Unlimited Sensing with Dictionary Sparsity

Morris-Luca Kühmeier
Dept. of Mathematics
Universität Innsbruck
6020 Innsbruck, Austria
Morris-Luca.Kuehmeier@uibk.ac.at

Karin Schnass
Dept. of Mathematics
Universität Innsbruck
6020 Innsbruck, Austria
Karin.Schnass@uibk.ac.at

Ayush Bhandari
Dept. of Electrical and Electronic Engg.
Imperial College London
SW7 2AZ, UK
A.Bhandari@imperial.ac.uk

***Abstract*—In conventional digital acquisition, a trade-off exists between allocating bits to capture a high dynamic range (HDR) signal and achieving high digital resolution (HDres). The Unlimited Sensing Framework (USF) overcomes this trade-off by introducing modulo non-linearity before sampling and encoding HDR signals as low dynamic range folded samples. These samples are then decoded or unfolded using smoothness or parametric signal priors. The question investigated in this paper is whether we can simplify the reconstruction if we assume that the signals are sparse in some transform domain or dictionary. The key insight there is that even though the modulo non-linearity disrupts the sparsity in the original dictionary the output signal exhibits sparsity in a new dictionary derived from the original. Capitalizing on this, we propose a strategy that leverages sparse recovery techniques, such as Orthogonal Matching Pursuit (OMP) and Basis Pursuit (BP), to directly reconstruct sparse signals from modulo samples. In addition to numerical simulations, hardware experiments with a modulo ADC demonstrate the stability and robustness of our approach in real-world scenarios. Our work thus opens up a new direction of exploration, leveraging methods from the established field of sparse approximation for the emerging topic of USF.**

I. Introduction

One of the major breakthroughs in signal processing over the past decade has been the effective harnessing of sparsity, [1], [2]. Concretely, the realisation that signals exhibit sparsity in particular transform domains or bases (*e.g.* wavelets [3], discrete cosine transform) or general dictionaries (*e.g.* Gabor frames [4]) has been widely exploited in various signal and image processing applications. This insight has enabled advances in areas including image denoising [5], audio [4], and seismic signal processing [6]. In addition to analytical dictionaries (*e.g.* wavelets, Gabor, DCT), algorithms for learning *sparsifying dictionaries*, [7]–[9], have also been developed, further expanding the applicability of sparsity based methods.

While sparsity offers a powerful signal model, digital acquisition of such signals often faces practical bottlenecks. For example, high dynamic range (HDR) signals can be clipped or saturated during digitization by analog-to-digital converters (ADCs). Additionally, real-world signals must undergo quantization [10], with the achievable digital resolution constrained by the bit budget and technology standards [11]. These limitations restrict the performance of ADCs and with a fixed bit budget there is typically a trade-off: one can optimize for either HDR capability or digital resolution, but not both simultaneously. Consequently, regardless of how advanced sparsity based methods are, their performance is ultimately tied to the quality of the digital sensing process.

From the acquisition perspective, the Unlimited Sensing Framework (USF) was recently introduced as a novel digital sensing method [12]–[14]. Similar to how sparsity based methods leverage non-linearity in algorithms, the USF harnesses non-linearity in the hardware. By applying modulo folding before sampling, the USF enables simultaneous HDR recovery while maintaining high digital resolution (HDRes). Real-world hardware experiments in applications such as tomography [15] and radars [16] have already demonstrated 10-12 dB improvement in digital resolution. USF-enabled MIMO systems have also made it possible to implement higher-order modulation schemes, such as 1024 QAM [17], which was previously challenging. In the recent work [18], hardware experiments have demonstrated recovery of HDR signals with a 60-fold increase in dynamic range.

Related Work. Since the USF relies on a co-design of hardware and algorithms, a key challenge is developing recovery methods [12]–[14] to efficiently and robustly [19] unfold the modulo folding operation. To address this, there are several reconstruction algorithms, focusing on various signal models, including *bandlimited* [12]–[14], [20], [21], *bandpass* [22], *smooth* [23], [24] and *parametric* [25], [26] or *compressive signals* with random projections [27]–[29].

Motivation. Despite the algorithmic progress since the inception of the USF [12], signals that exhibit dictionary sparsity (DS) have not yet been explored. This is likely due to (i) the recency of the USF where much of the foundational work is dedicated to Shannon-Nyquist like sampling strategy, (ii) the real-world performance assessment of the USF via modulo ADC hardware [14], and (iii) the non-triviality of inverting the folding non-linearity. However, given the *power* and *flexibility* of the sparse model, it is both relevant and timely to investigate the recovery of such signals within the USF.

