Compressed Sensing using Chaos Filters

Nguyen Linh-Trung*, Dinh Van Phong*, Zahir M. Hussain†, Huu Tue Huynh‡, Victoria L. Morgan§, and John C. Gore§

*College of Technology, Vietnam National University, 144 Xuan Thuy, Cau Giay, Hanoi, Vietnam, nltrung@vnu.edu.vn
†School of Electrical and Computer Engineering, RMIT University, 124 Latrobe Street, Melbourne, VIC 3000, Australia
‡Bac Ha International University, 103, D5 Nguyen Phong Sac noi dai, Cau Giay, Hanoi, Vietnam
§Institute of Imaging Science, Vanderbilt University, 1161 21st Avenue South, Nashville, TN-37232, USA

***Abstract*—Compressed sensing, viewed as a type of random undersampling, considers the acquisition and reconstruction of sparse or compressible signals at a rate significantly lower than that of Nyquist. Exact reconstruction from incompletely acquired random measurements is, under certain constraints, achievable with high probability. However, randomness may not always be desirable in certain applications. Taking a nonrandom approach using deterministic chaos and following closely a recently proposed novel efficient structure of chaos filters, we propose a chaos filter structure by exploring the use of chaotic deterministic processes in designing the filter taps. By numerical performance, we show that, chaos filters generated by the logistic map, while being possible to exactly reconstruct original time-sparse signals from their incompletely acquired measurements, outperforms random filters.**

***Index Terms*—Compressed sensing, random undersampling, random filters, chaos filters, chaotic undersampling.**

I. NONRANDOMNESS IN COMPRESSED SENSING?

Compressed sensing (CS), recently been introduced by Candes and Tao [1] and Donoho [2] as a type of random undersampling, allows for the acquisition and reconstruction of *sparse/compressible* signals at a rate lower than that of Nyquist. First, *random* linear projection is used to acquire efficient representations of the signals directly. Then, nonlinear techniques, such as l_1 optimization-based algorithms or sparse approximation algorithms, are used to faithfully reconstructed the acquired compressed data under certain constraints.

CS framework has attracted an overwhelming research attention, due to several advantages. In data compression, significant reduction of data storage can be obtained thanks to the incomplete measurements. In data acquisition, much less power consumption is used at the sensing device since simple and less signal acquisition is performed while the computational load is pushed toward the reconstruction side. In addition, it can trade for fundamental limits of physical systems, such as the limited sampling clock frequency of ADCs used in sensing extremely high frequency spectrum holes in ultra-wideband cognitive radio communications [3].

The use of randomness, which leads to the so-called incoherence property in CS, facilitates the faithful reconstruction with high probability. In practice, however, the use of purely random undersampling is expensive in hardware design. Concerned with the undesirable randomness in such situations, a question naturally arises: Can we use a *nonrandom* structure that still mimics or approximates the incoherence property for compressed sensing framework? Our approach to find an answer to this question is to explore the use of *deterministic chaos*. A chaos system is a nonlinear system that has a very unstable structure so that, under specific initial and control conditions, the output of the system behaves as random in just a few steps. Chaos have been studied in various scientific and engineering contexts [4], and recently found various interesting applications in communications [5]. In addition, since chaos are just deterministic equations, they can be easily implemented on hardware, as opposed to random sequences. It is the random-like behavior of chaos and their advantages over random sequences that leads to our study of chaos filter for CS in this paper. In particular, inspired from the efficiency of random filters proposed in CS [6], we propose in this paper chaos filters for CS that have the same structure as random filters, except that the filter tap values are generated by deterministic chaos.

