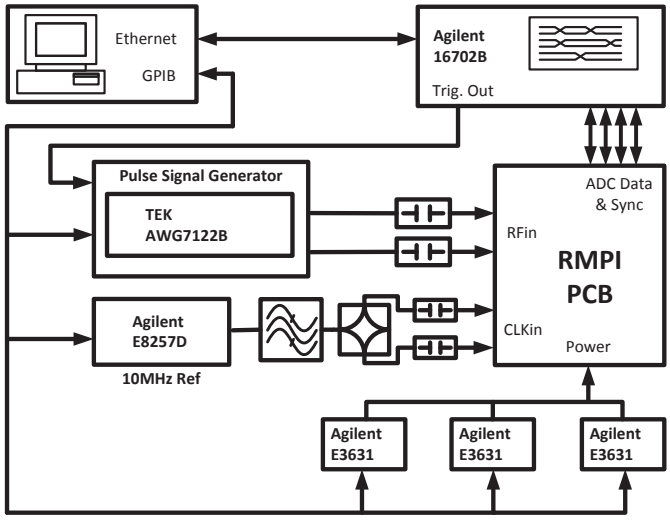


Fig. 12: Block Diagram of RMPI Test Setup

B. Parameter Estimation

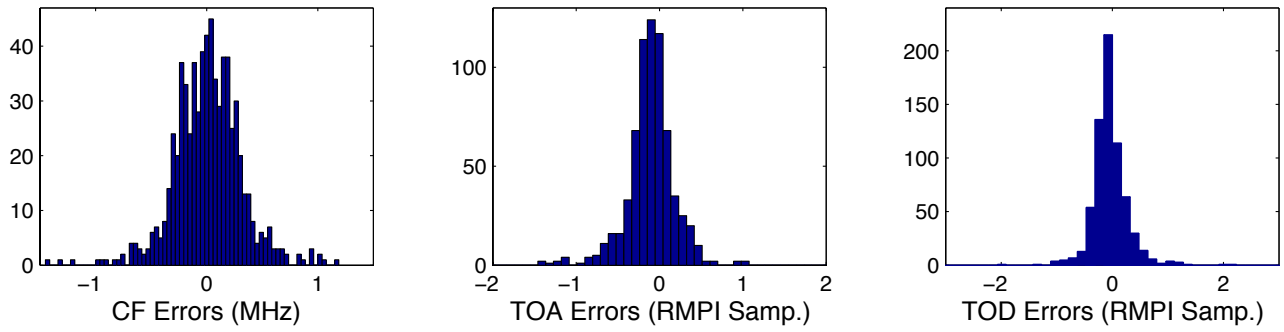
For each pulse, we estimated the CF only from measurements corresponding to times when the signal was active. We then estimated the TOA from RMPI samples corresponding to noise followed by the front end of the pulse. We repeated the procedure for the TOD, using RMPI samples corresponding to the end of the pulse followed by noise only. Fig. 13 shows the distribution of our estimation errors for CF, TOA, and TOD. Additionally, Table I shows statistics on the errors for each of the three parameters.

	CF	TOA		TOD	
	(MHz)	(Frames)	(ns)	(Frames)	(ns)
Max. Err.	1.451	4.176	43.43	24.55	255.32
Min. Err.	7.676e-4	8.149e-5	8.475e-4	3.053e-5	31.75e-4
Std. Dev.	0.305	0.339	3.526	1.540	16.02

TABLE I: Maximum, minimum, and standard deviation of estimation errors for CF, TOA, and TOD over 686 trials (1 Frame $= 1/f_{\text{ADC}} = 10.4$ *ns* is the length of one integration window).

C. Pulse Detection

We tested the pulse detection system by generating 60 test cases containing 12 pulses each (for a total of 720 pulses) with varying amplitudes (ranging over 60 *dB*), phases,



(a) (b) (c)

Fig. 13: Parameter estimation errors for (a) CF, (b) TOA, and (c) TOD over 686 trials.