Contributions. Our main contribution lies in the insight that the modulo representation itself is dictionary sparse. More importantly, the folded samples of an ambient signal, which is sparse in some dictionary, is then sparse in the concatenation

AB's work is supported by the European Research Council's Starting Grant for "CoSI-Fold" (101166158) and UK Research and Innovation council's FLF Program "Sensing Beyond Barriers via Non-Linearities" (MRC Fellowship award no. MR/Y003926/1).

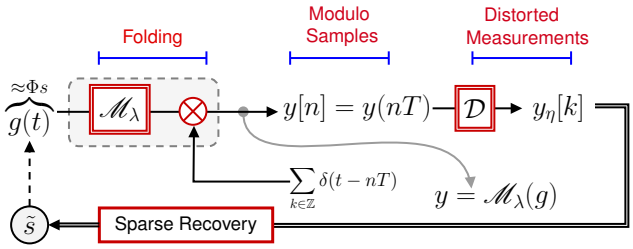

Fig. 1. Acquisition and Recovery Pipeline.

of two distinct dictionaries — one the transformed original dictionary $\mathbf{\Phi}$ and the other derived from the structure imposed by the folding architecture. In summary, we leverage sparse recovery techniques by associating the signal model and the folding architecture after the transform with an appropriate sparse model. We summarize our contributions as follows:
1) We propose a simple algorithm that uses well-known sparse recovery techniques, such as Orthogonal Matching Pursuit (OMP) and Basis Pursuit (BP). Our recovery method demonstrates flexibility across different dictionary structures. In this paper we achieve effective reconstruction of signals which are sparse in a DCT and Gabor-type dictionary.
2) Beyond numerical experiments we validate our method on hardware experiments in Section IV-B. These results demonstrate the empirical robustness of our approach, as it effectively handles quantization, non-idealities, and system noise, while also paving the way for practical applications.

II. Problem Setup

After introducing the input signal model and the acquisition pipeline that produces folded measurements, we provide a clear and concise statement of the problem.

Sparse Signal Model. We say that a signal $g \in \mathcal{H}$ where $\mathcal{H}$ is a Hilbert space such as $L^2(\mathbb{R})$ or $\mathbb{R}^N$, is (approximately) S-sparse in a dictionary $(\phi_\ell)_{\ell\in\Lambda}$, where $\phi_\ell \in \mathcal{H}$ and $\Lambda \neq \emptyset$, if there exists $I \subseteq \Lambda$ with $|I| = S$ and coefficients $(s_\ell)_{\ell\in I}$ with,

$$g = \sum_{\ell\in I} \phi_\ell s_\ell + e \quad \text{where} \quad \|e\|_{\mathcal{H}} \ll \|g\|_{\mathcal{H}}. \tag{1}$$

Setting $s_\ell = 0$ for $\ell \notin I$, we also use the shorthand $g = \Phi s + e$. If g is a continuous time signal, which is sparse in a finite continuous time dictionary, then its discretization $\boldsymbol{g}$ with sampling period T, defined via

$$\boldsymbol{g}[n] \stackrel{\text{def}}{=} g(n \cdot T) \quad \text{for} \quad n \in L \subseteq \mathbb{Z},$$

is sparse in the discretized dictionary $\mathbf{\Phi} = (\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_M)$ and we have $\boldsymbol{g} = \mathbf{\Phi} s + \mathbf{e}$. Most of the time we have either $L = \mathbb{Z}$ or $L = F_N \stackrel{\text{def}}{=} \mathbb{Z} \cap [-N/2 + 1, N/2]$ for $N \in 2\mathbb{Z}$, meaning $\boldsymbol{g} \in \mathbb{R}^N$ and $\mathbf{\Phi} \in \mathbb{R}^{N\times M}$.
Famous examples of dictionaries are wavelets, the Discrete Cosine Transform (DCT) and Gabor frames, which sparsely approximate music or natural images.

Unlimited Sensing Framework. Shannon's sampling theorem tells us that a bandlimited continuous-time signal, meaning $\operatorname{supp}(\hat{g}) \subseteq [-B, B]$, can be exactly recovered from its discretization if $L = \mathbb{Z}$ and $2TB \leq 1$, and approximately if $L = F_N$ and N large enough to capture most of the energy. In USF, instead of sampling g directly, it is first folded via modulo ADCs [14], [30]. For $\lambda \ll \|g\|_\infty = \sup_t |g(t)|$ the folding operator $\mathscr{M}_\lambda$ is implemented pointwise via the memoryless non-linearity defined as

$$\mathscr{M}_\lambda(r) = 2\lambda \left(\left[\!\left[\frac{r}{2\lambda} + \frac{1}{2} \right]\!\right] - \frac{1}{2} \right), \tag{2}$$