II. BACKGROUND ON COMPRESSED SENSING

Consider a discrete-time signal $\mathbf{x} \in \mathbb{R}^N$ and assume that $\mathbf{x}$ is K-sparse in the N-dimensional space spanned by the set of N basis vectors $\{\boldsymbol{\psi}_i\}_{i=1}^N$, that is:

$$\mathbf{x} = \sum_{i=1}^{N} \boldsymbol{\psi}_i s_i = \boldsymbol{\Psi}\mathbf{s}, \tag{1}$$

where $\boldsymbol{\Psi} = [\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_N]$ is the sparsifying matrix, and $\mathbf{s} = [s_1, \ldots, s_N]^T$ is the transform vector, containing exactly K nonzero coefficients, $K \ll N$. Examples of commonly used $\boldsymbol{\Psi}$ are Fourier transform, Discrete-Cosine transform, and Wavelet Transform. Note that when $\mathbf{x}$ is a time-sparse signal, i.e. sparse in the time domain, then $\boldsymbol{\Psi} = \mathbf{I}$. For the simplicity of presentation, we restrict ourselves to sparsity, rather than the more general case of compressibility. In the framework of CS, $\mathbf{x}$ is linearly acquired by an underdetermined system,

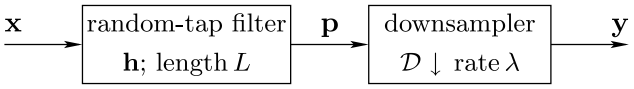

Fig. 1. Random filters for compressive signal acquisition.

represented by a measurement matrix $\mathbf{\Phi}$. The obtained measurements $\mathbf{y} \in \mathbb{R}^M$, $M < N$, are then given by

$$\mathbf{y} = \mathbf{\Phi x} = \mathbf{\Phi \Psi s} = \mathbf{\Theta s}. \tag{2}$$

Given $\mathbf{y}$, $\mathbf{\Phi}$ and $\mathbf{\Psi}$, the objective is then to faithfully recover $\mathbf{x}$ (and hence $\mathbf{s}$) from $\mathbf{y}$ with as small M as possible. If the sparsity information in $\mathbf{x}$ is still *fully* kept, though being hidden, in $\mathbf{y}$, exact reconstruction of $\mathbf{s}$ is feasible if we find a way to fully restore this sparsity from $\mathbf{y}$. It has been proved that if $\mathbf{\Theta}$ satisfies the so-called Restricted Isometry Property, then the sparsity information is maintained; in other words, $\mathbf{\Phi}$ is *incoherent* with $\mathbf{\Psi}$ in the sense that one cannot sparsify the other [7]. One way to ensure the incoherence is to have $\mathbf{\Phi}$ as a random matrix with Gaussian i.i.d. elements. Under such a condition $\mathbf{s}$ can be faithfully recovered from $\mathbf{y}$ when M is such that $cK \log(N/K) < M < N$, where c is some constant, using various sparse approximation techniques, for examples, l_1-optimization based Basis Pursuit (BP) [1] or Orthogonal Matching Pursuit (OMP) [8].

III. Random filters for CS

Recently, a novel random filter structure has been proposed to efficiently acquire random measurements $\mathbf{y}$, by Tropp *et al.* in [6]. First, a random filter h of length L is an FIR filter whose taps are i.i.d. random variables drawn either from the Gaussian distribution with zero mean and unit variance. On the encoding side, $\mathbf{x}$ is first filtered by a the random filter $\mathbf{h}$, and then downsampled at a rate $\lambda = \lfloor N/M \rfloor$, as shown in Fig. 1. The output of the downsampler is the M linear measurements of signal $\mathbf{x}$, that is described mathematically in convolution form as

$$y(m) = \sum_{k=0}^{L-1} x(m\lambda + k) h(L-k), \tag{3}$$

for $m = 0, \ldots, M-1$, or in matrix form as

$$\mathbf{y} = \mathbf{H}_\lambda \mathbf{x}, \tag{4}$$

where $\mathbf{H}_\lambda$, of size $M \times N$, is extracted from the usual linear convolution matrix $\mathbf{H}$, of size $(N+L-1)\times N$, by keeping only rows that are λ spaces apart. In fact, $\mathbf{H}_\lambda$ has a banded quasi-Toeplitz structure. By comparing (2) and (4), it is obvious that $\mathbf{H}_\lambda$ is viewed as the measurement matrix $\mathbf{\Phi}$ within the compressed sensing setting.

