Random Gabor Multipliers for Compressive Sensing: A Simulation Study

Shristi Rajbamshi, Georg Tauböck, Peter Balazs, and Luís Daniel Abreu

Acoustics Research Institute
Austrian Academy of Sciences, 1040 Vienna, Austria
shristi.rajbamshi@oeaw.ac.at

***Abstract*—In this paper, we analyze by means of simulations the applicability of random Gabor multipliers as compressive measurements. In particular, we consider signals that are sparse with respect to Fourier or Gabor dictionaries, i.e., signals that are sparse in frequency or time-frequency domains. This work is an extension of our earlier contribution, where we introduced random Gabor multipliers to compress signals that are sparse in time domain. As reconstruction technique we employ the well known ℓ_1–minimization procedure. Finally, we evaluate the compression performance of random Gabor multipliers by applying them to a specific audio signal with inherent time-frequency sparsity. Our results highlight the strong potential of random Gabor multipliers for present and future real-world audio applications.**

***Index Terms*—Compressive Sensing, Gabor Multiplier, Random Matrix, Dictionary, Audio**

I. Introduction

Compressive Sensing (CS) is a methodology to acquire and reconstruct signals with a rate lower than what is stated by the classical sampling theorem under the prior condition that the signals are S-sparse or compressible in some domain [1]–[3]. S-sparse means that the signal contains at most S non-zero entries whereas compressibility imposes a sufficiently strong energy decay of the entries.

One of the major challenges in CS is the choice of the measurement matrices, which could be either random [4] or deterministic [5]. On one hand, deterministic matrices are preferred as they allow for fast implementations with reduced storage requirements for practical purposes. But on the other hand, the usage of deterministic matrices is often avoided as deterministic matrix constructions are known to suffer from the *square-root bottleneck* [6]. Therefore, recent studies in this field have been focused on *structured* random measurement matrices. These matrices are optimal in the sense that they have enough structure to facilitate fast computations while still being sufficiently random to provide reliable measurements for reconstruction. Some instances of such structured random matrices are random partial Fourier matrices [2], [3], [7], partial random circulant matrices (subsampled random convolutions) [8]–[12], time-frequency structured random matrices [11], [13], [14], and, very recently, random Gabor multiplier matrices [15]. In [15], it was theoretically shown that random Gabor multipliers [16] exhibit essentially optimal compression capabilities of time-domain sparse signals and that ℓ_1–minimization is very efficient in reconstructing them in practice.

In the present work, we will concentrate on signals that have sparse representations in domains other than time. To illustrate the practical importance of our results, we apply our methods to a real-world audio signal. It is demonstrated that random Gabor multipliers can be used to exploit the inherent time-frequency sparsity of the signal in order to reduce the number of measurements.

A. Notation

In the following, we will denote matrices with uppercase letters, such as M, and column vectors as lowercase letters, such as m. We will write variables as lowercase letters, e.g., s and constants as uppercase letters, e.g., S. The superscripts $^\mathsf{T}$ and * will denote transpose and conjugate transpose, respectively, overlined scalars will be complex conjugated, and $\langle\cdot,\cdot\rangle$ will denote the inner product of two vectors.

For a vector $\mathrm{x} = [x_0, x_1, \cdots, x_{L-1}]^\mathsf{T}$, the ℓ_0 "norm" is defined as $\|\mathrm{x}\|_0 = \#\{l : x_l \neq 0\}$ and the ℓ_1 norm is defined as $\|\mathrm{x}\|_1 = \sum_{l=0}^{L-1} |x_l|$. With $[\,\cdot\,]_L \triangleq [\,\cdot \mod L]$ we abbreviate the modulo-L operation due to circular indexing.

The rest of the paper will be organized as follows: In Section II, we will briefly recall the basic idea of compressed sensing. In Section III, we will shortly describe the concept of Gabor multipliers. Section IV will contain results from our numerical experiments and Section V will be devoted to concluding remarks.

II. Compressive Sensing in a Nutshell

Let us start by considering a signal vector $\mathrm{x} \in \mathbb{C}^L$, a dictionary $\Phi \in \mathbb{C}^{L\times Q}$, and assume that the signal x can be expressed as

$$\mathrm{x} = \Phi\alpha, \tag{1}$$

where $\alpha \in \mathbb{C}^Q$ is the coefficient vector.