where $[\![r]\!] \stackrel{\text{def}}{=} r - \lfloor r \rfloor$ and $\lfloor r \rfloor = \sup\{ m \in \mathbb{Z} \,|\, m \leqslant r \}$. The folded signal $y = \mathscr{M}_\lambda(g)$ is subsequently sampled

$$\boldsymbol{y}[n] \stackrel{\text{def}}{=} y(nT) = (\mathscr{M}_\lambda(g))(nT) = \mathscr{M}_\lambda(\boldsymbol{g})[n], \tag{3}$$

as shown in Fig. 1. Note that sampling and folding commute. However in practice, such folded samples may be distorted with some noise η due to *non-idealities* [14], [30], *quantization* [13] and *system noise* [31]. We denote the distorted measurements by $\boldsymbol{y}_\eta \stackrel{\text{def}}{=} \boldsymbol{y} + \eta$. The task is to recover g from $\boldsymbol{y}$ or $\boldsymbol{y}_\eta$, which for a bandlimited signal g reduces to recovering $\boldsymbol{g}$ and using classical sampling theory.
Thus, we will next investigate if adding sparsity to the mix can lead to an efficient recovery scheme for $\boldsymbol{g}$ (thus g) from $\boldsymbol{y}_\eta$.

III. Recovering Sparse Signals

Let's assume that we have a signal g, that is exactly sparse in a finite dictionary and that there is no measurement noise. Our first observation is that $h = g - \mathscr{M}_\lambda(g)$ is a step function with a step for every folding action. This means that its discretization $\boldsymbol{h}$ is piecewise constant and the finite differences $\Delta\boldsymbol{h}$, where

$$\Delta\boldsymbol{h}[n] = \boldsymbol{h}[n] - \boldsymbol{h}[n-1] \quad \text{for } n \neq 0$$

(and $(\Delta\boldsymbol{h})[0] = \boldsymbol{h}[0] - \boldsymbol{h}[N]$), have a spike whenever two subsequent samples differ by a fold. Further if g is smooth then the number of folds should not be too high, meaning $\Delta\boldsymbol{h}$ is sparse in the Dirac dictionary $\Delta\boldsymbol{h} = \mathbf{\Phi}_\delta s_\delta$, where $\mathbf{\Phi}_\delta = \mathbb{I}_N$. Since the finite difference operator Δ is linear we get that

$$\Delta\boldsymbol{y} = \Delta(\boldsymbol{g} + \boldsymbol{h}) = \Delta(\mathbf{\Phi} s + \boldsymbol{h}) = (\Delta\mathbf{\Phi})s + \mathbf{\Phi}_\delta s_\delta$$

meaning $\Delta\boldsymbol{y}$ is sparse in the dictionary $(\Delta\mathbf{\Phi}, \mathbf{\Phi}_\delta)$ or rather $\hat{\mathbf{\Phi}} = (\Delta\mathbf{\Phi} D, \mathbf{\Phi}_\delta)$ with coefficients $\hat{s} = (s^t D, s_\delta^t)^t$ where D ensures normalisation of the atoms $\Delta\boldsymbol{\phi}_\ell$. This means that we can recover $\hat{s}$ and thus s as well as $\boldsymbol{g} = \mathbf{\Phi} s$ and $g = \sum_\ell \phi_\ell s_\ell$ via sparse approximation of $\Delta\boldsymbol{y}$ in $\hat{\mathbf{\Phi}}$. While any sparse approximation algorithm could be used, we use OMP, [32], and BP, [33], which have the advantage that we do not need to prescribe the correct sparsity level but can use an approximation error as stopping criterion. The proposed recovery strategy is summarized in Algorithm 1.

Before testing our idea on both synthetic and simulated hardware data, let us consider which requirements we can expect for our strategy to be successful, [34]–[36].
First, for OMP and BP to have a good chance of recovering a k-sparse $\hat{s}$ from $\Delta\boldsymbol{y}$ we need the coherence of $\hat{\mathbf{\Phi}}$ to be small,

$$\mu \stackrel{\text{def}}{=} \max_{j\neq k} |\langle \hat{\boldsymbol{\phi}}_j, \hat{\boldsymbol{\phi}}_k \rangle| \ll 1. \tag{4}$$