On the decoding side, the reconstruction of $\mathbf{x}$ from $\mathbf{y}$ is then done using OMP which take several general steps as shown in Algorithm 1. Additional information can be found in [6], for example the use of the Bernoulli/Rademacher

Algorithm 1 Random filter reconstruction

Inputs: $\mathbf{y}$, $\mathbf{H}_\lambda$, $\mathbf{\Psi}$
1. Initialize residual: $\mathbf{r}_0 = \mathbf{x}$.
For $m = 1, \ldots, M$, do:
2. Find column index i_m of $\mathbf{\Theta} = \mathbf{H}_\lambda \mathbf{\Psi}$ such that

$$i_m = \arg\max_i |\langle \mathbf{r}_{m-1}, \mathbf{\Theta}_i \rangle|$$

3. Compute new residual: $\mathbf{r}_m = \mathbf{y} - \mathbf{P}_m \mathbf{y}$
($\mathbf{P}_m$- orthogonal projector onto span of $\mathbf{\Theta}_{i_1}, \ldots, \mathbf{\Theta}_{i_m}$)
Outputs: $\mathbf{s}$, obtained from $\mathbf{\Theta}_{i_1}, \ldots, \mathbf{\Theta}_{i_M}$ and coefficients $\hat{\theta}_{i_1}, \ldots, \hat{\theta}_{i_M}$ such that: $\mathbf{P}_M \mathbf{y} = \sum_{m=1}^{M} \hat{\theta}_{i_m} \mathbf{\Theta}_{i_m}$

distribution to generate filter tap values, the use FFT to efficient implementation of the convolution, and discussion on the efficiency of random filters.

IV. Proposed Chaos Filters for CS

Though theoretical guarantees have not been provided in [6], empirical results there have shown that random filters can efficiently acquire different types of sparse signals and achieve exact reconstruction with high probability. In keeping up with the efficiency of random filters, we propose here to use chaos filters for compressed sensing that have the same structure as random filters, except that the filter tap values are generated by deterministic chaos instead of random variables.

A deterministic chaos denotes the irregular or chaotic motion that is generated by nonlinear systems whose dynamical laws uniquely determine the time evolution of a state of the system from a knowledge of its previous history. A simple chaos system is the Logistic map, which is defined by the following equation [9]:

$$h_L(n+1) = \alpha h_L(n)(1 - h_L(n)), \tag{5}$$

where $n = 0, \ldots, L-1$, and α is a control parameter. The required initial condition $h_L(0)$ is rather sensitive to the resulting behavior of the chaotic sequence. Note that for $h_L(n)$ to be chaotic, α must be equal to 4. Fig. 2(b) shows a particular logistic sequence, generated by (5) with $\alpha = 4$, $h_L(0) = 0.3$, $L = 128$, normalized to have zero mean and unity variance. Its distribution is non-Gaussian, as shown in Fig. 3(a).

To compare with the random filter, whose taps are Gaussian random variables, we would want to have a chaotic sequence that behaves Gaussian-like. One way to obtain such a sequence is to transform the Logistic map using the Logit Transform, to obtain the Gaussian-Logistic map $h_{GL}(n)$ as follows [9]:

$$h_{GL}(n) = \ln[h_L(n)/(1 - h_L(n)]. \tag{6}$$

The histogram of $h_{GL}(n)$ is shown in Fig. 3(b), resembling a Gaussian probability density function.

