# A Pragmatic Look at Some Compressive Sensing Architectures With Saturation and Quantization

Javier Haboba, *Student Member, IEEE*, Mauro Mangia, *Student Member, IEEE*, Fabio Pareschi, *Member, IEEE*, Riccardo Rovatti, *Fellow, IEEE*, and Gianluca Setti, *Fellow, IEEE*

***Abstract*—The paper aims to highlight relative strengths and weaknesses of some of the recently proposed architectures for hardware implementation of analog-to-information converters based on Compressive Sensing. To do so, the most common architectures are analyzed when saturation of some building blocks is taken into account, and when measurements are subject to quantization to produce a digital stream. Furthermore, the signal reconstruction is performed by established and novel algorithms (one based on linear programming and the other based on iterative guessing of the support of the target signal), as well as their specialization to the particular architecture producing the measurements. Performance is assessed both as the probability of correct support reconstruction and as the final reconstruction error.**

**Our results help highlighting pros and cons of various architectures and giving quantitative answers to some typical design-oriented questions. Among these, we show: 1) that the (Random Modulation Pre-Integration) RMPI architecture and its recently proposed adjustments are probably the most versatile approach though not always the most economic to implement; 2) that when 1-bit quantization is sought, dynamically mixing quantization and integration in a randomized $\Delta\Sigma$ architecture help bringing the performance much closer to that of multi-bit approaches; 3) for each architecture, the trade-off between number of measurements and number of bits per measurements (given a fixed bit-budget); and 4) pros and cons of the use of Gaussian *versus* binary random variables for signal acquisition.**

***Index Terms*—Analog-to-information converter (AIC), compressive sensing (CS), 1-bit compressive sensing.**

## I. Introduction

THIS PAPER aims at developing *design guidelines* to reduce the gap between theoretical development of Compressive Sensing (CS) [1], [2] for analog-to-information converters (AICs) [3] and their hardware implementation.

More precisely, following [1], we will assume that we wish to acquire analog signals possessing a property called *sparsity*, which amounts to know that a basis exists with respect to which the representation of a signal has a small (w.r.t. full dimensionality of the basis itself) number of non-null coefficients. These represent the *information content* of the signal, so that sparsity implies that the latter is limited and, for example, substantially smaller than the waveforms bandwidth. Furthermore, the above assumption paves the way for developing AICs [3], i.e., more efficient circuits/systems for translating the analog signals into a certain number of bits that are sufficient to reproduce their information content and which do not follow the classical analog-to-digital conversion framework based on Shannon–Nyquist theory. In other words, AICs must be able to identify the above coefficients relying on a number of samples (more generally, "measurements") which is lower than what the Nyquist criterion would dictate, so that the number of bits needed to encode the original signal may be less than what would be entailed by a straightforward pulse-code-modulated (PCM) stream with a number of bit-per-sample sufficient to match the same accuracy requirements.

The fact that the number of bits used is less than what a conventional ADC would produce to obtain the same quality by means of a PCM justifies the name "compressive" and is also the key feature that makes AICs a promising technique in all the cases in which the resources allocated to acquisition and/or transmission of the acquired data are limited.

Starting from the first example in [3], several architectures of AICs have been proposed in the literature [4]–[6], all of which enjoy the same theoretical ground, i.e., the possibility of performing a sparsity-informed conversion in digital form of the information carried by the input signal. Needless to say, any real implementation of any of the above architectures would violate at least some of those assumptions since, first of all, the unavoidable saturations and quantizations in real circuits prevent a truly linear processing of the signal. It is also fundamental to stress that the effect on AIC performance of the above implementation non-idealities can be properly defined only if one takes into account the result of the corresponding "decoding"/reconstruction operation, i.e., the one which that takes the bit stream at the AIC output and is able to compute the non-null coefficients in the basis expansion. More specifically, differently with respect to classical ADCs where this operation is performed by low-pass filtering[1], one needs to take into account that when "encoding" is performed via an AIC, reconstruction must be obtained by inverting an underconstrained linear system of equations when the corresponding matrix obeys some assumptions [8], [9] and by adopting specifically designed algorithms [10], [11].

With respect to this issue, each proposed architecture has its own approach, possibly even emphasizing saturation and quantization by using single-bit ADC stages [12], [13] whose result

Manuscript received May 18, 2012; revised August 29, 2012; accepted September 13, 2012. Date of publication November 16, 2012; date of current version December 05, 2012. This paper was recommended by Guest Editor D. Allstot.

J. Haboba and M. Mangia are with ARCES—University of Bologna, Bologna, Italy (e-mail: jhaboba@arces.unibo.it; mmangia@arces.unibo.it).

R. Rovatti is with DEI-University of Bologna, Bologna, Italy, and also with ARCES—University of Bologna, Bologna, Italy (e-mail: riccardo.rovatti@unibo.it).

F. Pareschi and G. Setti are with the ENDIF—University of Ferrara, Ferrara, Italy, and also with ARCES—University of Bologna, Bologna, Italy (e-mail: fabio.pareschi@unife.it, gianluca.setti@unife.it).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JETCAS.2012.2220392

[1]Plus, possibly, decimation filtering, in case of oversampled ADCs.

is then properly processed to recover linearity either by dynamic processing and/or algorithmic intelligence.

All these *ad hoc* solutions are difficult to analyze from a fully theoretical point of view though they are demonstrated to work at least in some cases by the researchers proposing and testing them.

What we do here (even if we are also describing a novel algorithm for the decoding stage of a recently proposed 1-bit architecture) is a survey of some of the most common architectures *in order to establish a common framework in which a fair comparison can be made by simulating them in various operating conditions, thus establishing pros and cons for all of them*,

The paper is organized as follows. Section II presents the general linear measurement model and some of the theoretical results guaranteeing that it may be used to perform AIC. Section III describes two types of algorithms able to reconstruct the information content of a signal from the bits produced by an AIC, along with some possible adjustments improving their performance in some situations. A representative is taken from each class to be used in simulations. Section IV devotes a subsection to the description of each of the three classes of architectures we analyze: 1) random modulation pre-integration; 2) random sampling; and 3) 1-bit compressive sensing. To evaluate and compare the above algorithms and architectures in an homogeneous environment, Section V details the operative conditions under which we perform simulations and defines the performance figures we consider. Finally, in Section VI we run simulations to investigate some implementation-related aspects, such as relative performance of reconstruction algorithms, the possibility of using antipodal random modulation instead of a Gaussian one, saturation and quantization in RMPI systems, the optimal partition of a bit budget into a certain number of measurements encoded into digital words, the possibility of reducing the gap between the performance of many-bit AIC and that of 1-bit AIC, etc.

## II. General Model

For an $n$-dimensional vector $a = (a_0, \ldots, a_{n-1})^\top$ we define the support of $a$ as

$$\mathrm{supp}(a) = \{j = 0, \ldots, n-1 | a_j \neq 0\}$$

its sparsity $\mathrm{spar}(a)$ (sometimes indicated as $L_0$ norm) as the cardinality of $\mathrm{supp}(a)$ and its usual $p$-norm as

$$\|a\|_p = \left( \sum_{j=0}^{n-1} |a_j|^p \right)^{1/p}$$

Sampled in time, the signal $x(t)$ that we want to acquire is represented by an $N$-dimensional vector of real numbers $x$. Following [1] we will assume that a suitable basis exists whose vectors are the columns of the $N \times N$ matrix $\Psi$, and that the signal of interest are $K$-sparse, which means that for any instance of $x$ there is an $N$-dimensional vector $\alpha$ such that $x = \Psi\alpha$ and $\mathrm{spar}(\alpha) \leq K$.

Starting from $x$, the acquisition system produces $M$ measurements by means of a linear projection, i.e., by correlating the elements of $x$ with some coefficients. By arranging those coefficients as the rows of the $M \times N$ matrix $\Phi$ we obtain an $M$-dimensional measurement vector $y = \Phi x$. Since the input signal will be corrupted by noise, what we will acquire is the signal $\tilde{x} = x + \nu$, so that the measurement vector becomes $y = \Phi\tilde{x} = \Phi(\Psi\alpha + \nu)$, where $\nu$ is the vector of noise coefficients superimposed to the samples in $x$, whose magnitude relative to the signal $x$ can be accounted for by defining an intrinsic signal-to-noise ratio (ISNR) $\|x\|_2/\|\nu\|_2$.

Since the sparsity basis is a feature of the signal to acquire, the matrix $\Psi$ is known, so that different AICs architectures can be distinguished based on how they build and apply the matrix $\Phi$.

The aim of a proper design of $\Phi$ is to guarantee that the original signal $x = \Psi\alpha$ can be reconstructed starting from the measurements, i.e., that setting $\Theta = \Phi\Psi$ and $\xi = \Phi\nu$, the equation:

$$y = \Theta\alpha + \xi \tag{1}$$

can be "solved" for $\alpha$ once that $y$ is known.

In addition to the fact that the matrix $\Theta$ is $M \times N$ with $M \ll N$ as it is assumed in any purely theoretical framework [8]–[10], finding a solution of (1) can in practice additionally complicated by the fact that: 1) $\nu$ is unknown; 2) that $y$ is usually available after its analog-to-digital conversion and is, therefore, subject to quantization errors; and 3) that the multiply-and-accumulate operations needed to compute correlations may suffer from limited signal range and thus be corrupted by saturation.

To counter all these problems one classically relies on the *a priori* knowledge that $\mathrm{spar}(\alpha) \leq K$ [8], though other priors can be added and exploited to increase reconstruction performance, as recently suggested in [7] and [16]. This is possible since priors help defining a subset of $\mathbb{R}^N$ containing all instances of $x$, so that the acquisition mechanism should map this subset into the measurement space $\mathbb{R}^M$ "quasi-bijectively" in a sense that will be clarified in the following.

More precisely [8], when sparsity is one of the priors, if $\Theta$ can be thought of as a realization of a random matrix with independent entries drawn according to a variety of distributions, then mapping by means of $\Theta$ provides, with high probability, the needed "quasi-bijection." Formally speaking, we say that a matrix $\Theta$ is a *Restricted Isometry* (RI) [8] when there is a constant $0 \leq \delta_K < 1$ such that

$$(1 - \delta_K)\|\alpha\|_2^2 \leq \|\Theta\alpha\|_2^2 \leq (1 + \delta_K)\|\alpha\|_2^2 \tag{2}$$

whenever $\mathrm{spar}(\alpha) \leq K$. Hence, even if the dimensionality $M$ of the co-domain of a restricted isometry is less than the dimensionality $N$ of its domain, the mapping of $K$-sparse vectors leaves lengths substantially unaltered.

If $\Theta$ is made of independent random entries characterized by a sub-Gaussian distribution [14], [15], then with an overwhelming probability, it is a restricted isometry with a constant $\delta_K$ that decreases as $M$ increases [9].

Intuitively, if $\Theta$ is a restricted isometry, once that $\mathrm{supp}(\alpha)$ is known, we may restrict $\Theta$ to that domain and obtain an injective mapping that can be reversed to yield the whole $\alpha$.

This is why an almost constant ingredient in the recipes for all compressive sensing architectures is *randomness as a mean of capturing information that is known to be sparse*. What is usually done is to overlook the fact that theory puts conditions

on the statistical structure of $\Theta$ and design a system in which $\Phi$ is random and (hopefully) transfers its beneficial properties to $\Theta = \Phi\Psi$.

An important side-effect of this assumption (widely verified in practice) is that one does not design the acquisition matrix $\Phi$ depending on the specific $\Psi$ but relies on randomness to implicitly "scan" all possible sparsity bases.

Yet, tuning is still possible since the many degrees of freedom that are available when designing a stochastic process may be used not only to ensure restricted isometry but to cope and exploit other priors (such as *rakeness* [7], [16]) that may be available for the original signal.

## III. Reconstruction Algorithms

Once that an "encoding" mapping allowing reconstruction has been devised, its "inversion" must be obtained by algorithmic means every time a measurement vector $y$ comes in.

Though reconstruction mechanisms should be designed jointly with the architectures producing the measurements, they are classically addressed as separate components of the overall acquisition system. Their development and analysis is a flourishing field that has recently produced strong and general results and taxonomies [10].

We will here take a very pragmatic approach, concentrate on the most frequently adopted methods, and note that those techniques fall in one of two categories: *optimization-based reconstruction* and *iterative support-guessing reconstruction*.

Both types of technique are commonly devised and set up in the noiseless and idealized case (i.e., for $\xi = \Phi\nu = 0$) and with neither quantization nor saturation, and are proved (or simply seen) to work in more realistic settings.

The key fact behind optimization-based methods is that, among all the possible counterimages $\alpha$ of the vector $y = \Theta\alpha$ the one that we are looking for is the "most sparse", i.e., the one for which $\mathrm{spar}(\alpha)$ is minimum.

Since we usually have $\mathrm{spar}(\alpha) \leq K \ll N$ this assumption is sensible. Moreover, it leads to some elegant results on the possibility of recovering $\alpha$ by means of simple optimization problems [8].