Algorithm 1 Signal Recovery Algorithm

Input: Folded sampled signal $\boldsymbol{y} \in \mathbb{R}^N$, sampled dictionary $\boldsymbol{\Phi} \in \mathbb{R}^{N\times M}$ and error threshold ε
Apply Δ and normalize by its column norms: $\Delta\boldsymbol{\Phi}\mathbf{D}^{-1}$
Concatenate dictionaries: $\hat{\boldsymbol{\Phi}} = [\Delta\boldsymbol{\Phi}\mathbf{D}^{-1}, \boldsymbol{\Phi}_\delta] \in \mathbb{R}^{N\times(M+N)}$
Recover $\hat{s}$ using OMP or BP:
$\hat{s} = \text{OMP}(\Delta\boldsymbol{y}, \hat{\boldsymbol{\Phi}}, \varepsilon)$ / $\hat{s} = \text{BP}(\Delta\boldsymbol{y}, \hat{\boldsymbol{\Phi}}, \varepsilon)$
$s_{\mathbf{D}} = \hat{s}(1:M)$ and $\tilde{s} = \mathbf{D}^{-1}s_{\mathbf{D}}$
Recover sampled signal: $\tilde{\boldsymbol{g}} = \boldsymbol{\Phi}\tilde{s}$
Output: $\tilde{s}, \tilde{\boldsymbol{g}}$

Algorithm 2 Orthogonal Matching Pursuit (OMP)

Input: Sampled Signal $\boldsymbol{y} \in \mathbb{R}^N$, sampled dictionary $\boldsymbol{\Phi} \in \mathbb{R}^{N\times M}$, error threshold ε
Initialize $\hat{s} = 0, J = \emptyset$ and $\mathbf{r}_J = \boldsymbol{y}$
while $\|\mathbf{r_J}\| > \varepsilon \cdot \|\boldsymbol{y}\|_2$ **do**
  $j = \text{argmax}_k \, |\langle \phi_k, \mathbf{r}_J \rangle|$
  $J \leftarrow J \cup \{j\}$
  $\hat{s}_J = \boldsymbol{\Phi}_J^\dagger \boldsymbol{y}$
  $\mathbf{r}_J = \boldsymbol{y} - \boldsymbol{\Phi}s$
end while
Output: $\hat{s}$

Algorithm 3 Basis Pursuit (BP)

Input: Sampled Signal $\boldsymbol{y} \in \mathbb{R}^N$, sampled dictionary $\boldsymbol{\Phi} \in \mathbb{R}^{N\times M}$, error threshold ε
Construct matrix: $\mathbf{A} = [\boldsymbol{\Phi}, -\boldsymbol{\Phi}]$
Solve linear program: $\min_{s_+\geq 0} \|\mathbf{A}s_+ - \boldsymbol{y}\| \leq \varepsilon$
Reconstruct signed coefficients: $\hat{s} = s_+(1:M) - s_+(M+1:\text{end})$
Output: $\hat{s}$

This means that on one hand we need,

$$\max_{j,n} \frac{|\Delta\boldsymbol{\phi}_j[n]|}{\|\Delta\boldsymbol{\phi}_j\|_2} \ll 1, \tag{5}$$

so the sampled atoms should not have big jumps, meaning the continuous atoms should be smooth, and on the other that

$$\max_{j\neq k} \frac{|\langle \Delta\boldsymbol{\phi}_j, \Delta\boldsymbol{\phi}_k \rangle|}{\|\Delta\boldsymbol{\phi}_j\|_2 \|\Delta\boldsymbol{\phi}_k\|_2} \ll 1, \tag{6}$$

so the sampled atoms should not disappear or coincide, meaning the continuous atoms should not be too smooth or similar.

Second, we also need $\hat{s}$ not to have too many non-zero entries. While the non-zero entries of s or Ds are fixed, the ones of s_δ depends on the number of folds meaning the size of λ compared to $\|g\|_\infty$.

IV. Experiments

We first explore the limits of folding, subsampling and recovery by Algorithm 1 by conducting experiments on S-sparse synthetic signals. In addition we test the robustness of the Algorithm 1 on real-world data, by recovering signals from the modulo ADC, which are only approximately S-sparse in a DCT and Gabor-type dictionary.

A. Synthetic Experiments.

We generate a Discrete Cosine Transformation (DCT) dictionary and two Gabor-type dictionaries as follows:

The DCT atoms are constructed in $\mathbb{R}^N$ with $N = 1024$ as

$$\boldsymbol{\phi}_m[n] \stackrel{\text{def}}{=} \sqrt{\frac{2}{N}} \cos\left(\frac{\pi(2n+1)m}{2N}\right), \tag{7}$$

where we restrict m to be in $\{64, 65 \ldots, 575\}$. This is to avoid both too smooth and not smooth enough atoms as discussed in (5) and (6).