The goal of CS is to recover x from a small number of linear measurements $\mathrm{y} \in \mathbb{C}^K$ $(K \ll L)$ given as,

$$\mathrm{y} \quad = \Psi\mathrm{x}, \tag{2}$$

where $\Psi \in \mathbb{C}^{K\times L}$ is the measurement matrix.

Hence, by (1) and (2),

$$\mathrm{y} \quad = \Psi\Phi\alpha, \quad = \mathrm{A}\alpha \tag{3}$$

where A will be referred to as sensing matrix.

Generally, it is impossible to uniquely recover x from y without additional constraints because (2) represents an underdetermined system of equations and an infinite number

978-9-0827-9703-9/19/$31.00 ©2019 IEEE

of solutions exist. However, the scenario is quite different, if x has a sparse representation in the dictionary Φ, i.e., $\|\alpha\|_0 \leq S$ [17]. Then, to reconstruct x from y, we can apply the following two steps.

In the first step, we aim to recover the sparse coefficient vector α. Theoretically, this can be done by solving the ℓ_0–minimization problem expressed as [18],

$$\hat{\alpha} = \arg\min_{\alpha} \|\alpha\|_0 \quad \text{subject to} \quad \mathrm{y} = \mathrm{A}\alpha. \tag{4}$$

However, since ℓ_0–minimization is an NP-hard problem, its convex relaxation, ℓ_1–minimization, is solved instead [7], [19], [20],

$$\hat{\alpha} = \arg\min_{\alpha} \|\alpha\|_1 \quad \text{subject to} \quad \mathrm{y} = \mathrm{A}\alpha. \tag{5}$$

The second step simply consists of the computation of $\hat{\mathrm{x}}$ from $\hat{\alpha}$ according to (1), i.e.,

$$\hat{\mathrm{x}} = \Phi\hat{\alpha}. \tag{6}$$

Note that the choice of the dictionary Φ directly influences the sparsity of the coefficient vector, which then automatically impacts the number of measurements required to reconstruct the signal. Furthermore, success of (5) is known to rely on properties of A, and again, via (3), on Φ. This means, that the observed reconstruction performance will crucially depend on the chosen sparse representation.

III. Compressed Sensing using Random Gabor Multipliers

In this section, we will recall the definitions of Gabor systems as well as of Gabor multipliers and we will describe how to use random Gabor multipliers as CS measurement matrices.

A. Gabor Systems

A discrete *Gabor system* [21] (γ, a, b) is defined as a collection of time-frequency shifts of the vector $\gamma \in \mathbb{R}^L$ expressed as,

$$\gamma_{n,m}[l] \triangleq \gamma[l - na]_L e^{2\pi i m b l/L}, \quad l \in \{0, \dots, L-1\}, \tag{7}$$

where $a, b \in \{0, \dots, L-1\}$ denote time and frequency parameters, respectively, and $n \in \{0, \dots, N-1\}$ chosen such that $N = L/a \in \mathbb{N}$ and $m \in \{0, \dots, M-1\}$ chosen such that $M = L/b \in \mathbb{N}$ denote the time and frequency shift indices, respectively. In general, the values of the parameters a and b are chosen to be greater than 1. But when $a = b = 1$, then (7) is referred to as full Gabor system.

The system (γ, a, b) according to (7) contains $P = MN$ elements. Note that the redundancy R of this system is defined as $R = P/L = MN/L = M/a$.

B. Gabor Analysis and Synthesis Operator

We can associate two operators to a Gabor system (γ, a, b). Let $\mathrm{A}_\gamma : \mathbb{C}^L \to \mathbb{C}^{L/b \times L/a}$ denote the *Gabor analysis operator*[1] that computes for any given signal $\mathrm{x} \in \mathbb{C}^L$,

$$\mathrm{c}_{m,n} = \sum_{l=0}^{L-1} \mathrm{x}[l]\overline{\gamma[l - na]_L} e^{-2\pi i m b l/L}, \tag{8}$$

where $m \in \{0, \dots, L/b-1\}$ and $n \in \{0, \dots, L/a-1\}$.

Furthermore, let $\mathrm{S}_\gamma : \mathbb{C}^{L/b \times L/a} \to \mathbb{C}^L$ denote the *Gabor synthesis operator* that computes a signal $\mathrm{z} \in \mathbb{C}^L$ according to

$$\mathrm{z}[l] = \sum_{n=0}^{N-1}\sum_{m=0}^{M-1} \mathrm{c}_{m,n}\gamma[l - na]_L e^{2\pi i m b l/L}, \tag{9}$$

where $l \in \{0, \dots, L-1\}$.