More formally, it can be shown if $\Theta$ is a restricted isometry with constant $\delta_K \leq \sqrt{2} - 1$, then, for any $\epsilon > 0$, the $\hat{\alpha}$ solution of the optimization problem

$$\begin{aligned} &\min \|\hat{\alpha}\|_1 \\ &\text{s.t. } \|\Theta\hat{\alpha} - y\|_2 \leq \epsilon \end{aligned} \tag{3}$$

is such that

$$\|\hat{\alpha} - \alpha\|_2 \leq C\epsilon$$

for some constant $C > 0$.

Hence, if we use $\epsilon$ to bound the maximum magnitude of the disturbances involved in the measurement process (for instance, by setting it proportional to the variance of the noise plus that of the quantization error) we can guarantee that the reconstruction error vanishes when disturbances go to zero.

Though not impossible, the straightforward application of the above result, depends on a reliable estimation of the parameter $\epsilon$ that quantifies the maximum foreseeable deviation between the unperturbed measurement and its actual value in presence of a mixture of known (e.g., quantization) and unknown (e.g., noise) disturbances.

It is therefore quite common to substitute $\|\Theta\hat{\alpha} - y\|_2 \leq \epsilon$ with $\Theta\hat{\alpha} = y$ by implicitly assuming that the system is working in a relative low-disturbance regime that allows to assume $\epsilon \simeq 0$. Within this approximation, it is convenient to re-express the resulting optimization problem within the framework of linear programming by defining $u = (1, \ldots, 1)^\top$ and by introducing the auxiliary unknown vector $\ell = (\ell_0, \ldots, \ell_{n-1})^\top$ to write

$$\begin{aligned} &\min u^\top \ell \\ &\quad \Theta\hat{\alpha} = y \\ &\text{s.t. } \ell \geq 0 \\ &\quad -\ell \leq \hat{\alpha} \leq \ell \end{aligned} \tag{4}$$

where vector inequalities are meant to hold component-wise.

The equality constraints in (4) can be adjusted to cope with specific features of a given architecture or to take into account quantization or saturation.

In particular, due to quantization, we know that the true value of the $j$th component of $y$ is somewhere in the interval $[y_j - \Delta y_j/2, y_j + \Delta y_j/2]$ with $y_j$ being the value known to the algorithm and $\Delta y_j$ the corresponding quantization step. Hence, in presence of a coarse quantization, it is sensible to substitute the equality constraints $\Theta\hat{\alpha} = y$ in (4) with $y - \Delta y/2 \leq \Theta\hat{\alpha} \leq y + \Delta y/2$, where $\Delta y = (\Delta y_0, \ldots, \Delta y_{m-1})^\top$. Though it surely models the acquisition procedure with greater accuracy, this adjustment does not necessarily lead to improvements and is commonly employed only when one may expect the various $\Delta y_j$ to be substantially different one from the other.

We will indicate the algorithm we use as representative of optimization-based reconstruction as $\texttt{MinL}_1$.

It is interesting to note that $\texttt{MinL}_1$ works without any knowledge of the exact value of $K$ further to that implicit in the number of measurements that must be enough to allow reconstruction. This may be a plus in situations where $K$ cannot be exactly determined in advance. Regrettably, this positive feature is balanced by the fact that, in general, linear programming solution is computationally more expensive that other kinds of iterative reconstruction.

As far an iterative support-guessing reconstruction is concerned, note that, if $\mathrm{supp}(\alpha)$ were known we could drop the columns in $\Theta$ that are surely multiplied by 0 and the corresponding entries in $\alpha$ to obtain an $M \times K$ matrix $\Theta_{\mathrm{supp}(\alpha)}$ and a $K$-dimensional vector $\alpha_{\mathrm{supp}(\alpha)}$ for which $y = \Theta_{\mathrm{supp}(\alpha)}\alpha_{\mathrm{supp}(\alpha)}$. Since $M > K$, this is an overconstrained problem that may be effectively (even "optimally" in case of Gaussian disturbances [17] inverted by using the Moore-Penrose pseudo-inverse $\Theta^\dagger_{\mathrm{supp}(\alpha)}$ and computing $\alpha_{\mathrm{supp}(\alpha)} = \Theta^\dagger_{\mathrm{supp}(\alpha)} y$.

Iterative support-guessing methods are, in general, procedures that alternate a rough, non-necessarily sparse, solution of $y = \Theta\alpha$ from which an estimate of $\mathrm{supp}(\alpha)$ is inferred (for example by thresholding on the magnitudes of the components of the temporary solution) that is then exploited in a pseudo-inverse-based step refining the value.

Though more sophisticated alternatives exists, a reference algorithm within this class is $\texttt{CoSaMP}$ [11] that has some definite advantages. First, it works for matrices $\Theta$ that are restricted

isometries and, if $K$ is known and the isometry constant $\delta_{2K}$ for vectors with $2K$ non-zero components can be bounded by $\delta_{2K} \leq c \leq 0.1$, then, given a tolerance $\epsilon > 0$, the reconstructed vector $\hat{\alpha}$ satisfies

$$\|\hat{\alpha} - \alpha\|_2 \leq C \max\left\{\epsilon, \frac{\|\alpha'\|_1}{\sqrt{K}} + \|\Phi\nu\|_2\right\}$$

where $\alpha'$ is the vector that can be obtained by $\alpha$ by setting to zero its $K/2$ largest entries.

The resulting algorithm is provably fast and, beyond the above formal guarantee on its performance, it is usually extremely stable and effective in recovering the original signal. These favorable properties are paid with the additional assumption that the isometry constant $\delta_{2K}$ must be quite low and that, in the most common implementations, the sparsity of $\alpha$ must be known.

In analogy to what happens for optimization-based reconstruction, `CoSaMP` can be tailored to specific architectures. This can be done, for example, if it is known that errors in the magnitudes of the entries of $\alpha$ are correlated by an implicit filtering in the acquisition scheme. Such an effect can be exploited by inserting a filtering step when passing from support-guessing to pseudo-inversion. More detail on how this may affect performance reconstruction will be given in the following.

## IV. Compressive Sensing Architectures

From the two previous sections, we get that to define a compressive sensing system we need to describe two stages:

- **encoder**: a hardware system (actually implementing the AIC) and performing some mixed analog-digital operations on the incoming signal to produce a stream of bits. The mixed analog-digital operations are modeled as an instance $\Phi$ of a random matrix linking the signal samples to the measurements whose quantization yields the stream of bits transferred from the encoder to the decoder;
- **decoder**: an algorithm that takes the incoming bits and, based on the knowledge of $\Phi$, reconstructs the original signal.

Note that, in a practical implementation, one may not want to communicate $\Phi$ to the decoder and thus most often exploits pseudorandom generators with a common initialization to yield matrices that can be simultaneously known at both stages.

Saturation and quantization are unavoidable in the signal path since the communication between encoder and decoder happens along a digital channel thus implying an ADC block with a finite range (we will assume $[-V^{\max}, V^{\max}]$ for a certain $V^{\max}$) and a finite number of levels.

In the following we will consider the number $B$ of bits generated by the encoder corresponding to the acquisition of the input signal over a given time interval. This can be considered as a "bit budget" since it may be partitioned into digital words of different depths corresponding to different measurements.

### A. Random Modulation Pre-Integration—RMPI

This is the first and probably the most straightforward implementation of an AIC [3].

With reference to Fig. 1, the samples of the incoming signal $x_k$ are multiplied by the quantities $\Phi_{j,k}$ for a given $j$ and then fed into an accumulation stage to yields the value of the $j$-th

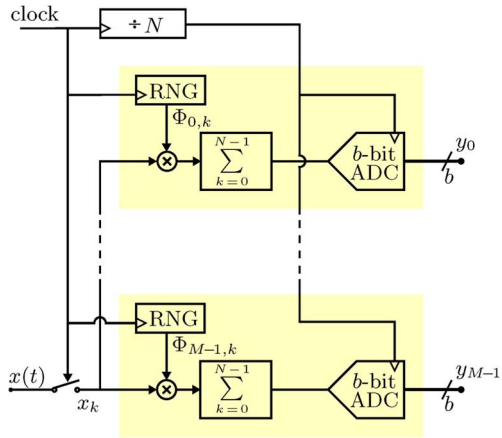



Fig. 1. Block scheme of an RMPI encoder.

measurement $y_j$ that is then quantized by an $b$-bit ADC and aggregated with all the other quantized measurements into the stream of bits that is passed to the decoding stage.

The implementation of the analog blocks preceding the ADC offers several options.

The structure of the multiplier depends on the quantities $\Phi_{j,k}$: some classical approaches adopt Gaussian random variables (Gaussian RMPI) and force the deployment of complete four-quadrant analog multipliers, while more aggressive approaches suggest to constrain $\Phi_{j,k} \in \{-1, +1\}$ (antipodal RMPI) so that multiplication can be implemented by simple switching.

The accumulation stage may be implemented either as a continuous time integrator or as a switched capacitor subcircuit that implicitly matches the discrete-time operation of the multiplier. In any case, the output of the accumulating device will be subject to saturation. Referring to a discrete-time implementation, where $y_j = \sum_{k=0}^{N-1} \Phi_{j,k} x_k$, and relying on the following assumptions: `i`) the elements of $x$ and $\Phi$ are independent random variables; `ii`) the elements $\Phi_{j,k}$ of $\Phi$ are *independent* and *identically distributed* (either Gaussian or binary antipodal) random variables, with zero mean and unity variance, i.e., $\mathrm{E}[\Phi_{j,k}] = 0$ and $\mathrm{E}[\Phi_{j,k}^2] = 1$; `iii`) the energy of $x$ in the accumulation time window is unitary, i.e., assuming that the variance of each $x_k$ is the same, $\sum_{k=0}^{N-1} \mathrm{E}[x_k^2] = 1$. Let us consider the random variable $\xi_{j,k} = \sqrt{N}\Phi_{j,k}x_k$, so that $y_j = 1/\sqrt{N}\sum_{k=0}^{N-1} \xi_{j,k}$. Since the random variables $\xi_{j,k}$ are independent with zero-mean, thanks to `i`) and `ii`), and have unity variance, due to `i`)–`iii`), for $y_j$ the hypotheses for applying the central limit theorem are satisfied, so that for large $N$, it converges to the standard normal distributed random variable. Note also that this results is signal independent, i.e., it holds no matter the mean value and the variance of the $x_k$.

Furthermore, the speed of convergence is large enough to guarantee that the practical approximation of $y_j$ with a normally distributed random variable is satisfied for reasonable values of $N$. As an example, Fig. 2(a) reports the observed distribution of the $y_j$ for the antipodal RMPI with $N = 256$ and where the input signals are a 4-sparse and a 16-sparse signal with respect to the Fourier basis, as well as a synthetic ECG signal, generated as in [18]. All signals have been imposed to satisfy the condition $\mathrm{E}[x_k^2] = 1/N$, so that, according to our model, $\mathrm{E}[y_j^2] = 1$.

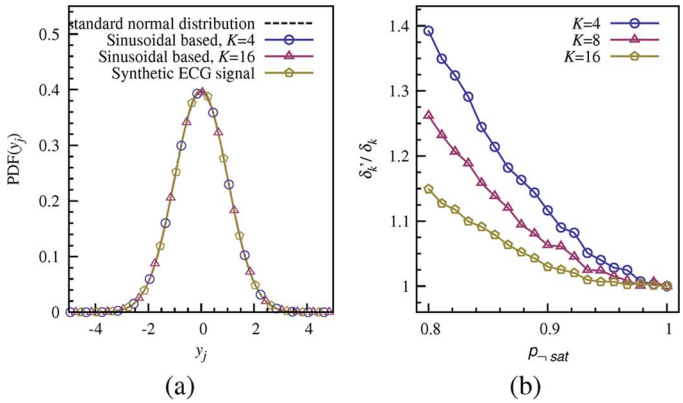



Fig. 2. (a) PDF of the value of $y_j$ for different input signals in the antipodal RMPI case, obtained with 1000 trials of a system with $M = 96$. (b) Average variation of restricted isometry constant $\delta'_K$ for the sampling matrix $\Phi'$ with respect to the reference value $\delta_K$ given by $\Phi$ in an antipodal RMPI system with $N = 256, M = 96$ and $K \in \{4, 8, 16\}$.

In all cases, the observed distributions are superimposing with the standard normal one.

Of course, since a normal distribution is not limited, wherever the input range of the ADC is set, there is an unavoidable nonzero probability that $y_j$ falls out of the ADC conversion range. We refer to this as *static saturation*. Finally, it is easy to extend the above mathematical model to the generic time step $k$ (instead of considering only the final one at $k = N - 1$), where *dynamic saturation* occurs whenever the intermediate accumulated value is out of the accumulator linear region.

Some theoretical considerations on the effect of static saturation can be found in [13] while a discussion of both static and dynamic saturation in the more realistic model we adopt here has been first proposed in [19]. From the latter, in particular, it is sensible to design the accumulation stage so that its saturations coincide with the extrema of the input range of the ADC that we identify with the interval $[-V^{\max}, V^{\max}]$.