Input (dBm)	Detection Rate (%)	False Positive Rate (%)	$\sigma_{\text{CF Error}}$ (MHz)	$\sigma_{\text{TOA Error}}$ (Frames)	(ns)	$\sigma_{\text{TOD Error}}$ (Frames)	(ns)
-6	92.08	3.91	0.914	4.266	44.366	4.287	44.585
-9	92.78	2.77	1.281	2.855	29.692	2.917	30.337
-12	92.92	3.74	1.665	3.715	38.636	3.521	36.618
-15	92.50	2.63	0.955	3.411	35.474	3.280	34.112
-18	92.78	1.47	1.665	3.715	38.636	2.521	26.218
-21	92.08	2.93	0.426	2.064	21.466	2.543	26.447

TABLE II: Detection performance as a function of the interferer strength (1 RMPI Samp. $= 1/f_{\text{ADC}} = 10.4$ *ns*).

Pulse Amp. (V)	Detection Rate (%)	$\sigma_{\mathbf{CF}}$ (MHz)	$\sigma_{\text{TOA Error}}$ (Frames)	(ns)	$\sigma_{\text{TOD Error}}$ (Frames)	(ns)
0.016	92.5	2.089	8.330	86.632	8.033	83.543
0.032	95.0	0.753	2.035	21.164	5.871	61.058
0.063	96.7	1.88	2.993	31.127	5.157	53.633
0.126	97.5	1.349	2.997	31.169	2.577	26.801
0.251	97.5	0.839	1.953	20.311	2.536	26.374
0.501	96.7	1.259	1.983	20.623	5.451	56.690
All	96.0	1.452	4.018	41.787	5.280	54.912

TABLE III: Detection rate and standard deviation of the parameter estimate errors as a function of pulse amplitudes (1 Frame $= 1/f_{\text{ADC}} = 10.4$ *ns*).

Pulse Length (Frames)	(ns)	Detection Rate %	$\sigma_{\text{CF Error}}$ (MHz)	$\sigma_{\text{TOA Error}}$ (Frames)	(ns)	$\sigma_{\text{TOD Error}}$ (Frames)	(ns)
19.23	200	89.58	0.713	2.672	27.789	2.775	28.860
48.08	500	98.75	1.045	2.290	23.816	2.413	25.095
96.15	1000	99.58	2.127	5.855	60.892	8.238	85.675
All		96.0	1.452	4.018	41.787	5.280	54.912

TABLE IV: Detection rate and standard deviation of the parameter estimate errors as a function of pulse lengths (1 Frame $= 1/f_{\text{ADC}} = 10.4$ *ns*).

durations (100 *ns*—1 μs), carrier frequencies (100 *MHz*—2.5 *GHz*), and overlaps. All pulse rise times were approximately 10 *ns*. The pulse amplitudes were taken from a set of 6 discrete values, with 120 pulses at each amplitude level, while the durations were taken from a set of 3 discrete values, with 240 pulses at each pulse length. We then ran the detector on each data capture and collected the detection statistics. The detector successfully detected 691 of the 720 pulses for a detection rate of 95.97%, while also allowing 23 false positives for a false positive rate of 3.22%.

Table III shows the detection rate and standard deviation of the parameter estimate errors as a function of the pulse amplitudes. Aside from the lowest-amplitude pulses, the detector's performance is relatively invariant to the pulse amplitude. Table IV similarly shows the detection rate and standard deviation of the parameter estimate errors as a function of the pulse length. As is to be expected, the pulse detection rate improves as the pulse length grows. The TOA and TOD estimates are worse for longer pulses than shorter pulses, but this is to be expected; since the pulses are longer, there are more potential locations to check, and therefore the possibility of error increases. However, it is surprising that the CF estimates get slightly worse as the pulse length increases.

D. Interferer Cancellation

To test the robustness of the detection and estimation system, we repeated our detection experiment and included a constant-frequency interferer at set amplitudes in each experiment. We tested 6 interferer strengths, running 60 experiments with 12 pulses per experiment, for a total of 720 pulses per interferer strength. In each case, all of the pulse amplitudes were the same (to keep the relative interferer strengths well-defined) and the pulses were allowed varying

amounts of overlap. For each experiment, we assumed that we knew the center frequency of the interfering bandwidth, and that the interferer occupied a band with a 25 *MHz* width. Table II shows the detector performance as a function of interferer strength. When the interference is very small in magnitude, the estimation is predictably better. As the interferer grows in strength, the performance only degrades slightly.