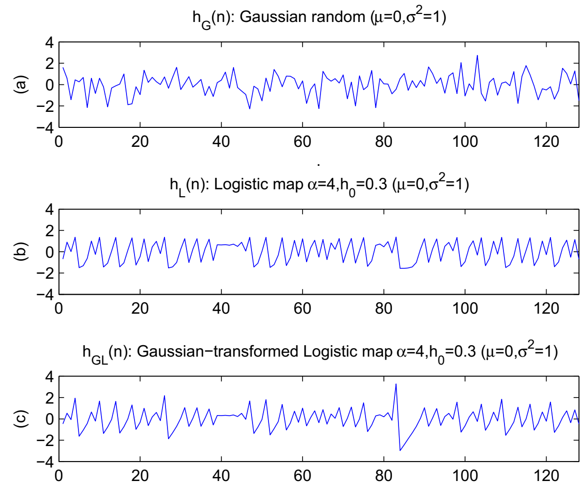

Fig. 2. Random and chaotic sequences: (a)– a Gaussian random sequence; (b)– Logistic map; (c)– Gaussian-Logistic map obtained by the logit transform. All sequences have been normalized to have zero mean and unit variance.

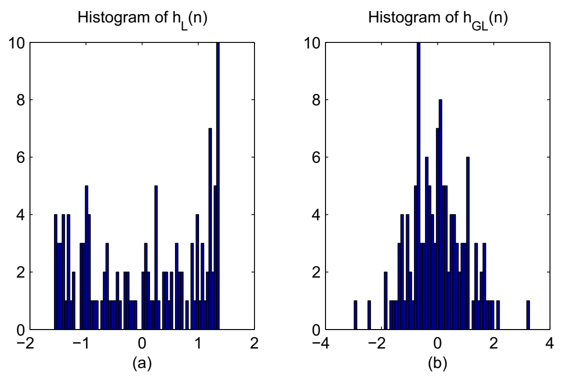

Fig. 3. Histogram of normalized Logistic Map, $h_L(n)$, and normalized Gaussian-like Logistic Map, $h_{GL}(n)$.

V. Numerical performance

We provide below some numerical examples to show, while comparing with random filters, the success of chaos filters for signal acquisition and reconstruction in compressed sensing.

A. Exact reconstruction using chaos filters

A discrete time-sparse signal $\mathbf{x}$ with $N = 512$ samples and $K = 10$ spikes is used, as shown in Fig. 4(a). Using a chaos filter of length $L = 128$, generated from the Gaussian-Logistic map and then normalized to have zero mean and unit variance, we obtained the chaotic measurements in Fig. 4(b) and the corresponding reconstructed signal using OMP in Fig. 4(c). Using a random filter of the same length, drawn from the Gaussian distribution with zero mean and unit variance, we achieved the measurements shown in Fig. 4(d) and the reconstructed signal in Fig. 4(e). In both cases, the same number of measurements was used, $M = 100$. From this illustrative simulation, it can be seen that, though the random measurements were different

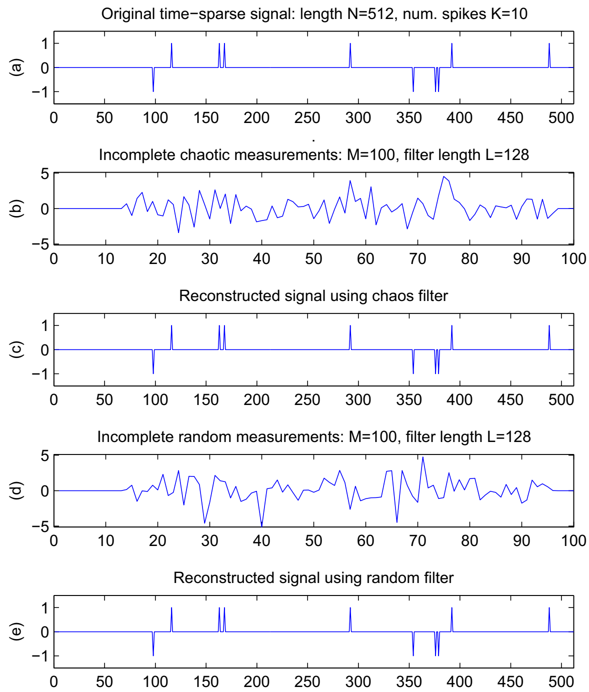

Fig. 4. Illustrative example showing successful reconstruction using chaos filters and random filters.

from the chaotic measurements, the signal reconstructed from the latter also resembles the original time-sparse signal.