To model the system behavior, we say that, while in the process of computing the $j$th measurement, the circuit produces the $k$th intermediate result $y_{j,k} = \sum_{l=0}^{k} \Phi_{j,l} x_l$ when accumulating the $k$th sample, to finally produce $y_j = y_{j,N-1}$.

Yet, depending on $\Phi$ and on the signal samples, there may be time instants $k$ in which saturation takes place and causes $y_{j,k} \neq y_{j,k-1} + \Phi_{j,\kappa_j} x_k$. If $\kappa_j$ is the first of these instants (i.e., we know that saturation did not affect the system before $\kappa_j$ for the $j$th measurement) we have $y_{j,\kappa_j-1} = \sum_{k=0}^{\kappa_j-1} \Phi_{j,k} x_k$ and we can also approximate $y_{j,\kappa_j} \simeq \sum_{k=0}^{\kappa_j} \Phi_{j,k} x_k \simeq \pm V^{\max}$, and certainly $y_j = y_{j,N-1} \neq \sum_{k=0}^{N-1} \Phi_{j,k} x_k$. In other words, if saturation takes place, the corresponding measurement is corrupted and, in general, brings little information on the original signal.

As a very rough but effective estimation strategy, we may think that the theoretical value of $y_j = \sum_{k=0}^{N-1} \Phi_{j,k} x_k$ is distributed as a Gaussian random variable with a variance that increases with $N$ and depends on the statistical properties of $\Phi$ and of the signal samples. It is then possible to, at least approximately, know the probability $p_{\neg\mathrm{sat}}$ with which such a theoretical value does not exceed the saturation limits.

Even though saturation may first happen at some $\kappa < N - 1$, it is sensible to take $p_{\neg\mathrm{sat}}$ as an indicator of the relative match between the signal actual range and the system saturation levels.

An important design parameter for this particular AIC is the probability $p_{\neg\mathrm{sat}}$ of no static saturation when dynamic saturation did also previously not occur. From the above consideration, it readily follows that since $y_j$ is normally distributed, then $p_{\neg\mathrm{sat}} = 1 - \mathrm{erfc}(V^{\max})$, where $\mathrm{erfc}(\cdot)$ is the complementary error function. Furthermore, despite $p_{\neg\mathrm{sat}}$ is formally defined in absence of dynamic saturation, as it will be confirmed in Section VI, it can be considered as a sensible indicator of the relative match between the signal actual range and the system saturation levels, even though dynamic saturation may first happen at some $\kappa < N - 1$.

Given the above definitions, the first straightforward design consideration is that, since $N$ is typically in the order of tens if not hundreds, no design strategy may actually aim at obtaining $p_{\neg\mathrm{sat}} \simeq 1$ since this would require to increase $V^{\max}$ to satisfy $-V^{\max} \leq \sum_{k=0}^{N-1} \Phi_{j,k} x_k \leq V^{\max}$ for any $N$ and $x$. To cope with this, a fairly simple approach would be what is referred to in [13] as *"democracy"* of the set of measurements, i.e., the fact that, under mild conditions, one may assume that the information content of each measurement is identical. If this were true, simply discarding saturated measurement would produce a graceful performance degradation since the acquisition system would behave as if it were designed to use a number of measurement equal to the number of the non-saturated ones. Moreover, non-degraded performance could be restored by simply taking further measurement until the original number is reached. Following [19], we will name this approach as RMPI–SPD since it concretizes in saturated projection dropping.

Regrettably, perfect democracy only holds between measurements that are taken as linear combinations of the samples. Saturation acts as a selector discarding those that have a larger value while keeping the smaller ones. From the point of view of the SNR this is clearly not a democratic behavior and causes non-saturated measurements to be less useful than those that have to be dropped and cannot be perfectly replaced by simply trying more measurements.

To cope with this, one may alternatively think [19] of exploiting all the information that is still available as a linear combination of the samples $x_k$, i.e., the fact that, if saturation first occurs at $\kappa_j$, we have $y_{j,\kappa_j} \simeq \sum_{k=0}^{\kappa_j} \Phi_{j,k} x_k \simeq \pm V^{\max}$. This allows to replace $\Phi$ with a modified matrix $\Phi'$ in which, the rows corresponding to saturated measurements have zeros in correspondence of time instants after the one in which saturation occurs, i.e., more formally we may set

$$\Phi'_{j,k} = \begin{cases} \Phi_{j,k} & \text{for } k = 0, \ldots, \kappa_j - 1 \\ 0 & \text{for } k = \kappa_j, \ldots, N - 1 \end{cases} \tag{5}$$

where $\kappa_j = N$ correspond of the case of no dynamic or static saturation, and

$$y'_j = \begin{cases} y_j, & \text{if no saturation occurred} \\ V^{\max}, & \text{if positive saturation occurred} \\ -V^{\max}, & \text{if negative saturation occurred} \end{cases} \tag{6}$$

so that the AIC output can be expressed as $y' = \Phi' \Psi \alpha = \Theta' \alpha$.

Note that this solution makes the matrix $\Phi'$ used for reconstruction a function of the signal samples $x_k$ that caused saturation. Hence, the measurement vector passed to the decoder must contain also the information needed to construct $\Phi'$ from the signal-independent $\Phi$. Assuming that $b$ bits are assigned to

the $j$th component of the measurement vector, we decide to reserve 1 bit to indicate whether saturation happened and use the remaining bits depending on its value: either $b-1$ bits for the quantized encoding of a non-saturated measurement, or 1 bit to discriminate between positive and negative saturation along with $b-2$ bits dedicated to the encoding of $\kappa_j$. We will indicate this approach as RMPI–SPW, standing for saturated projection windowing.

Even if we cannot provide exact results on the RIP for $\Phi'$, it is worth taking a pragmatic look at this issue by comparing the RI constant $\delta'_K$ associated to the $\Phi'$, with the RI constant $\delta_K$ associated to the original matrix $\Phi$ by means of extensive Monte Carlo simulations. To do so, we use the approach described in [7] which involves to compute the eigenvalues of the matrices $\Psi^*\Phi^*\Phi\Psi$ and $\Psi^*\Phi'^*\Phi'\Psi$, where $\cdot^*$ stand for transpose conjugate.

Fig. 2(b) reports the average variation over 500 trials of the ratio $\delta'_K/\delta_K$ as a function of $p_{\neg\mathrm{sat}}$ varying from 1 to a value as low as 0.8, when a Fourier basis matrix has been chosen for $\Psi$ and where $N=256, M=96$ and $K\in\{4,8,16\}$, so that the associated $\Phi$ is known to be a RI matrix[2]. As it can be noticed, when $p_{\neg\mathrm{sat}}\in[0.8,1]$, so that the number of entries in the matrix $\Phi'$ which are changed to zero is not too large, $\delta'_K$ smoothly increases with respect to $\delta_K$ in all considered examples. This can be intuitively accepted as a good indication that one can reasonably expect that $\Phi'$ satisfies condition (2) when $p_{\neg\mathrm{sat}}$ is not far from 1.

We would like to propose here a further observation. It is know that RIP property is only a sufficient condition to CS, but it is not necessary [1]. Once the projection matrix $\Phi'$ is obtained at the decoder, signal reconstruction happens by usual means though guarantees on the resulting $\delta'_K$ are not as strict as for the original $\Phi$.

In fact, the trailing zeros that may appear at certain rows of $\Phi'$ potentially degrade its ability to behave almost as an isometry for $K$-sparse vectors. Yet, as a partial relief from this concern, we know that $\Phi'$ is signal-dependent and we get from (5) and from the definition of $\kappa_j$ that when a row of $\Phi'$ contains trailing 0s it is because the projection of the signal along the initial part of the same row was large enough to cause saturation.

Hence it is sensible to assume that, with high probability, most of the energy of the signal had already been extracted by the first part of the row that still enters the reconstruction procedure.

### B. Random Sampling—RSAM

In classical acquisition systems, samples of the signal are taken regularly on the time axis at a given rate (usually not less than the Nyquist one). AICs relying on random sampling avoid this regularity to produce a number of measurements that, on the average, are less than those produced by Nyquist sampling, while still allowing the reconstruction of the whole signal thanks to sparsity and other priors.

In principle, sampling instants can happen anywhere along the time axis. Yet, a straightforward implementation chooses them among regularly spaced time points that can be selected by digital means. The result is the schema shown in Fig. 3

[2]The value $M=96$, for all considered $K$ values, obeys the theoretical inequality imposed by RI, i.e., $M>CK\log(N/K)$ with $C=4$ [8]

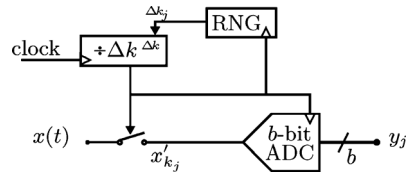



Fig. 3. Block scheme of an RSAM encoder.

where a backward counter is pseudorandomly reloaded each time it reaches zeros, triggering conversion. Grid spacing, and thus clock rate, depends on the resolution with which one wants to place the sampling instants and thus may be expected to be larger that Nyquist rate.

To translate the above block scheme into formulas, say that the clock identifies a vector $x'=(x'_0,\dots,x'_{\upsilon-1},\dots,x'_{\upsilon(N-1)},\dots,x'_{\upsilon N-1})^\top$ containing samples of $x(t)$ at a rate equal to $\upsilon N$. Since $x(t)$ is bandlimited it can be reconstructed from its $N$ Nyquist samples in $x=(x_0\dots,x_{N-1})^\top$, and the two vectors $x'$ and $x$ are linked by $x'=\Upsilon_{\mathrm{RSAM}}x$, being $\Upsilon_{\mathrm{RSAM}}$ an upsampling matrix.

With this, the $M\times N$ matrix $\Phi$ is nothing but the product $\Phi=\Sigma\Upsilon_{\mathrm{RSAM}}$, where $\Sigma$ is the random sampling matrix defined by the $M$ time instants $k_0<k_1<\cdots<k_{M-1}$ at which the counter reaches 0 as in

$$\Sigma_{j,k}=\begin{cases}1, & \text{if } k=k_j\\ 0, & \text{otherwise.}\end{cases}$$

The resulting sampling follows a so-called renewal-process in which all the inter-measurement intervals $\Delta k_j=k_{j+1}-k_j$ are drawn as independent integer random variables exponentially distributed in the interval $[\Delta k_{\min},\infty]$.

The minimum inter-measurement gap $\Delta k_{\min}\geq 1$ depends on the speed of the ADC, which must be ready for a new conversion each time a measurement is taken so that, by increasing $\Delta k_{\min}$ we loosen the constraints on the ADC implementation. The exponential trend is then tuned to have an average inter-measurement gap equal to $N/M$ so that (at least for large $N$) we expect an average of $M$ measurements.

Each of these measurements is commonly quantized by means of a $b$-bit ADC to yield the bit stream passed to the decoder to yield the most straightforward RSAM-ADC option for an AIC (equivalent to the implementation proposed in [6]).

Yet, one may note that, since the time between two measurements changes, the amount of time potentially available for conversion also changes. This can be exploited, for example, by deploying a Successive Approximation Register that, relying on an hardware much simpler than that of a full $b$-bit ADC, exploits the available time to increase conversion accuracy.

This translates in a RSAM–SAR system [4] in which each measurement contributes to the bit stream passed to the decoder with a number of bits that increases as the time before the next measurement increases.

Since, in principle, we increase accuracy by one bit for every iteration of the SAR, the $j$th measurement $y_j$ is quantized with $\Delta k_j$ bits. In this situation, it is sensible to exploit the possibility of embedding quantization into the equations used by minimization-based decoders. This can be done by defining $\Delta y_j=2V^{\max}2^{-\Delta k_j}$, $\Delta y=(\Delta y_0,\dots,\Delta y_{m-1})^\top$, and, as already highlighted in Section III, use $y-\Delta y/2\leq\Theta\hat{\alpha}\leq$

$y + \Delta y/2$ instead of equality constraints in laying formulating the minimization problem (4).

Though RSAM–ADC and RSAM–SAR treat quantization in a different way, they are both subject only to the static saturation due to the finite input range of the conversion stage. This poses no problem since it can be tackled at design time by simply rescaling the signal input range as in conventional acquisition systems.

### C. 1-Bit Compresive Sensing—1b RMPI and RADS

Given a total bit budget $B$, the trade-off between the number of measurements $M$ and the number of bits $b = B/M$ spent to encode each of them is a classical theme in signal acquisition and coding and applies also to CS architectures.

Among other issues, it may help coping with the unavoidable saturation of the ADC since the extreme solution $b = 1$ identifies the ADC with a pure saturation centered in 0, thus completely eliminating the problem.

In particular, RMPI systems may be optimized in each particular setting to see how much information in our original signal can be inserted into $B$ bits [20] and is possible to think that each measurement is represented by a single bit encoding its sign [12], [21].