The two Gabor-type dictionaries are generated in $\mathbb{R}^{N\times M}$ with dimension $N = 1080$ and $M \in \{675, 945\}$ in the following way. We use a Gaussian window $\gamma_N(t) = \exp\left(-\pi t^2/N\right)$ which is then discretized to $\boldsymbol{\gamma}_N[n]$ with sampling period 1 and $n \in F_N = \mathbb{Z} \cap [-N/2+1, N/2]$. For $k, \ell \in F_N$ we define the time-shift operators with stride $a = 24$ and frequency-shift operators with stride $b \in \{18, 24\}$ via

$$T_{ak}\boldsymbol{g}[n] \stackrel{\text{def}}{=} \boldsymbol{g}[n - ak \mid \text{mod } N]$$
$$M_{b\ell}\boldsymbol{g}[n] \stackrel{\text{def}}{=} \boldsymbol{g}[n] \cdot \exp(2\pi i b\ell(n-1)/N).$$

This yields our Gabor-type atoms $\boldsymbol{\phi}_{(k,\ell)} = \text{real}(T_{ak}M_{b\ell}\boldsymbol{\gamma}_N) + \text{imag}(T_{ak}M_{b\ell}\boldsymbol{\gamma}_N)$. Because of the Gaussian window γ_N we do not have too smooth atoms, as discussed in (6), and because a and b are large enough they are not too similar. However, we want to avoid non-smooth atoms, as discussed in (5), and therefore restrict ℓ to $\ell \in \{-10, -9, \ldots, 10\}$ if $b = 18$ and $\ell \in \{-7, -6, \ldots, 7\}$ if $b = 24$.

Set up: We next synthesize S-sparse signals mimicking our sampled discretized time signals as linear combinations of S such atoms according to our signal model in (1), where we chose the index set I with $|I| = S$ uniformly at random among all sets of size S and the non-zero coefficients i.i.d Gaussian, i.e. $s_j \sim \mathcal{N}(0,1)$. Each signal $\boldsymbol{g} = \boldsymbol{\Phi}s$ is folded, as in (2), leading to $\boldsymbol{y} = \mathscr{M}_\lambda(\boldsymbol{g})$. For DCT signals, we vary the sparsity level S from 10 to 100 in steps of 10, and for Gabor-type signals, from 4 to 40 in steps of 4. For each sparsity level S we generate 1000 signals $\boldsymbol{g}$ and fold them with λ ranging from 0.3 to 0.48 for DCT signals and from 0.36 to 0.54 for Gabor-type signals, both with a step size of 0.02. We then run Algorithm 1 with error threshold $\varepsilon = 0.01$ and dictionary $\hat{\boldsymbol{\Phi}} = (\Delta\boldsymbol{\Phi}D, \boldsymbol{\Phi}_\delta)$. We obtain $\tilde{s}$, reconstruct $\tilde{\boldsymbol{g}} = \boldsymbol{\Phi}\tilde{s}$, and calculate the relative reconstruction error $\|\boldsymbol{g} - \tilde{\boldsymbol{g}}\|_2/\|\boldsymbol{g}\|_2$. Figure 2 provides an overview of the recovery performance. For the DCT dictionary (first column) and the Gabor-type dictionary with strides $b = 18$ and $b = 24$ (second and third column), the first row shows the average total sparsity level $\|s\|_0 + \|s_\delta\|_0$ of the folded signals for each folding height λ and sparsity level S. As expected, we can see that the total sparsity level increases when the folding height λ decreases. In the second and third rows, we show for each folding height λ and sparsity level S the corresponding

average error of Algorithm 1 using OMP and BP, respectively. We can see that, consistent with the total sparsity of the folded signals in the first row, the average error remains low when the sparsity level is small and the folding height is large. We observe a sharp transition from small to large error for the DCT signals, while this transition is more gradual for the Gabor-type signals. Moreover, the Gabor-type dictionary with a larger stride ($b = 24$) performs better compared to the one with smaller stride ($b = 18$). Both effects can be explained by the larger coherence of the recovery dictionary used for Gabor-type signals, which is 0.59 for the one with stride $b = 24$ and 0.77 for the one with stride $b = 18$, compared to 0.40 for the DCT recovery dictionary. Due to the higher similarities of atoms, there is a greater risk for OMP and BP to confuse atoms, leading to higher recovery errors. The fourth and fifth rows show the percentage of successfully recovered signals for each folding height λ and sparsity level S, where we declare success if the error is below 0.05. We observe that DCT signals are more resilient to folds than Gabor-type signals, which can again be explained by the lower coherence. In particular, Algorithm 1 successfully recovers folded DCT signals up to a total sparsity level of 200, whereas for folded Gabor-type signals, the limit is around 50. Finally, we observe that OMP outperforms BP for DCT signals, while BP is more effective for Gabor-type dictionaries. This can again be explained by the larger coherence, to which BP is quite robust, while the greedy algorithm OMP will start to pick wrong atoms that once chosen cannot be removed.