For signal reconstruction using OMP in all the simulations in this paper, we used the algorithm `greed_omp` as part of the MATLAB package SPARSIFY (`http://www.dsp.ece.rice.edu/cs/`), with its parameters: `stopCrit='M'` and `stopTol=2*K`. Exact reconstruction was detected if the mean-squared error between original signal and reconstructed signal was less than 1.0E-16.

B. Random filters vs Chaos filters

Fig. 5 compares the performance between random filters and chaos filters, using 1000 Monte Carlo runs. At each run, a time-sparse signal was generated with $K = 20$ spikes, whose locations were also randomly generated. The tap values of the random filter were drawn from the Gaussian distribution with zero mean and unit variance, while those of the chaos filter were generated by the Gaussian-Logistic map, also with zero mean and unit variance. It can be seen that the chaos filter with the Gaussian-Logistic map performed better than the random filter for filter taps of $L = 64$, 128, or 256, in the sense that chaos filter required less measurements than random filters to achieve the same probability of exact reconstruction. Chaos filter with length $L = 32$ however dropped its performance as the number of measurements increased above 175. We do not

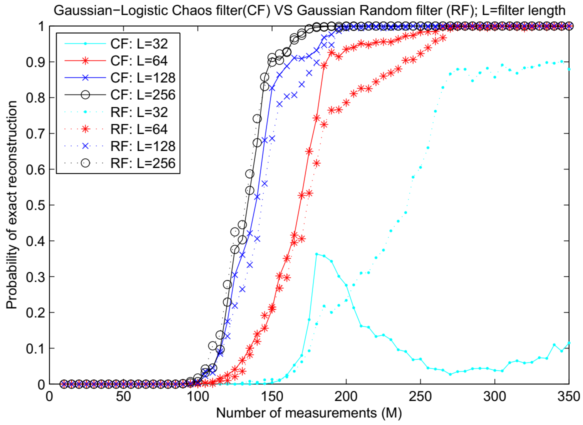

Fig. 5. Performance comparison: Gaussian-Logistic chaos filter vs Gaussian random filter.

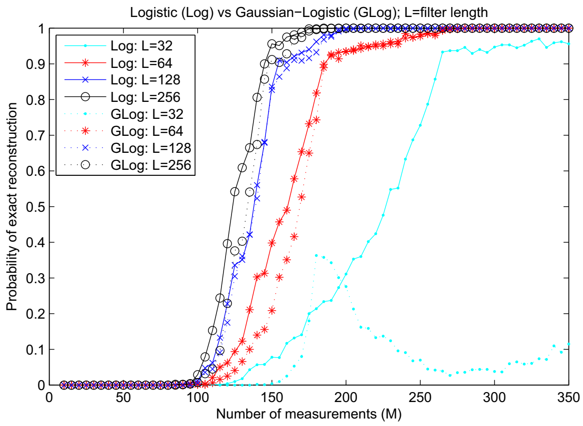

Fig. 6. Performance comparison: Chaos filter using Logistic map vs Gaussian-Logistic map.

know yet the cause of this drop in performance.

C. Logistic map vs Gaussian-Logistic map

Performance of the chaos filter with taps generated only by Logistic map was also compared with the chaos filter using Gaussian-Logistic previously obtained, as shown in Fig. 6. It is interesting to see that the use of the Logistic map lead to better performance than the use of Gaussian-Logistic map, while there was no drop in performance for the short filter, i.e. $L = 32$.