C. Gabor Multipliers

Let us consider two Gabor systems $(\mathrm{g}, a_\mathrm{g}, b_\mathrm{g})$ and $(\mathrm{h}, a_\mathrm{h}, b_\mathrm{h})$, where $\mathrm{g} \in \mathbb{R}^L$ and $\mathrm{h} \in \mathbb{R}^K$, and assume that both systems contain the same number P of elements. Clearly,

- redundancy of $(\mathrm{g}, a_\mathrm{g}, b_\mathrm{g})$ is $R_\mathrm{g} = P/L = L/a_\mathrm{g}b_\mathrm{g}$,
- redundancy of $(\mathrm{h}, a_\mathrm{h}, b_\mathrm{h})$ is $R_\mathrm{h} = P/K = K/a_\mathrm{h}b_\mathrm{h}$.

A *Gabor multiplier* [16] is defined as the linear operator[2] taking the form: Gabor analysis with respect to $(\mathrm{g}, a_\mathrm{g}, b_\mathrm{g})$, proceeded by point-wise multiplication with the so-called symbol, and then followed by Gabor synthesis with respect to $(\mathrm{h}, a_\mathrm{h}, b_\mathrm{h})$.

Hence, the *Gabor multiplier* operator $\mathrm{M}_{\mathrm{s},\mathrm{h},\mathrm{g}} : \mathbb{C}^L \to \mathbb{C}^K$ can be expressed as,

$$\mathrm{y}[k] = \sum_{n=0}^{N-1}\sum_{m=0}^{M-1} \mathrm{s}_{m,n}\mathrm{c}_{m,n}\mathrm{h}[k - na_\mathrm{h}]_K e^{2\pi i m b_\mathrm{h} k/K}, \tag{10}$$

where $k \in \{0, \dots, K-1\}$, $\mathrm{s}_{m,n}$ denotes the symbol and, cf. (8),

$$\mathrm{c}_{m,n} = \sum_{l=0}^{L-1} \mathrm{x}[l]\overline{\mathrm{g}[l - na_\mathrm{g}]_L} e^{-2\pi i m b_\mathrm{g} l/L}. \tag{11}$$

A graphical illustration of the structure of a Gabor multiplier is depicted in Fig. 1.

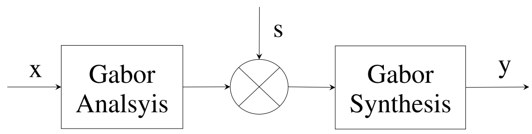

Fig. 1: Structure of a Gabor multiplier: $\mathrm{x} \in \mathbb{C}^L$, $\mathrm{s} \in \mathbb{C}^P$, $\mathrm{y} \in \mathbb{C}^K$, $K \ll L < P$.

Usually, Gabor multipliers are limited to the case where $K = L$, $a_\mathrm{g} = a_\mathrm{h}$, and $b_\mathrm{g} = b_\mathrm{h}$ as this setting preserves the time-frequency structure induced by the two Gabor systems. But in this work, we are particularly interested in obtaining an output measurement vector y which is smaller in size than the input vector x. Hence, we will strictly restrict ourselves to the case, $K \ll L$.

[1] Also referred to as Discrete Gabor Transform (DGT) [22].

[2] Such an operator can also be defined for other systems [23].

D. Random Gabor Multipliers as Measurement Matrices

The linear map from the input vector $\mathrm{x} \in \mathbb{C}^L$ to the measurement vector $\mathrm{y} \in \mathbb{C}^K$ given by (10) and (11) can be rewritten in the form,

$$\mathrm{y} = \mathrm{S_h D_s A_g x} = \Psi \mathrm{x} \tag{12}$$

with the $K \times L$-dimensional measurement matrix Ψ being the product of three matrices $\mathrm{S_h}$, $\mathrm{D_s}$, and $\mathrm{A_g}$, where

- $\mathrm{S_h} \in \mathbb{C}^{K \times P}$ is the matrix representation of the synthesis operator associated to the Gabor system $(h, a_\mathrm{h}, b_\mathrm{h})$,
- $\mathrm{A_g} \in \mathbb{C}^{P \times L}$ is the matrix representation of the analysis operator associated to the Gabor system $(g, a_\mathrm{g}, b_\mathrm{g})$, and
- $\mathrm{D_s} \in \mathbb{C}^{P \times P}$ is the diagonal matrix containing the elements of symbol vector $\mathrm{s} \in \mathbb{C}^P$ on its diagonal.