B. Hardware Experiments.

The goal of our hardware experiment is to show stability of our algorithm in the presence of quantization, non-idealities, and system noise. We construct a 10-sparse signal in the DCT and the Gabor-type dictionary as described in the last section, but in higher dimensions. So for the DCT dictionary we now set $N = 2^{15}$ and keep the same $\boldsymbol{\phi}_m$. For the Gabor-type dictionary we set $N = 32400$, $a = 720$ and $b \in \{18, 24\}$. We take all possible time-shifts and frequency-shifts with $\ell \in \{-10, \dots, 10\}$ if $b = 18$ and $\ell \in \{-7, \dots, 7\}$ if $b = 24$. The generated signals serve as templates for a waveform generator, that provides the corresponding continuous time signals. In our setup, we observed that the waveform generator introduces a small DC bias, meaning it adds a constant, that we assumed to be known. The modulo ADC discretizes the 10-sparse signals g to dimension $\widetilde{N} = 1024$ for DCT and dimension $\widetilde{N} = 1080$ for Gabor and folds them, which leads to $\boldsymbol{y}_\eta = \mathscr{M}_\lambda(g) + \eta$. For reconstruction, we use the downsampled dictionaries above with strides of $T = 32$ for DCT and $T = 30$ for Gabor. Algorithm 1 is then applied to $\boldsymbol{y}_\eta$ using OMP with an error threshold $\varepsilon = 0.01$ to recover $\tilde{\boldsymbol{g}}$. After correcting for bias, we obtain a relative reconstruction error of 0.03 for the DCT signal compared to the discretized signal $\boldsymbol{g}$. This is a remarkable result, considering that the DCT signal underwent 285 folds and was folded up to three times in amplitude, meaning $\|\boldsymbol{g}\|_\infty \geq 5\lambda$. Similarly, for the Gabor-type signal, which underwent 34 folds and was folded up to twice in amplitude, Algorithm 1 using OMP and the Gabor-type dictionary with stride $b = 24$ successfully recovered the signal with a relative error of 0.03. For the second Gabor-type signal, which underwent 28 folds and was also folded up to two times in amplitude, Algorithm 1 using OMP and the Gabor-type dictionary with stride $b = 18$ achieved a relative error of 0.05. Figure 3 displays the discrete signal, the reconstructed signal and the sensed signal of the DCT signal and Gabor-type signal with frequency stride $b = 18$.

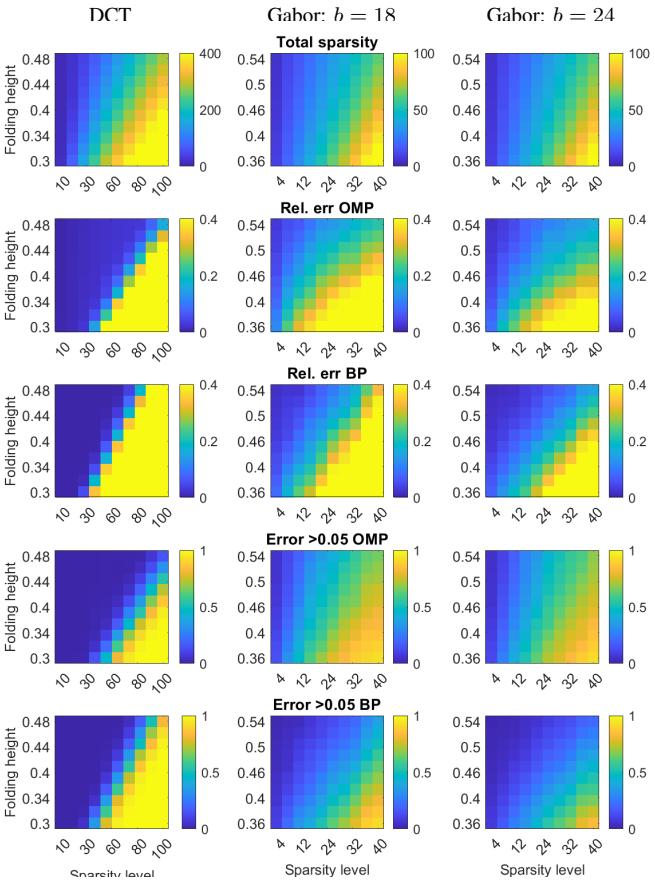

Fig. 2. For the DCT dictionary (first column) and the Gabor-type dictionaries with stride $b = 18$ and $b = 24$ (second and third column) the rows show (top to bottom): average total sparsity level of the folded signals, average relative error using OMP and BP, percentage of signals with relative error above 0.05 using OMP and BP.[1]