VI. Discussion

A. Convolution

There have been various ways of designing the measurement matrix $\mathbf{\Phi}$. Bajwa *et al.* in [10] used a partial Toeplitz matrix whose M rows are taken consecutively from the convolution matrix. Tropp *et al.* in [6] on the other hand obtained the matrix, which is $\mathbf{H}_\lambda$ in (4), by extracting rows that are equally separated by a factor of $\lambda = \lfloor N/m \rfloor$ through the downsampling operator. Recently, Romberg in [11] proposed to select the M rows randomly. Note that theoretical performance have been addressed in [10], [11]. It is by all means that one can apply similar structures for the measurement matrix using chaos convolution in the design of chaos filters for compressed sensing.

B. Types of chaos

In this paper, we only explored the use of a very simple chaotic sequence, the Logistic map, and its transformed version to have Gaussian-like behavior. In addition, we only used the initial condition $h_L(0) = 0.3$ in the performance study. We have neither addressed here other types of chaos nor exploited any properties of chaos, that are very rich in the literature. Study on various types of chaos and their parameters should be performed to find out good candidates, especially for signals with sparsity/compressibility in different domain: time, frequency, or wavelet.

C. Benefits of chaos

In the case of randomly generated filter coefficients, we have to essentially send all these coefficients to the decoder. In the case of chaotic generated filter coefficients, we only need to send the parameters of the chaos generators (initial condition, and control parameters) and the decoder can easily regenerate the coefficients. The use of chaos becomes more beneficial when adaptive design of the chaos filter is needed. One of such scenarios is the application of CS in ultra-wideband cognitive radios [3], in which we have to detect empty frequency bands (frequency holes) in the spectrum so as to initiate a communication on the frequency hole. Obviously, the sparsity of the spectrum varies from time to time, that is the number of frequency holes varies in time. Hence, it is likely that from time to time we have to work with different classes of signals; difference here is the amount of sparsity in the frequency domain. Therefore, we have to design the filters adaptively.

VII. Conclusion

This paper has shown by empirical performance that, using chaos filters, we can still acquire a time-sparse signal and further successfully reconstruct it from undersampled chaos measurements. In addition, chaos filters using the Logistic map outperforms chaos filters using the Gaussian-Logistic map as well as random filters using the Gaussian distribution. To the best knowledge of the authors, there has been no work done using chaos for compressed sensing. We would like to note here also a recent work by Saligrama in which a deterministic sequence is used [12].

ACKNOWLEDGEMENT

This article was funded in part by a grant from the Vietnam Education Foundation (VEF). The opinions, findings, and conclusions stated herein are those of the authors and do not necessarily reflect those of VEF.

REFERENCES

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

[2] D. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, April 2006.

[3] Z. Tian and G. Giannakis, "Compressed sensing for wideband cognitive radios," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2007*, vol. 4, 2007, pp. IV–1357–60.

[4] R. C. Hilborn, *Chaos and Nonlinear Dynamics: An Introduction for Scientists and Engineers*. Oxford University Press, 2000.

[5] P. Stavroulakis, Ed., *Chaos Applications in Telecommunications*. CRC Press, Taylor & Francis Group, 2006.

[6] J. Tropp, M. Wakin, M. Duarte, D. Baron, and R. Baraniuk, "Random filters for compressive sampling and reconstruction," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2006*, vol. 3, 2006, pp. III/872–875.

[7] E. Candès and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Problems*, vol. 23, no. 3, pp. 969–985, 2007.

[8] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[9] J. C. Sprott, *Chaos and Time-Series Analysis*. Oxford University Press, 2003.

[10] W. U. Bajwa, J. D. Haupt, G. M. Raz, S. J. Wright, and R. D. Nowak, "Toeplitz-structured compressed sensing matrices," in *Proc. IEEE/SP 14th Workshop on Statistical Signal Processing SSP '07*, 2007, pp. 294–298.

[11] J. Romberg, "Compressive sensing by random convolution," 2008. [Online]. Available: http://www.dsp.ece.rice.edu/cs/RandomConvolution.pdf

[12] V. Saligrama, "Deterministic designs with deterministic guarantees: Toeplitz compressed sensing matrices, sequence designs and system identification," 2008. [Online]. Available: http://arxiv.org/abs/0806.4958v2