Note that the symbol vector $\mathrm{s} \in \mathbb{C}^P$ is obtained by stacking the columns of $\mathrm{s}_{m,n}$ in a vector. With the notation introduced in (7), we have

$$\mathrm{S_h} = \begin{bmatrix} \vdots & & \vdots & \vdots & & \vdots \\ \mathrm{h}_{0,0} & \cdots & \mathrm{h}_{0,M-1} & \mathrm{h}_{1,0} & \cdots & \mathrm{h}_{N-1,M-1} \\ \vdots & & \vdots & \vdots & & \vdots \end{bmatrix}$$

and

$$\mathrm{A_g} = \begin{bmatrix} \cdots & \mathrm{g}_{0,0}^* & \cdots \\ & \vdots & \\ \cdots & \mathrm{g}_{0,M-1}^* & \cdots \\ \cdots & \mathrm{g}_{1,0}^* & \cdots \\ & \vdots & \\ \cdots & \mathrm{g}_{N-1,M-1}^* & \cdots \end{bmatrix}.$$

We will consider two different stochastic models for Gabor multipliers. In the first model, which we denote by RGM-I, the windows g and h are both chosen as sampled and periodized Gaussian *deterministic* windows, whereas the symbol is assumed to be *random* with i.i.d. $\mathrm{s}_{m,n} \sim \mathcal{CN}(0,1)$ drawn from the standard complex Gaussian distribution. In the second model, denoted by RGM-II, the symbol is distributed as in RGM-I but both windows g and h are chosen at random as well. More specifically, the elements of the windows are i.i.d. $\mathrm{g}[l], \mathrm{h}[k] \sim \mathcal{N}(0,1)$ drawn from the standard real Gaussian distribution. In Fig. 2 (a) and (b) we depict a realization of a measurement matrix $\Psi \in \mathbb{C}^{128 \times 256}$ for the two stochastic models RGM-I and RGM-II. The exact parameters of the involved Gabor systems are specified in the second row of Table I. For comparison, cf. Fig. 2 (c), we also show a realization of a measurement matrix, for which each entry is i.i.d. and drawn from the standard complex Gaussian distribution $\mathcal{CN}(0,1)$. Note that this *Fully Gaussian (FG)* stochastic model corresponds to a completely unstructured random matrix.

As can be seen, the measurement matrix Ψ given by a Gabor multiplier is quite sparse for model RGM-I but for model RGM-II the Gabor multiplier matrix becomes more dense, almost resembling the FG measurement matrix. This is remarkable because the FG model involves $2 \cdot 128 \cdot 256 = 65536$ real Gaussian random variables, whereas the RGM-II model only involves $128 + 256 + 2 \cdot 512 = 1408$ (the RGM-I model even only 1024) real Gaussian random variables.

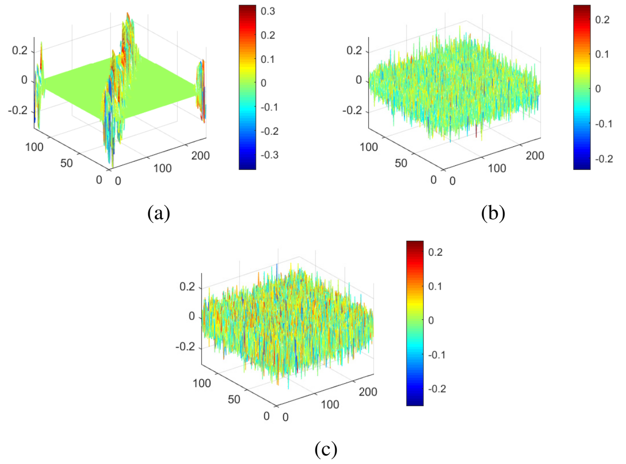

Fig. 2: Realizations of measurement matrices for various stochastic models (real parts, properly scaled): (a) RGM-I, (b) RGM-II, and (c) Fully Gaussian.