V. Conclusion & Future Work

In this work, we have shown that sparse recovery techniques can successfully reconstruct signals despite the challenges introduced by folding, making them a promising tool in the setting of USF. Additionally, we demonstrate that our approach to recovering folded signals performs well using out-of-the-box recovery algorithms such as Orthogonal Matching Pursuit (OMP) and Basis Pursuit (BP), even without any modifications tailored to the folding structure. Motivated by

[1]The simulations underlying this figure and the figure itself can be reproduced using the code available at: https://github.com/Morris-Luca/Unlimited_Sensing_with_Dictionary_Sparsity

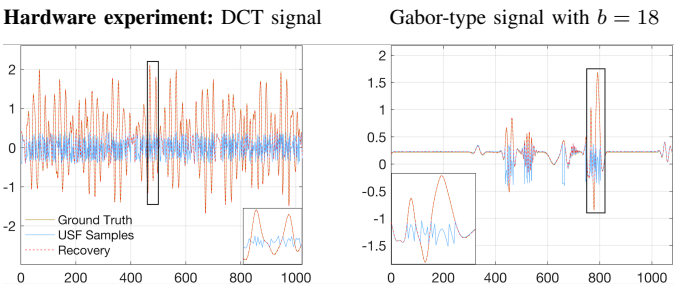

Fig. 3. The left plot displays the 10-sparse DCT signal, while the right plot shows the 10-sparse Gabor-type signal with frequency stride $b = 18$. Each plot presents the ground truth signal (yellow), the folded signal from the modulo ADC (blue), and the recovered signal (red).
A larger version of the image can be viewed at: https://github.com/Morris-Luca/Unlimited_Sensing_with_Dictionary_Sparsity

this performance, our next goals are to refine the recovery process to account for the folding structure and to incorporate side constraints for the Dirac dictionary $\mathbf{\Phi}_\delta$, such as using atoms only if their response is close to the spike height of 2λ. Moreover, we want to exploit the symmetry of the finite differences, which leads to an equal number of upward and downward spikes, implying that the number of positive coefficients in the Dirac dictionary must match the number of negative coefficients. Finally, we aim to further analyze the recovery dictionary: To avoid unintentionally favoring high-frequency atoms – since they exhibit larger amplitudes after applying finite differences – we currently normalize the atoms. However, based on the insights from [37], we propose replacing normalization with multiplication by a weight matrix. This approach allows us to adapt to the given data and the fact that some atoms, e.g. low frequency atoms, are used more often.

REFERENCES

[1] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ_1 minimization," *Proc. Natl. Acad. Sci.*, vol. 100, pp. 2197–2202, Feb. 2003.

[2] R. Rubinstein, A. M. Bruckstein, and M. Elad, "Dictionaries for sparse representation modeling," *Proc. IEEE*, vol. 98, pp. 1045–1057, June 2010.

[3] B. Ophir, M. Lustig, and M. Elad, "Multi-scale dictionary learning using wavelets," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, pp. 1014–1024, Sept. 2011.

[4] R. Gribonval and E. Bacry, "Harmonic decomposition of audio signals with matching pursuit," *IEEE Trans. Signal Process.*, vol. 51, pp. 101–111, Jan. 2003.

[5] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, pp. 2080–2095, Aug. 2007.

[6] F. J. Herrmann, "Randomized sampling and sparsity: Getting more information from fewer samples," *Geophysics*, vol. 75, pp. WB173–WB187, Nov. 2010.

[7] K. Engan, S. Aase, and J. Husoy, "Method of optimal directions for frame design," in *ICASSP99*, vol. 5, pp. 2443 – 2446, 1999.

[8] K. Kreutz-Delgado, J. F. Murray, B. D. Rao, K. Engan, T.-W. Lee, and T. J. Sejnowski, "Dictionary learning algorithms for sparse representation," *Neural Computation*, vol. 15, pp. 349–396, Feb. 2003.

[9] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing.*, vol. 54, pp. 4311–4322, Nov. 2006.

[10] R. M. Gray and D. L. Neuhoff, "Quantization," *IEEE Trans. Inf. Theory*, vol. 44, pp. 2325–2383, Oct. 1998.

[11] R. Walden, "Analog-to-digital converter survey and analysis," *IEEE J. Sel. Areas Commun.*, vol. 17, pp. 539–550, Apr. 1999.

[12] A. Bhandari, F. Krahmer, and R. Raskar, "On unlimited sampling," in *SampTA17*, July 2017.

[13] A. Bhandari, F. Krahmer, and R. Raskar, "On unlimited sampling and reconstruction," *IEEE Trans. Signal Process.*, vol. 69, pp. 3827–3839, Dec. 2020.