VI. Conclusion

We have presented a detailed overview of the design of both hardware and software used in a novel radar-pulse receiver in which information is extracted without performing full signal reconstruction. This novel approach obtains desired information with high accuracy while considerably reducing the back-end computational load. The reduced computational load for parameter extraction potentially expands the applicability of CS-based systems, particularly for real-time processing.

The system was validated using parameter estimates obtained from testing with a large and exhaustive set of realistic radar pulses spanning the parameter space. The *physically* measured results generated from this prototype proof-of-concept system demonstrates the feasibility of the approach. In addition, the data obtained provides ample motivation for further investigation of the merit of CS-based signal acquisition schemes in general.

VII. Acknowledgements

This work was supported by the DARPA/MTO Analog-to-Information Receiver Development Program under AFRL contract FA8650-08-C-7853. In particular, the authors are indebted to Dr. Denis Healy for his foresight, encouragement, and support of this endeavour.

References

[1] B. Murmann, "Trends in low-power, digitally assisted A/D conversion," *IEICE Trans. Electron.*, vol. E93-C, no. 6, pp. 718–727, 2010.

[2] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inform. Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[3] E. J. Candès, "Compressive sampling," *Int. Cong. Math.*, vol. 3, pp. 1433–1452, 2006.

[4] E. J. Candès and M. Wakin, "An introduction to compressive sampling," *IEEE Sig. Proc. Mag.*, vol. 25, no. 2, pp. 21–30, 2008.

[5] E. J. Candès and J. K. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Problems*, vol. 23, no. 3, pp. 969–988, 2007.

[6] D. Donoho, "Compressed sensing," *IEEE Trans. Inform. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[7] M. Wakin, J. Laska, M. Duarte, D. Baron, S. Sarvotham, D. Takhar, K. Kelly, and R. Baraniuk, "An architecture for compressive imaging," in *Proc. Intl. Conf. Image Proc.*, Atlanta, 2006.

[8] M. Lustig, D. L. Donoho, and J. M. Pauly, "Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magn. Reson. Med.*, vol. 58, no. 6, pp. 1182–1195, 2007.

[9] J. Yoo, S. Becker, M. Loh, M. Monge, E. Candès, and A. Emami-Neyestanak, "Design and implementation of a fully integrated compressed-sensing signal acquisition system," in *Proc. if IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing*, 2012.

[10] J. Yoo, S. Becker, M. Loh, M. Monge, E. Candès, and A. Emami-Neyestanak, "A 100MHz-2GHz 12.5× sub-Nyquist rate receiver in 90nm CMOS," in *Proc. IEEE Radio Freq. Integr. Circ. Conf.*, 2012, pp. 31–34.

[11] M. Mishali, Y. C. Eldar, O. Dounaevsky, and E. Shoshan, "Xampling: Analog to digital at sub-Nyquist rates," *IET Cir. Dev. and Systems*, vol. 5, no. 1, pp. 8–20, Jan. 2011.

[12] S. Becker, *Practical Compressed Sensing: modern data acquisition and signal processing*, Ph.D. thesis, California Institute of Technology, 2011.

[13] G. Pope, "Compressive sensing: A summary of reconstruction algorithms," M.S. thesis, ETH, Zürich, Feb. 2009.

[14] J. A. Tropp and S. J. Wright, "Computational methods for sparse solution of linear inverse problems," *Proc. IEEE*, vol. 98, no. 6, pp. 948–958, 2010.

[15] A. Eftekhari, J. Romberg, and M. B. Wakin, "Matched filtering from limited frequency samples," *Arxiv preprint arXiv:1101.2713*, 2011.