Since signs give no hint on the magnitude of the involved signals, the problem in (3), with $\epsilon = 0$, with $y = \mathrm{sign}(\Theta\hat{\alpha})$ and where the $\mathrm{sign}(\cdot)$ operator applied component-wise, is recast into [12]

$$\begin{aligned} \min \quad & \|\hat{\alpha}\|_1 \\ \text{s.t.} \quad & y \circ \Theta\hat{\alpha} \geq 0 \\ & \|\Theta\hat{\alpha}\|_2 = 1 \end{aligned} \tag{7}$$

where $\circ$ stands for component-wise product (so that this results in a set of $M$ component-wise inequalities) and the second, unit-energy constraint is introduced as a scale-fixing prior. This approach is referred to in [12] as 1-bit CS or 1bRMPI.

The above optimization problem is a non-convex problem and must be addressed by specialized algorithms. A state-of-the-art version of these algorithms is described in [23] and will be indicated here as `RSS` for *Restricted Step Shrinkage* algorithm. This algorithm is proved to achieve a higher average recovery SNR, and is an order of magnitude faster than other previous proposed algorithms in [12] and [21].

Regrettably, even with `RSS`, typical performance of an 1bRMPI architecture are largely inferior with respect to multi-bit RMPI or RSAM solutions for comparable bit budgets.

In addition to the 1bRMPI architecture, we here analyze an approach that does not require additional priors and provides a performance that is closer to that of other methods [5]. The RAndom Delta-Sigma (RADS) architecture illustrated in Fig. 4 is nothing but a conventional $\Delta\Sigma$ converter whose input signal is pre-multiplied by a random sequence of symbols.

The nonlinear dynamics of the $\Delta\Sigma$ takes the place of the simple integrator in an RMPI architecture so that there is no one-to-one relation between single bits and projections but rather a progressive encoding of widening windows from the original signals. On the one hand, such a technique has the desired effect to allow squeezing amplitude information into a sequence of sign informations. On the other hand, its nonlinearity prevents us from writing a simple linear model linking the signal samples $x_k$ with the bits produced by the encoder and thus seems to fail a cardinal assumption on which the decoding procedures rely. Though this is formally true, we may waive the detailed modeling of the $\Delta\Sigma$ operations to concentrate on its high-level functionality of oversampling converter with noise-shaping abilities to arrive at a model that can be effectively plugged into reconstruction algorithms. More precisely and following [22], the output of the $\Delta/\Sigma$ at time $n$ can be expressed as the sum of the corresponding input sample and of a term accounting for the quantization noise whose spectral profile is dictated by the noise transfer function (NTF) of the converter loop.

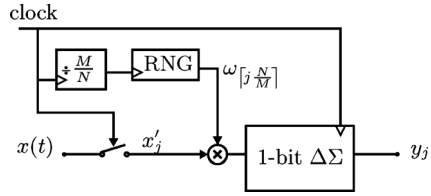



Fig. 4. Block scheme of an randomized $\Delta\Sigma$ encoder. The input signal is multiplied by a random sequence and fed into a $\Delta\Sigma$ converter made of a 1-bit ADC and a loop filter in charge of noise shaping.

To begin with, since we are dealing with 1-bit measurements we have $M = B$ that, in general is much larger than $N$ (in contrast to what typically happens for RMPI and RSAM architectures) and thus implies oversampling.

Hence, $\Delta\Sigma$ operations do not apply to the original components of the vector but to a vector oversampled by a factor $M/N$ that we will indicate as $x' = (x'_0, \ldots, x'_{M-1})^\top$. The two vectors $x'$ and $x$ are linked by $x' = \Upsilon_{\mathrm{RADS}} x$, being $\Upsilon_{\mathrm{RADS}}$ an upsampling matrix that considers the components of $x$ as the Nyquist samples of a bandlimited signal.

With reference to Fig. 4, the samples in $x'$ are multiplied by a Nyquist-rate random sequence $\omega_j$ thus applying a further linear operator that we will indicate with $\Omega$ and is defined by

$$\Omega_{j,k} = \begin{cases} \omega_{\lceil j\frac{N}{M}\rceil}, & \text{if } j = k \\ 0, & \text{if } j \neq k. \end{cases} \tag{8}$$

Hence, the input of the $\Delta\Sigma$ is the vector $\Omega x' = \Omega\Upsilon_{\mathrm{RADS}} x$ and the corresponding output is the binary vector $y = \Omega\Upsilon_{\mathrm{RADS}} x + \zeta$ where $\zeta$ accounts for the quantization noise shaped by the NTF of the loop.

Conventional approaches have $\Omega$ equal to the identity and exploit this construction by noting that low-pass filtering $y$ is equivalent to low-pass filtering $\Omega\Upsilon_{\mathrm{RADS}} x + \zeta = \Upsilon_{\mathrm{RADS}} x + \zeta$ and thus inverts upsampling to recover $x$ with an error equal to the low-pass filtering of $\zeta$, a term that can be made very small by playing with the NTF, i.e., making it as high-pass as possible given other implementation constraints.

In our case, the matrix $\Omega$ introduces spreading so that higher frequency components of the upsampled signal enter the baseband range in which the bits in $y$ are processed. This alias normally prevents signal reconstruction. Yet, sparsity can be exploited to counter alias and allow the acquisition of signal components that would otherwise fall out of the reach of the $\Delta\Sigma$ range (or, conversely, allow smaller oversampling to acquire the same signal).

ALGORITHM I. THE ALGORITHM PROPOSED BASED ON CoSaMP TO INVERT(1) FOR HIGH-PASS $\xi$

FCoSaMP

| | |
|---|---|
| $d_j$ | downsampling ratios such that $M = 2Kd_0d_1\dots d_{J-1}$ |
| $H^{(d)}$ | low-pass filter and downsample by a factor $d$ |
| $\Theta^+$ | conjugate transpose of $\Theta$ |
| $\Theta_T^\dagger$ | pseudoinverse of the columns of $\Theta$ whose indexes are in $T$ |
| $a_{\underline{K}}$ | set to zero all but the $K$ largest components of $a$ |
| $a_T$ | the components of $a$ with indexes in $T$ |
| $a_{\neg T}$ | the components of $a$ with indexes not in $T$ |

$\alpha \Leftarrow (0,\dots,0)^\top$
**for** $j = 1,\dots,J-1$ **do**
  $y \Leftarrow H^{(d_j)}y$
  $\Theta \Leftarrow H^{(d_j)}\Theta$
  $\xi \Leftarrow y - \Theta\alpha$
  **do** $I$ times
    $\zeta \Leftarrow \Theta^+\xi$
    $T \Leftarrow \mathrm{supp}\,(\zeta_{\underline{K}}) \cup \mathrm{supp}\,(\alpha_{\underline{K}})$
    $\beta_T \Leftarrow \Theta_T^\dagger\alpha$
    $\beta_{\neg T} \Leftarrow (0,\dots,0)^\top$
    $\alpha \Leftarrow \beta_{\underline{K}}$
    $\xi \Leftarrow y - \Theta\alpha$
  **end do**
  $T \Leftarrow \mathrm{supp}\,(\alpha_{\underline{K}})$
  $\beta_T \Leftarrow \Theta_T^\dagger\alpha$
  $\beta_{\neg T} \Leftarrow (0,\dots,0)^\top$
  $\alpha \Leftarrow \beta_{\underline{K}}$
**end for**

To see how, recall that $x = \Psi\alpha$ and set $\Theta = \Omega\Upsilon_{\mathrm{RADS}}\Psi$ so that the above high-level model for RADS operations can be mapped into (1) and fall within the reach of classical CS techniques.

Actually, in this case sparsity is only one of the priors we have, the other being the high-pass nature of the disturbance. This further information fits particularly well into algorithms iterating an elementary step that estimates $\mathrm{supp}(\alpha)$ and then calculates the corresponding non-null entries. In these algorithms, it is sensible to low-pass filter (and decimate to remove redundant samples) the input vector at each iteration. Doing so, as the reconstruction proceeds, its refinement happens with values that are progressively less affected by disturbances since, while signal energy decreases linearly as the band shrinks around DC, disturbance energy decreases polynomially thanks to the NTF.

The algorithm we propose to exploit this intuition is reported in Algorithm I and will be referred to as FCoSaMP in the following.

The number of measurements is written as $M = 2Kd_0d_1\dots d_{J-1}$ with $d_j$ being a small downsampling factor (typically 2 or 3) and $J$ being the total number of downsampling steps of the algorithm.

At the $j$th iteration the outer loop filters the signal and downsamples it by a factor $d_j$ to reduce quantization noise. In our case, low-pass filtering was obtained by sinc frequency profiles with lobes matched with the subsampling ratio.

Downsampling continues until the number of available samples is $2K$ since this is the minimum information needed to discriminate between two different $K$-sparse vectors.

The inner loop is performed a fixed number of times $I$ and is based on CoSaMP to iteratively produce an improved estimation of $\alpha$ by least squares over a reduced support made of the support of the previous iteration plus the support of the largest components of the residuals of the previous iteration.

### D. A Note on Hardware Complexity

It is clear that the above-described architectures have different complexities from an hardware point of view. Furthermore, given an architecture, many circuital trade-offs can be arranged by the designer depending on the target application. Since we want to keep our comparison at a high abstraction level, we can consider only a generic amount of resources required, which can be used as an indicator either for circuit area, for power consumption, or for both. An additional hardware parameter which needs to be addressed is the design effort required from the designer to implement the AIC. This is an indicator of how many commercially available components or IP blocks can be used to design the converter, which has strong influence on the the design cost of the circuit and on its possible time-to-market.

Most notably, RMPI is the one requiring probably the least established circuitry (continuous-time or discrete-time analog multiply-and-accumulate blocks) and the largest amount of resources (though serial implementations can be conceived, their performance is largely impaired by projecting disjoint time windows of the signals and that option is not considered here).

On the other extreme, RSAM–ADC and RADS are surely the least complicated to design and implement since they employ standard blocks (a multi-bit ADC or a standard $\Delta\Sigma$) to which a randomization stage is prepended.

Yet, differences in implementation complexity and cost are difficult to assess without reference to constraints of a specific application and are intentionally left out of the scope of the paper.

What we are doing here is to put the various options at trial in a common arena to highlight respective strengths and weaknesses.

Though it is easily expected (and confirmed) that the strengths-versus-weaknesses balance may favor solutions entailing a larger use of resources, we must leave any final judgment to the application-aware designer that will be able to manage the trade-off between performance and implementation effort by also addressing issues like analog/digital partitioning, power budget, etc.

## V. SIMULATION SETTING AND PERFORMANCE FIGURES

We simulated the previously described AIC architectures and corresponding decoding stages in a normalized configuration.

Each input $x$ is generated starting from a sparse vector of coefficients $\alpha$ with $N = 256$ entries, only $K$ of which are non-null at the same time, and assuming that the probability of each of the $\binom{N}{K}$ choices of the non-null components is equally probable. These non-null elements are realizations of independent random variables uniformly distributed in the set $[-1,-(1/2)] \cup [(1/2),1]$. This distribution has been selected in order to avoid components in $\mathrm{supp}(\alpha)$ which are too close to zero.

The base matrix $\Psi$ is taken from the family of matrices $\Lambda^{(n,\ell)}$ defined by

$$\Lambda^{(n,\ell)} = \begin{cases} \Gamma^{(n)}, & \text{if } \ell = 0 \\ \begin{pmatrix} \Lambda^{(n/2,\ell-1)} & 0 \\ 0 & \Lambda^{(n/2,\ell-1)} \end{pmatrix}, & \text{if } \ell > 0 \end{cases}$$

where $\Gamma^{(n)}$ is the $n \times n$ matrix of the orthonormal sinusoidal base for vectors of $n$ samples, i.e.,

$$\Gamma^{(n)}_{j,k} = \frac{1}{\sqrt{n}} \begin{cases} 1, & \text{for } j = 0 \\ \sqrt{2}\cos\left(2\pi \frac{jk}{n}\right), & \text{for } j = 1, \ldots, n/2 - 1 \\ (-1)^k, & \text{for } j = n/2 \\ \sqrt{2}\sin\left(2\pi \frac{jk}{n}\right), & \text{for } j = n/2 + 1, \ldots, n - 1. \end{cases}$$

Most of the simulations are done with $\Psi = \Lambda^{(N,0)}$ that is a common choice to present new architectures and compare performance, yet some numerical evidence for $\Psi = \Lambda^{(N,\ell)}$ with $\ell > 0$ is also reported to assess how much each architecture is sensitive to the concentration of energy along the time axis, a phenomenon that is known to impair acquisition based on compressive sensing (see, e.g., [24]).

In order to allow a fair comparison, the sample vector $x = \Psi\alpha$ is normalized coherently with the design needs of each architecture. Roughly speaking, we are assuming that the input signal level and the converter range are matched with each other. This is what happens also in the design of conventional ADCs, where the system is conceived so to avoid saturation while simultaneously exploiting all the available range not to increase quantization noise.

Note that this assumption allows us also to avoid the introduction of a new parameter representing the ratio between the optimum and the actual maximum level of the input signal, thus avoiding to unnecessarily complicate notation.

In particular, signals entering the RMPI encoders are scaled to have unit energy in the integrating window, since this makes unitary also the variance of the $y_j$ and thus helps assessing the probability that such an accumulation saturates the ADC range. Signals fed into an RSAM encoder are scaled to have unit peak amplitude thus matching the ADC conversion range. For the RADS converter, signals are scaled to have a normalized power since this impacts on the variance of the accumulations entailed in the loop filter of the $\Delta\Sigma$.