IV. Numerical Experiments

In this section, we evaluate the performance of a random Gabor multiplier as a measurement matrix. For this purpose, we carry out simulations with synthetic signals that are sparse with respect to Fourier or Gabor dictionaries. We will use the FG model as a benchmark for comparison. Finally, we will apply a random Gabor multiplier to an audio signal. All the simulations are implemented in MATLAB using the LTFAT toolbox [22], [24].

A. Sparse Signal in Frequency Domain

For our first simulation, we choose an input signal x of length $L = 256$ that is sparse in the Fourier domain, i.e., $\mathrm{x} = \Phi\alpha$, where the sparsifying dictionary Φ is a 256×256 discrete Fourier transform (DFT) matrix. The transform coefficient vector α of length 256 is S-sparse with $S = \{10, 20, 30, 40, 50, 60\}$. The values of the non-zero coefficients are assumed to be i.i.d. and are drawn from the standard complex Gaussian distribution $\mathcal{CN}(0,1)$. The support sets of cardinality S are chosen uniformly random.

B. Sparse Signal in Time-Frequency (TF) Domain

For our second simulation, we choose an input signal x of length $L = 256$ that is sparse in the TF domain, i.e. $\mathrm{x} = \Phi\alpha$, where Φ of dimension 256×512 is a Gabor synthesis matrix generated using the Hann window with parameters $a = 8$ and $b = 16$. Just like above, we generate the S-sparse transform coefficient vector α of length 512 for $S = \{10, 20, 30, 40, 50, 60\}$. Again, the values of the non-zero coefficients are assumed to be i.i.d. and are drawn from the standard complex Gaussian distribution $\mathcal{CN}(0,1)$. The support sets of cardinality S are chosen uniformly random.

For the two scenarios specified in Sections IV-A and IV-B, we obtain the measurement vector y using $\mathrm{y} = \Psi\mathrm{x}$, where Ψ is a realization of a random Gabor multiplier matrix, which is chosen according to the stochastic models RGM-I and RGM-II defined in Section III-D. We recover the coefficient vector $\hat{\alpha}$ from the measurement vector y by solving the ℓ_1–minimization problem using the SeDuMi solver in CVX software [25] and then reconstruct $\hat{\mathrm{x}}$ using (6). For each value of sparsity S, we compute 500 realizations of x and plot the performance curve depicting the success rate for the two random Gabor multipliers specified in Table I. Here, reconstruction is declared as successful if the relative reconstruction error, $\|\mathrm{x} - \hat{\mathrm{x}}\|/\|\mathrm{x}\|$ is smaller than 10^{-5}.

Gabor system $(\mathrm{h}, a_\mathrm{h}, b_\mathrm{h})$	Gabor system $(\mathrm{g}, a_\mathrm{g}, b_\mathrm{g})$
$K = 64$ $a_\mathrm{h} = 4 \quad b_\mathrm{h} = 2$ $R_\mathrm{h} = 8 \quad P = 512$	$L = 256$ $a_\mathrm{g} = 16 \quad b_\mathrm{g} = 8$ $R_\mathrm{g} = 2 \quad P = 512$
$K = 128$ $a_\mathrm{h} = 16 \quad b_\mathrm{h} = 2$ $R_\mathrm{h} = 4 \quad P = 512$	$L = 256$ $a_\mathrm{g} = 32 \quad b_\mathrm{g} = 4$ $R_\mathrm{g} = 2 \quad P = 512$

TABLE I: Parameters of Gabor multipliers.

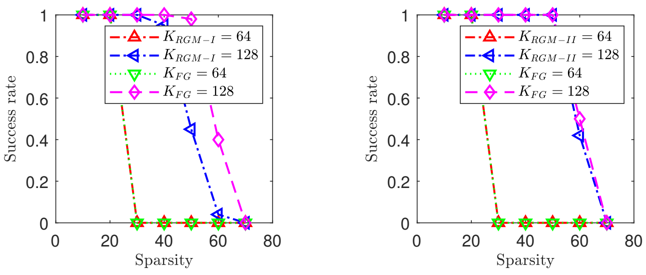

Fig. 3: Sparsity vs Success Rate for Φ = DFT matrix: (a) RGM-I and (b) RGM-II.