[16] M. A. Davenport, P. T. Boufounos, M. B. Wakin, and R. G. Baraniuk, "Signal Processing with Compressive Measurements," *IEEE J. Sel. Top. Signal Proc.*, vol. 4, pp. 445–460, 2010.

[17] J. A. Tropp, J. N. Laska, M. F. Duarte, J. Romberg, and R. G. Baraniuk, "Beyond Nyquist: Efficient sampling of sparse bandlimited signals," *IEEE Trans. Inform. Theory*, vol. 56, no. 1, pp. 520–544, 2010.

[18] E. J. Candès and Y. Plan, "A probabilistic and RIPless theory of compressed sensing," *Preprint: arXiv:1011.3854*, 2010.

[19] S. Kirolos, J. N. Laska, M. Wakin, M. F. Duarte, D. Baron, T. Ragheb, Y. Massoud, and R. G. Baraniuk, "Analog-to-information conversion via random demodulation," in *Proc. IEEE DCAS*, Sept. 2006.

[20] J. N. Laska, S. Kirolos, M. F. Duarte, T. Ragheb, R. G. Baraniuk, and Y. Massoud, "Theory and implementation of an analog-to-information converter using random demodulation," in *Proc. IEEE ISCAS, New Orleans*, May 2007, pp. 1959–1962.

[21] F. Chen, A. Chandrakasan, and V. Stojanovic, "A signal-agnostic compressed sensing acquisition system for wireless and implantable sensors," in *Proc. IEEE Cust. Integr. Circ. Conf.*, 2010.

[22] S. Becker, E. Candès, D. Ching, A. Emami-Neyestanak, M. Grant, E. Nakamura, J. Romberg, E. Sovero, M. Wakin, and J. Yoo, "A non-uniform sampler for wideband spectrally-sparse environments," *submitted to IEEE J. on Emerging and Sel. Topics in Cir. and Sys.*, 2011.

[23] J. Romberg, "Compressive sensing by random convolution," *SIAM J. Imag. Sci.*, vol. 2, no. 4, pp. 1098–1128, 2009.

[24] Y. C. Eldar, "Compressed sensing of analog signals in shift-invariant spaces," *IEEE Trans. Sig. Proc.*, vol. 57, no. 8, pp. 2986–2997, Aug. 2009.

[25] J. P. Slavinsky, J. N. Laska, M. A. Davenport, and R. G. Baraniuk, "The compressive multiplexer for multi-channel compressive sensing," in *IEEE Intl. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, Prague, May 2011, pp. 3980–3983.

[26] B. Chan, B. Oyama, C. Monier, and A. Gutierrez-Aitken, "An ultra-wideband 7-bit 5-GSPS ADC implemented in submicron InP HBT technology," *IEEE J. of Solid-State Circ.*, vol. 43, no. 10, pp. 2187–2193, 2008.

[27] A. Mirzaei, S. Cherazi, and R. Bacheri, "Analysis of first-order anti-aliasing integration sampler," *IEEE Trans. of First-Order Anti-Aliasing Integration Sampler*, vol. 55, no. 10, pp. 2994–3005, 2008.

[28] R. Gold, "Optimal binary sequences for spread spectrum multiplexing," *IEEE Trans. on Inform. Theory*, vol. 13, no. 4, pp. 619–621, 1967.

[29] P. Vorenkamp and J. Verdaasdonk, "Fully bipolar, 120-Msample/s 10-b track-and-hold circuit," *IEEE J. of Solid-State Circ.*, vol. 27, no. 7, pp. 988–992, 1992.

[30] R. L. Pickholtz, D. L. Schilling, and L B. Milstein, "Theory of spread-spectrum communications—a tutorial," *IEEE Trans. on Comm.*, vol. 30, no. 5, May 1982.

[31] M. A. Davenport and M. B. Wakin, "Compressive sensing of analog signals using discrete prolate spheroidal sequences," *Arxiv preprint arXiv:1109.3649*, 2011.

[32] D. Slepian, "Prolate spheroidal wave functions, fourier analysis, and uncertainty. v - the discrete case," *Bell Sys. Technical J.*, vol. 57, pp. 1371–1430, 1978.