Finally, $x$ is perturbed by Gaussian noise with a power controlled by the ISNR parameter, and results for ISNR ranging from 20 to 60 dB are presented.

When a vector $x'$ oversampled by a factor $\upsilon$ is needed, it is computed as $x' = \Upsilon x$, where

$$\Upsilon_{j,k} = \operatorname{sinc}\left(\frac{j}{\upsilon} - k\right).$$

The number of iteration used in `CoSaMP` was set to 200. This value was chosen since it presents a good compromise between accuracy and execution time in our simulation setup. In order to have a similar running time and comparable number of total iterations with that of `CoSaMP`, the number of iterations of the inner loop `FCoSaMP` was select to be $I = 40$. This value also presents a good compromise between accuracy and execution time.

Performance is evaluated by matching the reconstructed vector $\hat{\alpha}$ with the original vector $\alpha$ and using two merit figures: the probability of support reconstruction (PSR) and the average reconstruction signal-to-noise ratio (ARSNR), i.e.,

$$\mathrm{PSR} = \Pr\{\operatorname{supp}(\alpha) \subseteq \operatorname{supp}_{\min\{|\alpha|\}/5}\hat{\alpha}\}$$

$$\begin{aligned} \mathrm{ARSNR(dB)} &= \mathbf{E}\left[\mathrm{dB}\left(\frac{\|\alpha\|_2^2}{\|\alpha - \hat{\alpha}\|_2^2}\right)\right] \\ &= \mathbf{E}\left[\mathrm{dB}\left(\frac{\|x\|_2^2}{\|x - \hat{x}\|_2^2}\right)\right] \end{aligned}$$

where the thresholded support is conventionally defined as

$$\operatorname{supp}_\tau(a) = \{j = 0, \ldots, n-1 | |a_j| \geq \tau\}$$

for some suitably small $\tau$. Probabilities and expectations are estimated by Monte Carlo simulations.

Given the achieved ARSNR of the AIC, it may be sensible to compare the number of bit $B$ used in the AIC with the number of bit required by a standard PCM coding to achieve an output SNR equal to the ARSNR.

In a PCM coding based on Nyquist rate sampling (i.e., with $N$ samples for each time window) and a total number of bit equal to $B_{\mathrm{PCM}}$, we can roughly estimate the output SNR by computing the average number of bits for each sample

$$\mathrm{SNR}_{\mathrm{PCM}}(\mathrm{dB}) = \frac{B_{\mathrm{PCM}}}{N} 6.02.$$

We define the equivalent compression rate $\gamma$ as the ratio between $B_{\mathrm{PCM}}$ and $B$ when $\mathrm{ARSNR} = \mathrm{SNR}_{\mathrm{PCM}}$, i.e.,

$$\gamma = \frac{B_{\mathrm{PCM}}}{B} = \frac{\mathrm{ARSNR(dB)}}{6.02}\frac{N}{B}$$

Clearly, it is convenient to apply CS techniques only if they result in $\gamma \geq 1$ since $\gamma < 1$ implies that the same quality in signal reconstruction could be obtained by a simple PCM entailing a smaller number of total bits. As such, this merit figure is also presented in some of the pictures in Section VI.

The code needed for simulating all the architectures and algorithms that we have sketched before was written for a standard `Matlab`[3] environment and exploits `cplex`[4]. Sources are available at http://www.wuala.com/rrovatti/Shared/JETCAS2012/.

Monte Carlo simulations are taken over respect to 5000 trials.

## VI. Numerical Evidence

The standardized environment described above can be employed to test different options for encoding and decoding in a variety of operating conditions as far as sparsity $K$, ISNR, total bit budgets $B$ and its partition into $M$ measurements.

With that, we may explore some areas of the design space of an AIC and highlight some interesting issues and guidelines.

### A. Comparison Between Decoding Strategies

A first set of simulations was dedicated to compare performance of the algorithms that we took as representatives of the two different approaches to signal reconstruction, i.e., minimization based procedures (`MinL`$_1$) and iterative support guessing procedures (`CoSaMP`).

The support of $\alpha$ is never explicitly computed in minimization-based approaches, which exploit the sparsity promoting

[3][Online] http://www.mathworks.it/products/matlab/

[4][Online] http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/

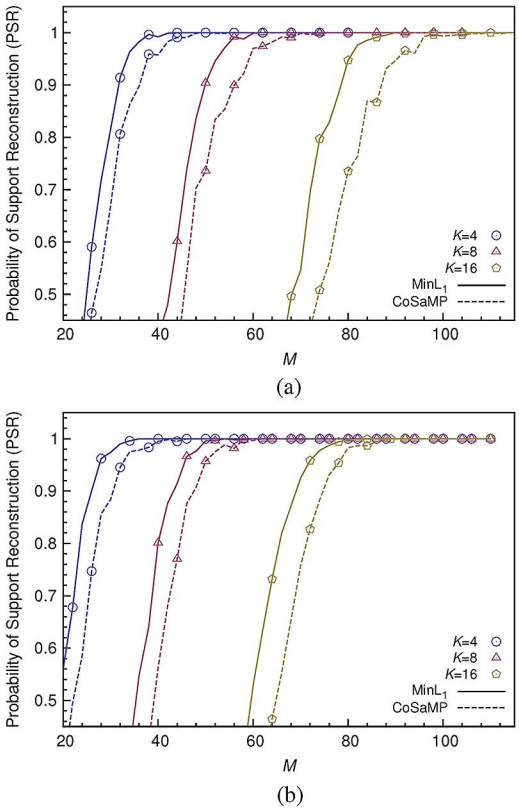



Fig. 5. The PSR achieved by MinL1 and CoSaMP as a function of the number of measurements $M$ for different levels of sparsity $K$ and $\mathrm{ISNR} = 40$ dB matched by $b = 8$ bits per measure. The bit stream comes: (a) from an RMPI encoder, (b) from an RSAM encoder.

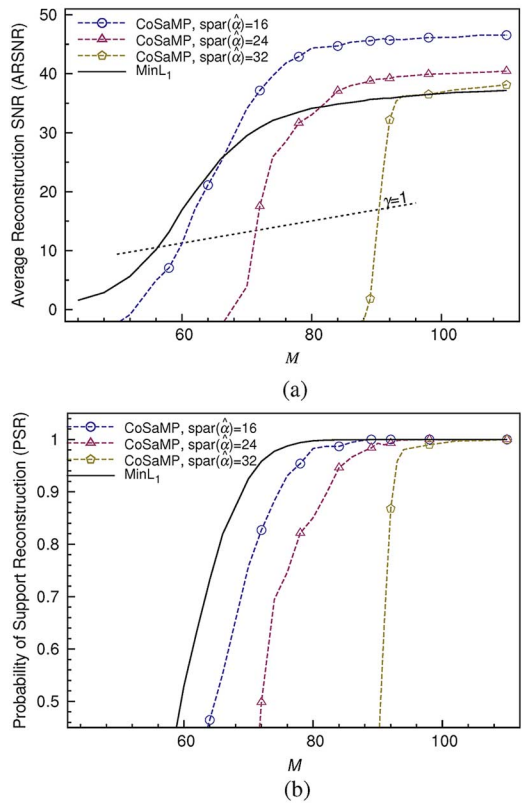



Fig. 6. Performance of CoSaMP as a function of the number of measurements $M$ for $K = 16$ and $\mathrm{ISNR} = 40$ dB matched by $b = 8$ bits per measure when the bit stream comes from an RSAM encoder and CoSaMP is run with different assumptions on $\mathrm{spar}(\hat{\alpha})$. The case of $\mathtt{MinL_1}$ decoding is taken as a reference solid curve.

features of the $\|\cdot\|_1$ norm to simultaneously obtain an estimate of the support and of the non-null entries.

On the contrary, as already mentioned in Section III, iterative support-guessing methods concentrate on the computation of the support that, once known, is used to transform the under-constrained inversion problem (in which the matrix is $M \times N$ with $M < N$) into an over-constrained problem (in which the matrix is $M \times K$ with $M > K$) that can be pseudo-inverted. Since pseudo-inversion of over-constrained problems is very effective in rejecting small disturbances, if the support is correctly guessed then the final estimation $\hat{\alpha}$ is extremely good.

Fig. 5 shows how well support reconstruction works for the two algorithms. The PSR is plotted against the number of measurements for different levels of sparsity (i.e., different difficulty of reconstruction). All the trends clearly highlight a minimum value of $M$ under which reconstruction is ineffective. Yet, as $K$ increases, the optimization-based method exhibits better performance since it achieves the same PSR with a smaller number of measurements (and thus of total bits). Note that this happens independently of the encoder and thus may sensibly be ascribed to the decoding strategy.

Further to that, we investigate whether this difference in PSR affects ARSNR and how much the performance of CoSaMP are affected by possible errors on the value of $K$ with which it is made to work. To do so we consider an RSAM encoder providing a variable number $M$ of measurements, each quantized by $b = 8$ bits, of a signal with $\mathrm{ISNR} = 40$ dB and $K = 16$.

For each $M$ we run CoSaMP instructing it to assume $\mathrm{spar}(\hat{\alpha}) = K = 16$, $\mathrm{spar}(\hat{\alpha}) = 1.5K = 24$, and $\mathrm{spar}(\hat{\alpha}) = 2K = 32$ (no underestimation of $\mathrm{spar}(\alpha)$ is considered since it causes CoSaMP to always perform very poorly).

Fig. 6(a) shows the ARSNR when we focus on an RSAM encoder (qualitative trends are identical for the RMPI) and a CoSaMP decoder based on different assumptions for $\mathrm{spar}(\hat{\alpha})$. The solid curve shows the ARSNR associated to $\mathtt{MinL_1}$ decoding, and can be used as a reference level. CoSaMP is capable to achieve better results in terms of ARSNR with respect to $\mathtt{MinL_1}$ in all considered cases, but at same time it suffers form an higher sensitivity in terms of PSR, which is shown in Fig. 6(b).

It is interesting to note that the reference $\mathtt{MinL_1}$ curve perfectly obeys the theoretical CS framework prediction that the input signal can be recovered from compressed measurement if $M > CK \log_{10}(N/K)$. In fact, if we can consider the signal recovered, once, for example, $\mathrm{PSR} > 0.99$, by setting $C = 4$ as suggested in [8], we get $M > 77.06$ and simulation results from Fig. 6(b) confirms that for $M = 77$ the PSR is equal to 0.991.

When $\mathrm{spar}(\alpha) = K = 16$, we get from Fig. 5(b) that the PSR of CoSaMP is dominated by that of $\mathtt{MinL_1}$. Notwithstanding this, a comparison between the two methods in terms of ARSNR is clearly in favor of CoSaMP when its PSR is at least 0.5 (meaning that $\mathrm{supp}(\alpha)$ is correctly recovered at least 50% or the times). This is due to the pseudo-inversion step in CoSaMP

that is dedicated to (and extremely effective in) computing the values of the non-null components of $\hat{\alpha}$ once that they are the same a those of $\alpha$.

This property holds until problems in the support-guessing step of CoSaMP prevail. This is evident by matching Fig. 6(b) with Fig. 6(a) for $\mathrm{spar}(\hat{\alpha}) > K$. As an extreme case, when $\mathrm{spar}(\hat{\alpha}) = 2K, \mathtt{MinL}_1$ with no knowledge of $K$ is able to consistently outperform CoSaMP in both PSR and ARSNR.

The evidence we collect pushes towards a general guideline favoring CoSaMP-like methods when support-guessing is relatively easy while suggesting $\mathtt{MinL}_1$-like methods when support recovery is the critical issue, for example in biomedical applications.

### B. Gaussian or Antipodal Modulation

The statistics of the random modulating symbols is a degree of freedom in the design of architectures that entail a random modulation, such as RMPI and RADS.

Such a degree of freedom may be effectively exploited to match priors further to sparsity that may be available in certain applications (see, e.g., [16]) but also to ease implementation.

In particular, generating, storing and applying to the sample stream a sequence of Gaussian random variables implies an expensive circuitry that can be avoided by resorting to much simpler antipodal random variables taking values in $\{-1, +1\}$.

Actually, in the RMPI case, the distinction between the two modulations appears to be quite artificial if one considers that what can be decided by the implementation is $\Phi$ that enters the definition of $\Theta = \Phi\Psi$. If $\Psi$ is orthonormal and $\Phi$ is made of independent entries, then $\Theta$ is made of independent rows in which entries (that happens to be linear combinations of independent random variables with a normalized set of coefficients) distribute very similarly to Gaussian random variables. Since this happens independently of the original distribution, Gaussian and antipodal modulations perform in a substantially equivalent way.

In the RADS case, we have $\Theta = \Omega\Upsilon_{\mathrm{RADS}}\Psi$ where the random symbols $\omega_j$ are along the diagonal of $\Omega$ as in (8) and act differently on the original samples making the equivalence between Gaussian and antipodal modulation less evident.