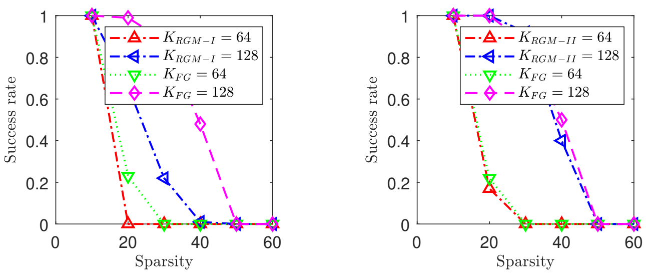

Fig. 4: Sparsity vs Success Rate for Φ = Gabor synthesis matrix: (a) RGM-I and (b) RGM-II.

Note that the realizations of the Gabor multiplier matrices are drawn only once and remain fixed throughout all 500 realizations of x and all sparsity values S. Figure 3 depicts the result for the DFT dictionary, whereas Figure 4 shows the result for the Gabor dictionary.

It can be observed that random Gabor multipliers have excellent compression capabilities of signals that are sparse in frequency or TF domains. For $K = 64$ measurements, which corresponds to a compression ratio of $1:4$, the performance of RGM-I, RGM-II, and FG is very similar for all considered scenarios. For $K = 128$ measurements (compression ratio of $1:2$) we notice a performance degradation of RGM-I compared with RGM-II. Hence, a random choice of the windows of the Gabor multiplier improves its applicability as CS measurement matrix. Even more, we see that RGM-II and FG have similar performance although RGM-II corresponds to a highly structured random matrix, whereas FG is fully unstructured. Note that the involved randomness in the RGM-II model is only a small fraction of the randomness in the FG model: RGM-II involves $128 + 256 + 2 \cdot 512 = 1408$ real Gaussian random variables, whereas FG involves $2 \cdot 128 \cdot 256 = 65536$ real Gaussian random variables (for $K = 128$).

C. *Approximately Sparse Signal in Time-Frequency Domain*

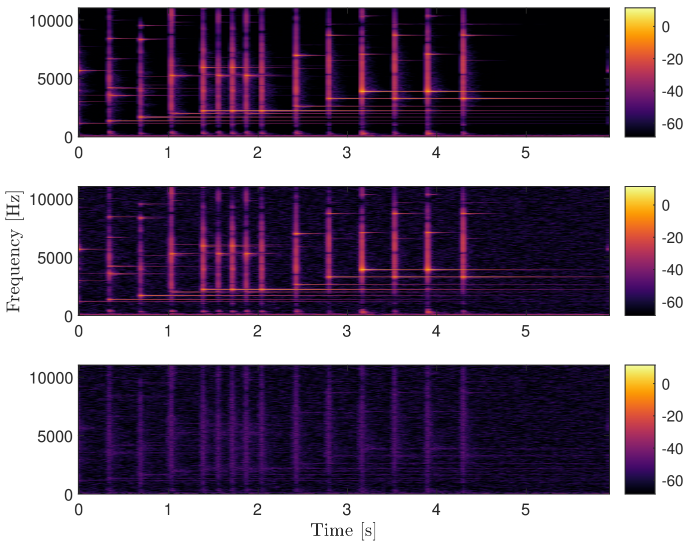

Fig. 5: Reconstruction of the DGT magnitude of the audio signal "Glockenspiel": original signal (top), reconstructed signal (middle), and difference between original and reconstructed signal (bottom).

In this simulation, we choose the (in)famous "Glockenspiel" [22] as our input audio signal. This signal is sampled at 22.05 kHz and consists of 131072 samples. Since the audio signal is real-valued, we aim to recover only the DGT coefficients of positive frequencies instead of the full DGT coefficients. This means that, for $\mathrm{x} = \Phi\alpha$, Φ is a Gabor synthesis matrix that only computes positive frequency DGT coefficients. The chosen DGT parameters are $a = 512$ and $b = 64$ with a Hann window. Hence, the dimension of Φ is 131072×262400. As shown in Fig. 5, this signal is approximately sparse in the TF domain. Similar to the previous two examples, we obtain the measurement vector y using $y = \Psi x$, where Ψ is a realization of a random Gabor multiplier matrix based on the stochastic model RGM-II. Table II summarizes the parameters used in the Gabor systems.

Gabor system; (h, a_h, b_h)	Gabor system; (g, a_g, b_g)
$K = 65536$	$L = 131072$
$a_h = 128 \quad b_h = 32$	$a_g = 256 \quad b_g = 64$
$R_h = 16 \quad P = 524800$	$R_g = 8 \quad P = 524800$

TABLE II: Parameters of Gabor multiplier.