[14] A. Bhandari, F. Krahmer, and T. Poskitt, "Unlimited sampling from theory to practice: Fourier-Prony recovery and prototype ADC," *IEEE Trans. Signal Process.*, pp. 1131–1141, Sept. 2021.

[15] M. Beckmann, A. Bhandari, and M. Iske, "Fourier-domain inversion for the modulo Radon transform," *IEEE Transactions on Computational Imaging*, vol. 10, pp. 653–665, Apr. 2024.

[16] T. Feuillen, B. Shankar, and A. Bhandari, "Unlimited sampling radar: Life below the quantization noise," in *ICASSP23*, IEEE, June 2023.

[17] Z. Liu, A. Bhandari, and B. Clerckx, "λ-MIMO: Massive MIMO via modulo sampling," *IEEE Trans. Commun.*, pp. 1–12, 2023.

[18] Y. Zhu and A. Bhandari, "Unleashing dynamic range and resolution in USF via novel hardware," in *IEEE Sensors*, 2024.

[19] R. Guo and A. Bhandari, "ITER-SIS: Robust unlimited sampling via iterative signal sieving," in *ICASSP23*, IEEE, June 2023.

[20] E. Azar, S. Mulleti, and Y. C. Eldar, "Residual recovery algorithm for modulo sampling," in *ICASSP22*, IEEE, May 2022.

[21] E. Romanov and O. Ordentlich, "Above the Nyquist rate, modulo folding does not hurt," *IEEE Signal Processing Letters*, vol. 26, pp. 1167–1171, Aug. 2019.

[22] G. Shtendel, D. Florescu, and A. Bhandari, "Unlimited sampling of bandpass signals: Computational demodulation via undersampling," *IEEE Trans. Signal Process.*, pp. 1–12, Sept. 2023.

[23] A. Bhandari and F. Krahmer, "HDR imaging from quantization noise," in *ICIP20*, pp. 101–105, Oct. 2020.

[24] S. Rudresh, A. Adiga, B. A. Shenoy, and C. S. Seelamantula, "Wavelet-based reconstruction for unlimited sampling," in *ICASSP20*, pp. 4584–4588, Apr. 2018.

[25] A. Bhandari, F. Krahmer, and R. Raskar, "Unlimited sampling of sparse signals," in *ICASSP18*, Apr. 2018.

[26] A. Bhandari, "Back in the US-SR: Unlimited sampling and sparse super-resolution with its hardware validation," *IEEE Signal Process. Lett.*, vol. 29, pp. 1047–1051, Mar. 2022.

[27] O. Musa, P. Jung, and N. Goertz, "Generalized approximate message passing for unlimited sampling of sparse signals," in *GlobalSIP18*, IEEE, Nov. 2018.

[28] D. Prasanna, C. Sriram, and C. R. Murthy, "On the identifiability of sparse vectors from modulo compressed sensing measurements," *IEEE Signal Process. Lett.*, vol. 28, pp. 131–134, Jan. 2021.

[29] V. Shah and C. Hegde, "Sparse signal recovery from modulo observations," *EURASIP J. on Adv. in Sig. Proc.*, vol. 2021, Apr. 2021.

[30] D. Florescu, F. Krahmer, and A. Bhandari, "The surprising benefits of hysteresis in unlimited sampling: Theory, algorithms and experiments," *IEEE Trans. Signal Process.*, vol. 70, pp. 616–630, Jan. 2022.

[31] A. Bhandari, "Unlimited sampling with sparse outliers: Experiments with impulsive and jump or reset noise," in *ICASSP22*, pp. 5403–5407, May 2022.

[32] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal Matching Pursuit: recursive function approximation with application to wavelet decomposition," in *Asilomar Conf. on Signals Systems and Comput.*, 1993.

[33] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, 1998.

[34] J. Tropp, "Greed is good: Algorithmic results for sparse approximation.," *IEEE Transactions on Information Theory*, vol. 50, pp. 2231–2242, Oct. 2004.

[35] R. Gribonval and P. Vandergheynst, "On the exponential convergence of matching pursuits in quasi-incoherent dictionaries," *IEEE Transactions on Information Theory*, vol. 52, no. 1, pp. 255–261, 2006.

[36] K. Schnass, "Average performance of Orthogonal Matching Pursuit (OMP) for sparse approximation," *IEEE Signal Processing Letters*, vol. 25, no. 12, pp. 1865–1869, 2018.

[37] K. Schnass and P. Vandergheynst, "Dictionary preconditioning for greedy algorithms," *IEEE Transactions on Signal Processing*, vol. 56, no. 5, pp. 1994–2002, 2008.