Yet, the evidence we collect by simulation indicates that no true difference exists. Fig. 7 reports the ARSNR obtained when decoding with CoSaMP a bit stream produced either by an RMPI or a RADS encoder in various operating conditions. It is evident that antipodal modulation is always not worse than Gaussian modulation.

Hence, even if Gaussian modulation is often referred to even in implementation oriented contributions (see, e.g., [25], [26]), any optimized design may safely rely on the more hardware-friendly (linear feedback shift registers instead of random-number generators, switches instead of multipliers) antipodal modulation and aim for the same final performance.

### C. RMPI Specific Issues

The most distinctive feature of an RMPI architecture is the presence of multiply-and-accumulate blocks deployed to straightforwardly implement CS principles.

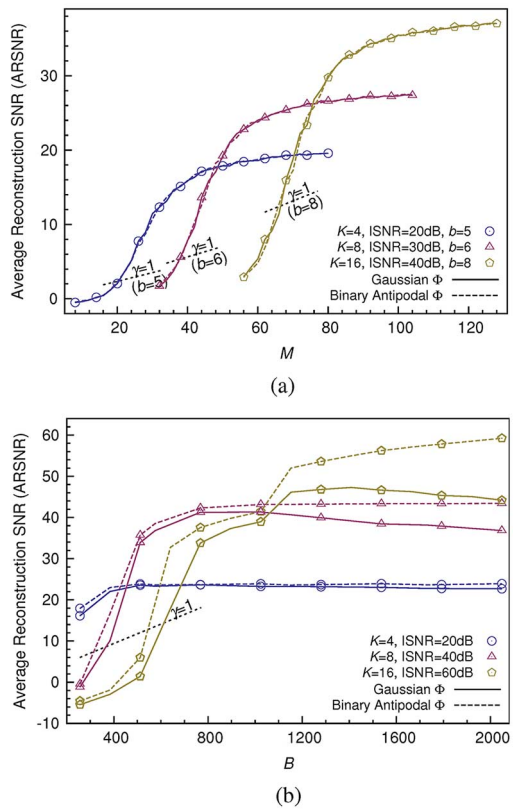



Fig. 7. Performance of random modulating architectures with Gaussian or antipodal modulating symbols plotted against the number of measurements $M$ for different sparsities $K$ and ISNR.(a) Bit streams are produced by an RMPI encoder in which each sample is quantized with $b$ bits to approximately match the corresponding ISNR. (b) Bit streams are produced by a RADS encoder producing $M = B$ 1-bit measurements.

The discussion in Section IV.A already highlighted that such operations are saturation-prone since tend to increase the dynamic range of the signals. When saturation happens the information content of the current measurement may be completely lost if no countermeasure is taken.

Actually, once that the input signal range is set (in terms of its average energy, i.e., variance), the probability of saturation is linked to the amplitude of the ADC range. Since the number of bits output by the ADC is finite, reducing its input range not only eases its implementation and paves the way for some resource saving (for example, the increment in the range of the ADC relaxes noise constraints in the analog stages, allowing a reduction of biasing currents) but also decreases the quantization intervals thus increasing resolution. Hence, joint design of saturation and quantization is a key issue for RMPI architectures that has been at least partially addressed, for example, in [19].

Fig. 8 reports the performance of an RMPI architecture in which the range of the ADC varies thus varying the probability of non-saturation $p_{\neg\mathrm{sat}}$ and, implicitly, the resolution with which measurements are quantized. Decoding is performed by CoSaMP.

The SPD and SPW strategies described in Section IV.A are compared when $M = 96$ measurements are taken in a signal with sparsity $K = 16$, and different condition as far as ISNR and bit-per-measurements $b$ are concerned.

Fig. 8(b) shows that as $b$ increases, SPW becomes more effective in countering the effect of saturation. In fact, when $b = 5$,

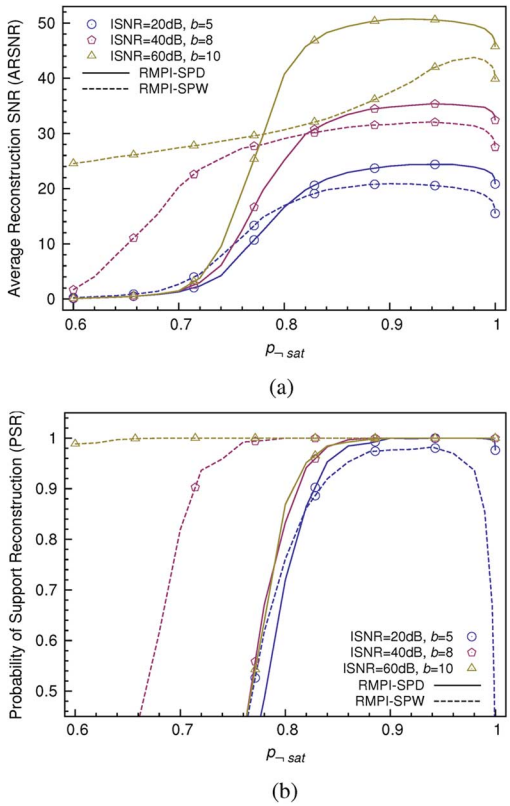



Fig. 8. ARSNR (a) and PSR (b) of an RMPI producing $M = 96$ measurements decoded by CoSaMP for ranges of the ADC such that the probability of non-saturation ($p_{\neg\mathrm{sat}}$) is 60% or above and in a signal with $K = 16$ and various operating conditions as far as ISNR and $b$ are concerned. The SPD curves are obtained by dropping the saturated measurements, while the SPW curves are obtained windowing the projections so that a non-saturated value is always obtained.

SPW does not offer any significant improvement with respect to SPD. This is mainly due to the fact that the 5-bit word encoding each measurement should reserve 1 bit to signal whether a saturation has happened and either 4 bits to encode a non-saturated measurement, or 1 bit to distinguish positive from negative saturation and 3 bits to encode the time at which saturation happened. This extremely coarse quantization of times significantly reduces the amount of information that can be squeezed from saturated measurements and makes SPW practically equivalent to SPD.

When $b = 8$ things are different since SPW is able to make both PSR and ARSNR raise to sensible levels for values of $p_{\neg\mathrm{sat}}$, for instance in the range $[0.7, 0.8]$, significantly smaller than those needed for SPD, and thus for ADC ranges definitely narrower. With reference to the compression ratio $\gamma$, note that with $N = 256, M = 96$ and $b = 8$, CS is a viable opportunity (i.e., enjoys $\gamma > 1$) only if $\mathrm{ARSNR} > 18.06$ dB, a threshold that is achieved for $p_{\neg\mathrm{sat}} \geq 0.68$ by SPW, and for $p_{\neg\mathrm{sat}} \geq 0.78$ by SPD.

Things get drastically different for $b = 10$ since in this case SPW exploits the relative abundance of bits to ensure that every measurement carry a significant information. In this case, CS is sensible only for $\mathrm{ARSNR} > 22.58$ dB, a level that can be achieved by SPW for any $p_{\neg\mathrm{sat}} \geq 0.6$, while SPD still requires $p_{\neg\mathrm{sat}} \geq 0.76$.

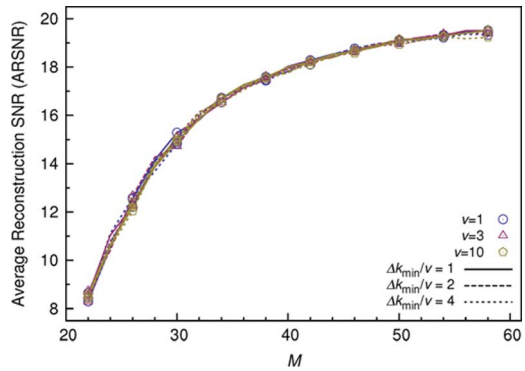



Fig. 9. ARSNR of an RSAM-ADC achieved using $\mathtt{MinL}_1$ for different oversampling ratios $\upsilon$ and related different $\Delta k_{\min}$ on signals with $K = 4$ and $\mathrm{ISNR} = 20$.

Notice also that, as $p_{\neg\mathrm{sat}}$ increases (due to the fact that the ADC range is widened) SPW follows a trend that is eventually dominated by SPD due to the fact that even non-saturated measurement are encoded by $b - 1$ bits instead of $b$ bits.

In all cases, performance decreases as $p_{\neg\mathrm{sat}} \to 1$. This is due to the trade-off between non-saturation and fine quantization. In fact, to increase $p_{\neg\mathrm{sat}}$ one must enlarge the ADC range that eventually becomes so large that the finite number of bits spent to quantize it produces a too coarse subdivision.

### D. RSAM Specific Issues

In principle, the random delay that RSAM architectures insert between measurements may be thought as a real quantity. Hence, the grid on which, for the convenience of the implementation, the $\Delta k_j$ are taken is potentially a design parameter.

Given this temporal grid and indicating with $\upsilon$ its oversampling ratio with respect to the entries of the signal vector $x$, the other key design parameter is $\Delta k_{\min}/\upsilon$. In fact, since no two points on the oversampled grids whose distance is less than $\Delta k_{\min}$ are taken as sampling instants, $\Delta k_{\min}/\upsilon$ defines how slower the ADC in an RSAM architecture can be with respect to one that must convert all the components of $x$ at their original rate.

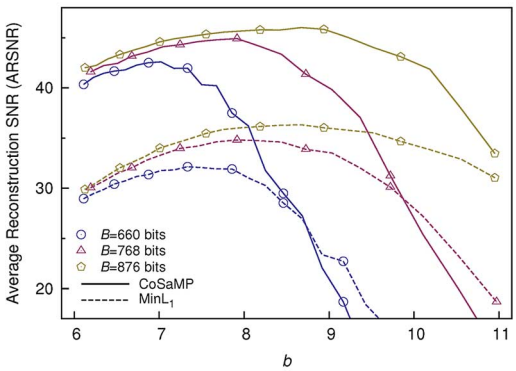
Fig. 9 reports the ARSN of an RSAM bit stream decoded by $\mathtt{MinL}_1$ for different oversampling $\upsilon$ and $\Delta k_{\min}/\upsilon$ where the encoded signals have $\mathrm{ISNR} = 20$ and $K = 4$. The same profile can be observed changing either the decoding strategies or $K$ and ISNR.

The fact that performance appears to be invariant with respect to both parameters suggests that oversampling is actually unnecessary even if samples are not equally spaced in time and that a conspicuous relaxation of the requirements put on the ADC can be obtained until $\Delta k_{\min}/\upsilon$ is not too close to $N/M$.

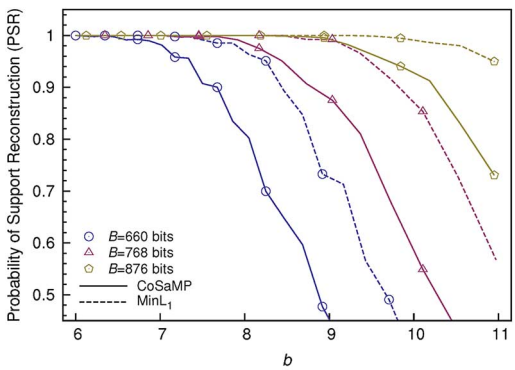
### E. Bit Budget Management

When the total bit budget $B$ is assigned, some CS architectures (RMPI and RSAM in our case) are able to trade the number of measurements $M$ for the depth $b$ of the digital word encoding each of them, thus allowing some performance optimization.

Actually, the $M$-vs-$b$ trade-off may be seen as a further incarnation of an abstract trade-off showing itself in many fields of information processing, like the exploration-versus-exploitation trade-off in nonlinear optimization strategies.

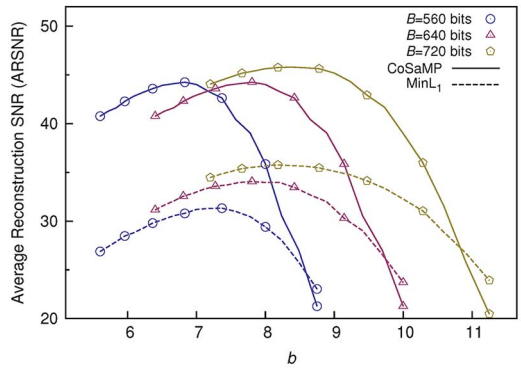



(a)

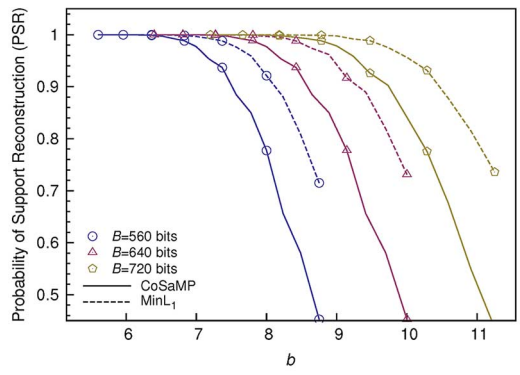



(b)

Fig. 10. Performance of an RMPI architecture in terms of (a) ARSNR and (b) PSR plotted against the number of bits $b$ used to encode each measurement when the total number of bits is constrained by the bit budget $B$. In this case $K = 16, \mathrm{ISNR} = 40$ dB, and $p_{\neg\mathrm{sat}} = 0.9$.




(a)




(b)

Fig. 11. Performance of an RSAM–ADC architecture in terms of (a) ARSNR and (b) PSR plotted against the number of bits $b$ used to encode each measurement when the total number of bits is constrained by the bit budget $B$. In this case $K = 16$ and $\mathrm{ISNR} = 40$ dB.

The key factor here is the ISNR since it can be intuitively accepted that there is little advantage in allocating bits that are then spent to encode the value of a random perturbation.

If this may yield a clear guideline in straightforward acquisition schemes, CS architectures manipulate the signal in different ways before quantization takes place, and the optimum granularity of the ADC may be less easy to anticipate.

Fig. 10 shows what happens to the performance of an RMPI architecture when a certain bit budget is partitioned by using a different number of bits for each measure.

Fig. 11 reports the same data for RSAM-ADC architecture.

Note that all PSR plots are non-increasing in $b$, indicating that support reconstruction is easier when the number of measurements $M = B/b$ is high, i.e., when most resources are devoted to "exploration" to cope with the intrinsically combinatorial problem of support finding.

It is then natural to see that, for the largest possible value of $b$ that maintains such an "exploration" highly effective, i.e., for the rightmost point of the PSR curves that still yields a PSR value close to 1, all ARSNR curves exhibit a maximum. That is the point at which the maximum effectiveness of the "exploration" pairs with the highest possible resolution of the measurement, i.e., with an intense "exploitation" of the available data.

As a first intuition, one may expect that the trade-off between $b$ and $M$ given $B$ which optimizes the ARSNR is expected to be around $b = \mathrm{ISNR}/6.02$, since the SNR of the measurements is equal to the ISNR. This is immediate for the RSAM-ADC; it can also be seen for the RMPI by applying the central limit theorem as in Section IV.A to the noisy input signal as defined in Section II. Yet, performance is influenced by many other parameters, which may move the optimum far from this expected point.

By looking at the numerical results in Figs. 10 and 11 we can see how the optimum point it is affected by $B$. When $B$ is small it may be sensible to reduce $b$ and increase $M$ in order to better satisfies some CS constraints such as the restricted isometry property. On the contrary, when $B$ is large, one could get more advantages by increasing $b$ with respect to increasing $M$, since when the number of measurement is large enough to allow signal reconstruction it is certainly preferable increasing measurements quality instead of their quantity.

As a final remark, recall that while RSAM-ADC allows to optimize the trade off between $M$ and $b$, RSAM–SAR automatically assigns a different number of bits to every measurement.

To know whether this automatic assignment is beneficial to the general performance, or must be only suffered as a side-effect of a lighter architecture (a SAR compared to a full ADC) we compare RSAM-ADC and RSAM–SAR in Fig. 12.

The performance of RSAM–ADC always dominates that of RSAM–SAR whose automatic bit allocation does not help reconstruction, especially for large ISNR values and a small $\Delta k_{\min}$ that sets the minimum for the number of bits of each measurement. Note that the performance of the RSAM–SAR are improved by increasing $\Delta k_{\min}$, thus pushing bit allocation

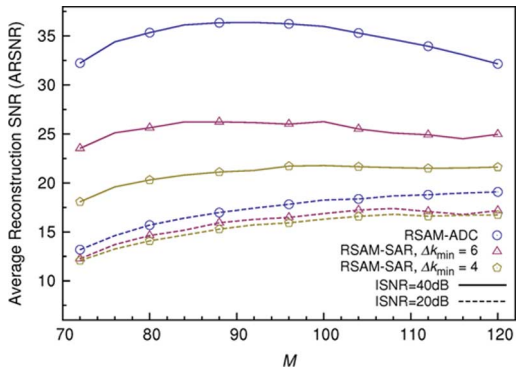



Fig. 12. Performance of an RSAM–ADC and of an RSAM–SAR architecture in terms of ARSNR plotted against the number of measurements for different ISNR, with sparsity $K = 16$ and a the total bit budget $B = 768$. The number of bits for each measurement for the RSAM–SAR depends on the random inter-measurement interval and on $B$.

towards the uniformity that characterizes the RSAM–ADC option.

For small values of ISNR the performance difference is strongly reduced. In the specific case of the figure $\Delta k_{\min}$ is large enough to ensure a quantization noise smaller with respect to the ISNR, and both architectures reach an ARSNR value similar to the ISNR.

As a conclusion, the trade-off between $M$ and $b$ is a complex issue, and the optimum point may not be coincident with what is suggested by intuition, thus hinting the fact that the signal processing before quantization affects the trade-off. Furthermore, this optimization is strongly depended on the ISNR, making a signal independent analysis impossible.

### F. 1 bit Architectures Issues

1-bit CS bases its appeal on its extremely simple implementation that practically avoids analog-to-digital conversion and all its related issues, such as saturation and quantization.

The price for this is a loss of performance that may require a special care not to impair overall applicability.

Fig. 13 reports the comparison between an 1bit RMPI AIC paired with an `RSS` decoder and a RADS encoder paired with an `FCoSaMP` decoder when working on signals of different sparsity $K$ and different ISNR.

Note that, despite the fact that hardware complexity of 1bit RMPI is in principle larger than that of RADS, the performance of the former are strictly dominated by the latter.

Actually, for most bit budget values, adopting an 1 bit RMPI architecture may not be convenient since performance lies below the $\gamma = 1$ line identifying improvement in terms of compression with respect to conventional acquisition strategies. On the contrary, RADS achieves $\gamma > 1$ for a wide rage of bit budgets and operating conditions thus reducing the gap between 1 bit compressive sensing architectures and methods that rely on a larger amount of resources.

### G. General Comparison and Discussion

As a final step we compare the considered architectures to establish what is the best performing in a certain number of operative conditions.

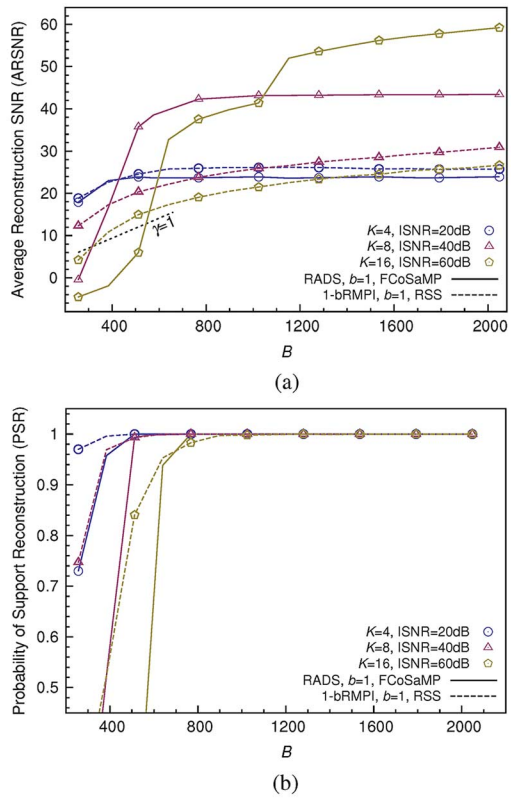



Fig. 13. Performance of a RADS architecture with an `FCoSaMP` decoder compared to a 1-bit RMPI with a `RSS` decoder plotted against the total bit budget $B$ in different operating conditions as far as sparsity $K$ and ISNR are concerned.

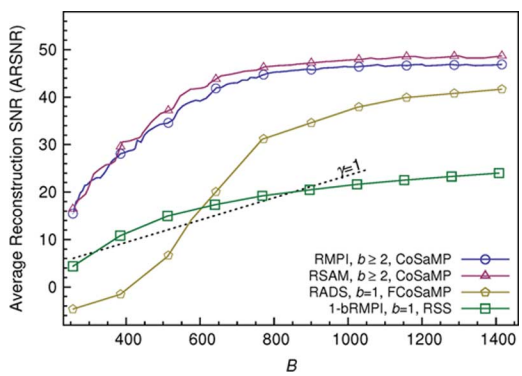



Fig. 14. Largest achievable ARSNR for each architecture plotted against the total bit budget $B$.

Results are reported in Fig. 14 for the case in which $K = 16$ and $\mathrm{ISNR} = 40$ dB assuming that $K$ is also known at the decoder so that `CoSaMP` is the best decoding choice.

For each bit budget $B$, we plot the maximum ARSNR that can be obtained by (numerically) optimizing $M$ (and thus $b = B/M$) for RMPI and for RSAM–ADC. RMPI is designed so that $p_{\neg\mathrm{sat}} = 0.9$. RADS and 1bit RMPI are also considered. This plot allows us to see which of the architectures achieve the best compression *versus* reconstruction quality trade-off.

By comparison with the $\gamma = 1$ line, it is evident that more than one architecture makes AIC convenient with respect to

straightforward acquisition. It is also clear that performance increases as the resources spent in acquisition increase thus allowing an informed trade-off once that application specific constraints are known.

Though these performance evaluation are significative, one of the issues that is not yet clarified is how much they are "robust" with respect to the basis along which the original signal is sparse.

It can be intuitively accepted that the sinusoidal basis we use is a good representative of bases whose waveforms cover most of of the time interval in which the signal is observed. Yet, it is well known [24] that if the waveforms of the basis exhibits some kind of concentration along the time axis, then compressive sensing becomes increasingly more difficult (more formally, it requires an increasing number of measurements).

The formal argument quantifying the above fact can be paralleled by an independent intuitive argument. Despite the fact that for abstract signal processing basis choice is arbitrary, all operations eventually get implemented by time-domain operations. If these operations are not time-invariant then the amount of energy collected by the system from the incoming signals can vary depending on its support.

From this we get that strategies like RSAM or RADS are less robust with respect to energy concentration along the time axis since they heavily rely on non-time-invariant processing: RSAM samples at single points ignoring the signal even in a strict neighborhood of the sampling instant, while RADS exploits memory and long term averages to perform noise shaping and increase overall linearity.

This is fully confirmed by the numerical evidence reported in Fig. 15 in which we adopt $\Psi = \Lambda^{(256,\ell)}$ for $\ell = 0, 1, 2, 3, 4$. This produces bases whose waveforms cover $2^\ell$ of the total time range of $N = 256$ instants.

Simple visual inspection reveals that RMPI-based processing is completely insensitive to the support of the basis waveforms while the performance of non-time-invariant methods such as RSAM and RADS suffer a consisten performance decrease as that support decreases.

Obviously, this is a key feature to keep in mind when selecting an AIC architecture for a particular application since, for example, suggests the adoption of RMPI-based strategies for signals like ECGs (that are usually decomposed along Gabor atoms whose support is only a fraction of, say, an heartbeat time) [26], [25], [27], while allowing an extremely parsimonoius acquistion of EEGs whose sparsity can be revealed in close-to-Fourier bases [5].

## VII. Conclusion

We took some steps in clarifying relative strengths and weaknesses of some of the recently proposed architectures for hardware implementation of AICs based on CS.

Though theoretical support to CS techniques is in rapid evolution and gives some fundamental guarantees on the fact that CS is feasible and may be advantageous over conventional acquisition strategies, coping with the most basic nonlinearity unavoidable in any real implementation (i.e., saturations and quantization of the measurements) and yield fair quantitative comparison required extensive simulations in an homogenous environment.

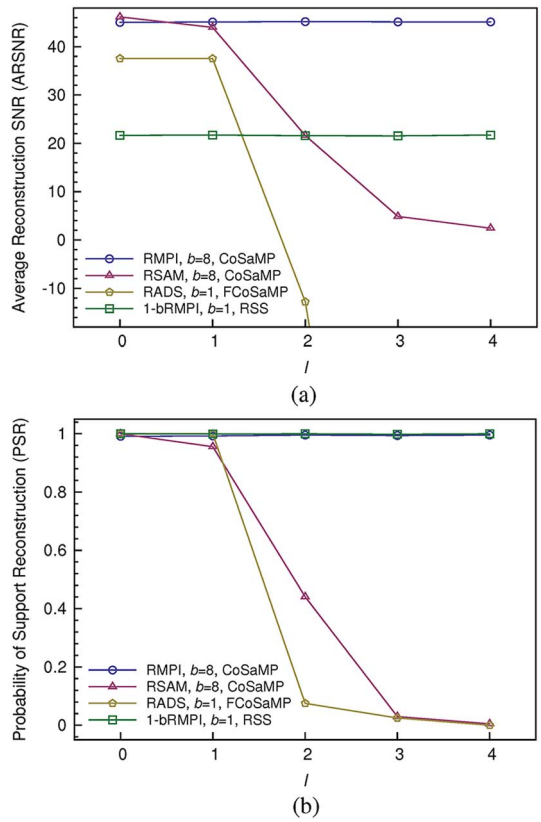



Fig. 15. Performance of RMPI, RSAM-ADC, 1bit RMPI and RADS when acquiring a signal characterized by $\mathrm{ISNR} = 40$ dB with $M = 96$ measurements quantized with $B = 768$ bits. The signal is sparse with respect to a basis whose waveforms have a support covering a fraction of the acquisition interval equal to $2^\ell$ for $\ell = 0, 1, \ldots, 4$.

TABLE I
Summary of Advantages and Disadvantages of the Different Architecture Analyzed in This Paper, Both From the Signal Processing Point of View and From Hardware Complexity Point of View

| | RMPI | RSAM | 1bRMPI | RADS |
|---|---|---|---|---|
| Performance: | | | | |
| • compression *vs* quality | + | + | −− | − |
| • sparsity base robustness | ++ | − | ++ | − |
| Hardware complexity: | | | | |
| • resources required | −− | + | − | ++ |
| • design effort | − | + | − | + |

Based on numerical evidence, we were able to discuss some typical issues in AIC design, such as architecture selection for a particular operative setting, or the partitioning of a total bit budget into digital words corresponding to individual measurements. Pros and cons of each architecture have been discussed, evaluating both the hardware complexity of all solutions (see Section IV-D) and the performance in terms of ARSNR and PSR (see Section VI-G). This comparison has been briefly summarized in Table I.

We were also able to at least envision some more general guidelines such as, for example, that the RMPI achitecture and its recently proposed adjustments are probably the most versatile approach though not always the most economic to implement, or that, when 1-bit quantization is sought, dynamically

mixing quantization and integration in a randomized $\Delta\Sigma$ archicture, and designing a slightly specialized reconstruction algorithm, help yielding performance much closer to that of multi-bit approaches.

## REFERENCES


[1] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[2] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Proc. Mag.*, vol. 25, pp. 21–30, 2008.

[3] J. N. Laska, S. Kirolos, M. F. Duarte, T. S. Ragheb, R. G. Baraniuk, and Y. Massoud, "Theory and implementation of an analog-to-information converter using random demodulation," in *IEEE Int. Symp. on Circuits Syst.*, 2007, pp. 1959–1962.

[4] C. Luo and J. H. McClellan, "Compressive sampling with a successive approximation ADC architecture," in *IEEE Int. Conf. on Acoust., Speech, Signal Proc. (ICASSP)*, May 2011, pp. 3920–3923.

[5] J. Haboba, M. Mangia, R. Rovatti, and G. Setti, "An architecture for 1-bit localized compressive sensing with applications to EEG," in *Proc. IEEE Biomed. Circuits Syst. Conf.*, 2011, pp. 137–140.

[6] P. K. Yenduri, A. C. Gilbert, M. P. Flynn, and S. Naraghi, "Rand ppm: A low power compressive sampling analog to digital converter," in *IEEE Int. Conf. on Acoust., Speech, Signal Proc. (ICASSP)*, May 2011, pp. 5980–5983.

[7] M. Mangia, R. Rovatti, and G. Setti, "Analog-to-information conversion of sparse and non-white signals: Statistical design of sensing waveforms," in *IEEE Int. Symp. on Circuits Syst.*, 2011, pp. 2129–2132.

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

[9] E. J. Candés, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Commun. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, Aug. 2006.

[10] A. Maleki and D. L. Donoho, "Optimally tuned iterative reconstruction algorithms for compressed sensing," *IEEE J. Sel. Topics Signal Proc.*, vol. 4, no. 2, pp. 330–341, 2010.

[11] D. Needell and J. A. Tropp, "CoSaMP iterative signal recovery from incomplete and inaccurate samples," *Appl. Computat. Harmonic Anal.*, vol. 26, pp. 301–321, 2009.

[12] P. T. Boufounos and R. G. Baraniuk, "1-bit compressive sensing," in *42nd Annu. Conf. on Inf. Sci. Syst.*, 2008.

[13] J. N. Laska, P. T. Boufounos, M. A. Davenport, and R. G. Baraniuk, "Democracy in action: Quantization, saturation, and compressive sensing," *Appl. Computat. Harmonic Anal.*, vol. 31, pp. 429–443, Nov. 2011.

[14] R. G. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, "A simple proof of the restricted isometry property for random matrices," *Constructive Approx.*, vol. 28, pp. 253–263, 2008.

[15] S. Mendelson, A. Pajor, and N. Tomczac-Jaegermann, "Uniform uncertainty principle for Bernoulli and subGaussian ensembles," *Constructive Approx.*, vol. 28, pp. 277–289, 2008.

[16] M. Mangia, R. Rovatti, and G. Setti, "Rakeness in the design of analog-to-information conversion of sparse and localized signals," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 59, no. 5, pp. 1001–1014, May 2012.

[17] S. Haykin, *Adaptive Filter Theory*. Upper Saddle River: Prentice Hall, 2002.

[18] P. McSharry, G. Clifford, L. Tarassenko, and L. Smith, "A dynamical model for generating synthetic electrocardiogram signals," *IEEE Trans. Biomed. Eng.*, vol. 50, no. 3, pp. 289–294, 2003.

[19] M. Mangia, F. Pareschi, R. Rovatti, G. Setti, and G. Frattini, "Coping with saturating projection stages in RMPI-based compressive sensing," in *IEEE Int. Conf. on Circuits Syst.*, May 2012, pp. 2805–2808.

[20] J. N. Laska and R. G. Baraniuk, "Regime change: bit-depth versus measurement-rate in compressive sensing," *IEEE Trans. Signal Proc.*, vol. 60, no. 7, pp. 3496–3505, Jul. 2012.

[21] P. T. Boufounos, "Greedy sparse signal reconstruction from sign measurements," in *Proc. 43rd Asilomar Conf. on Signals, Syst. Computers*, 2009, pp. 1305–1309.

[22] R. Schreier and G. Temes, *Understanding Delta-Sigma Data Converters*. Pisacataway, NJ: Wiley-IEEE Press, 2004.

[23] J. N. Laska, Z. Wen, W. Yinm, and R. G. Baraniuk, "Trust, but verify: Fast and accurate signal recovery from 1-bit compressive measurements," *IEEE Trans. Signal Proc.*, vol. 59, no. 11, pp. 5289–5301, Nov. 2011.

[24] E. Candés and J. Romberg, "Sparsity and incoherence in compressive sensing," *Inverse Problems*, vol. 23, pp. 969–985, 2007.

[25] E. G. Allstot, A. Y. Chen, A. M. R. Dixon, D. Gangopadhyay, and D. J. Allstot, "Compressive sampling of ECG bio-signals: Quantization noise and sparsity considerations," in *Proc. 2010 IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, 2010, pp. 41–44.

[26] H. Mamghanian, N. Khaled, D. Atienza, and P. Vandergheynst, "Compressed sensing for real-time energy-efficient ECG compression on wireless body sensor nodes," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 9, pp. 2456–2466, Sep. 2011.

[27] M. Mangia, J. Haboba, R. Rovatti, and G. Setti, "Rakeness-based approach to compressed sensing of ECGs," in *IEEE Int. Symp. on Biomed. Circuits Syst.*, Nov. 2011, pp. 424–427.


**Javier Haboba** (S'11) received the Electrical Engineering degree from the National University of La Plata, Argentina in 2005, and is currently working toward the Ph.D. degree in information technology at the University of Bologna, Italy, mainly studying analog-to-information converter architectures, and its hardware implementation.

Since 2005, he has worked in a variety of different projects at private companies and academic institutions. Most notably, in 2009–2010, while working at the Argentinean Institute of Radioastronomy, he designed and developed an FPGA based digital back-end for a beam-forming radar. He has also made research stages at the Technion, Israel Institute of Technology where he specialized in measurement and characterization of MOS devices in 2007. More recently, in 2011, he founded Bife Supercomputing S.A., a startup company specialized in the development of Low-cost High Performance computing system for individuals, where he received multiple national and international prizes for its innovative ideas in this area. His research interests include Compressive Sensing, Statistical Signal Processing, High Performance algorithm implementation of signal processing systems and parallel computing.

**Mauro Mangia** (S'10) was born in Lecce, Italy. He received the B.S. and M.S. degree in electronic engineering from the University of Bologna, Italy, in 2004 and 2009, respectively, and is currently working toward the Ph.D. degree in information techology, under the European Doctorate Project (EDITH) from University of Bologna, Italy.

In 2009, he was a Visiting Scholar at Non-Linear System Laboratory of the École Polytechnique Fédérale de Lausanne (EPFL), Lasaunne, Switzerland. His research interests are in nonlinear systems, compressed sensing, ultra wide band systems and system biology.

Mr. Mangia was the recipient of the Best Student Paper Award at ISCAS2011.

**Fabio Pareschi** (S'05–M'08) received the Dr. Eng. degree (with hons) in electronic engineering from University of Ferrara, Italy, in 2001, and the Ph.D. degree in information technology under the European Doctorate Project (EDITH) from University of Bologna, Italy, in 2007.

He is currently with Department of Engineering (ENDIF), University of Ferrara, Ferrara, Italy, and also with Advanced Research Center on Electronic Systems, University of Bologna, Bologna, Italy. In 2006, he spent six months as a Visiting Scholar at the Department of Electrical Engineering of the Catholic University of Leuven, Belgium. His research activity is mainly focused on Analog and Mixed Mode electronic circuit design (in particular on nonlinear and chaotic circuit implementation), statistical signal processing, random number generation and testing, and electromagnetic compatibility.

Dr. Pareschi was co-recipient of the Best Paper Award at ECCTD2005 and also the best student paper award at EMCZurich2005. He is currently Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS.

**Riccardo Rovatti** (M'99–SM'02–F'12) received the M.S. degree (*summa cum laude*) in electronic engineering and the Ph.D. degree in electronics, computer science, and telecommunications from the University of Bologna, Bologna, Italy, in 1992 and 1996, respectively.

Since 2001, he has been Associate Professor of Electronics with the University of Bologna, Bologna, Italy. He is the author of almost 300 technical contributions to international conferences and journals and of two volumes. He is coeditor of the book *Chaotic Electronics in Telecommunications* (CRC press, 2000) as well as one of the guest editors of the May 2002 special issue of the PROCEEDINGS OF THE IEEE on "Applications of Non-linear Dynamics to Electronic and Information Engineering." His research focuses on mathematical and applicative aspects of statistical signal processing especially those concerned with nonlinear dynamical systems.

Prof. Rovatti was an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS. In 2004, he received the Darlington Award of the IEEE Circuits and Systems Society.

**Gianluca Setti** (S'89–M'91–SM'02–F'06) received the Dr. Eng. degree (with hons) in electronic engineering and the Ph.D. degree in electronic engineering and computer science from the University of Bologna, Bologna, Italy, in 1992 and in 1997, respectively, for his contribution to the study of neural networks and chaotic systems.

From May 1994 to July 1995, he was with the Laboratory of Nonlinear Systems (LANOS) of the Swiss Federal Institute of Technology, Lausanne (EPFL), Switzerland, as visiting researcher. Since 1997, he has been with the School of Engineering, University of Ferrara, Ferrara, Italy, where he is currently a Professor of Circuit Theory and Analog Electronics. He held several visiting position at Visiting Professor/Scientist at EPFL (2002, 2005), UCSD (2004), IBM T. J. Watson Laboratories (2004, 2007) and at the University of Washington, Seattle (2008, 2010), and is also a permanent faculty member of ARCES, University of Bologna. His research interests include nonlinear circuits, recurrent neural networks, implementation and application of chaotic circuits and systems, statistical signal processing, electromagnetic compatibility, wireless communications and sensor networks. He is co-editor of the book *Chaotic Electronics in Telecommunications* (CRC Press, 2000), *Circuits and Systems for Future Generation of Wireless Communications* (Springer, 2009) and *Design and Analysis of Biomolecular Circuits* (Springer, 2011), as well as one of the guest editors of the May 2002 special issue of the PROCEEDINGS OF THE IEEE on "Applications of Non-linear Dynamics to Electronic and Information Engineering".

Dr. Setti received the 1998 Caianiello prize for the best Italian Ph.D. dissertation on Neural Networks, and is co-recipient of the 2004 IEEE CAS Society Darlington Award, as well as of the Best Paper Award at ECCTD2005 and the Best Student Paper Award at EMCZurich2005 and at ISCAS2010. He served as an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I : REGULAR PAPERS (1999–2002 and 2002–2004) and for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS (2004–2007), the Deputy-Editor-in-Chief, for the *IEEE Circuits and Systems Magazine* (2004–2007) and as the Editor-in-Chief for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS (2006–2007) and of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS— I : REGULAR PAPERS (2008–2009). He was the 2004 Chair of the Technical Committee on Nonlinear Circuits and Systems of the of the IEEE CAS Society, a Distinguished Lecturer (2004–2005), a member of the Board of Governors (2005–2008), and he served as the 2010 President of the same society. He was the Track Chair for Nonlinear Circuits and Systems of ISCAS2004 (Vancouver, Canada), the Special Sessions Co-Chair of ISCAS2005 (Kobe, Japan) and ISCAS2006 (Kos, ), the Technical Program Co-Chair of NDES2000 (Catania, Italy), ISCAS2007 (New Orleans, LA), ISCAS2008 (Seattle, WA), ICECS2012 (Seville, Spain), as well as the General Co-Chair of NOLTA2006 (Bologna, Italy).