Since, this is a large scale problem we employ SPG-l1 software [26] to recover the TF coefficients and apply (6) to reconstruct the original audio signal. Note that a comparison with the FG model is computationally infeasible[3] since this would require to generate and store a realization Ψ of an i.i.d. Gaussian matrix of size 65536 × 131072. It can be seen from Fig. 5 that the difference between original and reconstructed signal is very small.

V. Conclusion

We demonstrated that random Gabor multipliers can be efficiently used for compressive sampling of signals that are sparse with respect to Fourier or Gabor dictionaries. We investigated two different stochastic models for Gabor multipliers and showed that the resulting measurement matrices have (almost) similar compression performance as i.i.d. Gaussian random measurement matrices. This is remarkable because the involved randomness is only a small fraction compared with the FG model. Furthermore, we showed that random Gabor multipliers exhibit excellent compression performance not only for ideally sparse but also for approximately sparse (compressible) signals. This is important because real-world signals such as audio signals are rarely ideally sparse. Our experiments indicated that for audio signals a compression ratio of 1 : 2 is easily possible without notable differences. Therefore, we believe that random Gabor multipliers will be a very useful tool for present and future audio applications.

Acknowledgement

The work of S.R., P.B., and L.D.A. was supported by the Austrian Science Fund (FWF) START-project Y 551-N13 (FLAME). The work of G.T. and P.B. was supported by the Vienna Science and Technology Fund (WWTF) project MA16-53 (INSIGHT). The work of L.D.A. was supported by the Austrian Science Fund (FWF) project P 31225.

References

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

[2] E. J. 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, Feb. 2006.

[3] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*, ser. Applied and Numerical Harmonic Analysis. Basel: Birkhäuser, 2013.

[4] R. G. Baraniuk, "Compressive sensing [lecture notes]," *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 118–121, July 2007.

[5] R. Calderbank, S. Howard, and S. Jafarpour, "Construction of a large class of deterministic sensing matrices that satisfy a statistical isometry property," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 358–374, April 2010.

[6] D. G. Mixon, "Explicit matrices with the restricted isometry property: Breaking the square-root bottleneck," in *Compressed Sensing and its Applications*. Springer, 2015, pp. 389–417.

[7] M. Rudelson and R. Vershynin, "On sparse reconstruction from Fourier and Gaussian measurements," *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, vol. 61, no. 8, pp. 1025–1045, 2008.

[8] J. A. Tropp, M. B. Wakin, M. F. Duarte, D. Baron, and R. G. Baraniuk, "Random filters for compressive sampling and reconstruction," in *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, vol. 3. IEEE, 2006, pp. III–III.

[9] J. Romberg, "Compressive sensing by random convolution," *SIAM Journal on Imaging Sciences*, vol. 2, no. 4, pp. 1098–1128, 2009.

[10] H. Rauhut, J. Romberg, and J. A. Tropp, "Restricted isometries for partial random circulant matrices," *Applied and Computational Harmonic Analysis*, vol. 32, no. 2, pp. 242–254, 2012.

[11] F. Krahmer, S. Mendelson, and H. Rauhut, "Suprema of chaos processes and the restricted isometry property," *Communications on Pure and Applied Mathematics*, vol. 67, no. 11, pp. 1877–1904, 2014.

[12] S. Mendelson, H. Rauhut, and R. Ward, "Improved bounds for sparse recovery from subsampled random convolutions," *The Annals of Applied Probability*, vol. 28, no. 6, pp. 3491–3527, 2018.

[13] G. E. Pfander and H. Rauhut, "Sparsity in time-frequency representations," *Journal of Fourier Analysis and Applications*, vol. 16, no. 2, pp. 233–260, 2010.

[14] G. E. Pfander, H. Rauhut, and J. A. Tropp, "The restricted isometry property for time–frequency structured random matrices," *Probability Theory and Related Fields*, vol. 156, no. 3-4, pp. 707–737, 2013.

[15] G. Tauböck, S. Rajbamshi, P. Balazs, and L. D. Abreu, "Random Gabor multipliers and compressive sensing," in *Sampling Theory and Applications 2019 (SampTA 2019)*, Bordeaux, France, July 2019.

[16] H. G. Feichtinger and K. Nowak, "A first survey of Gabor multipliers," in *Advances in Gabor analysis*. Springer, 2003, pp. 99–128.

[17] H. Rauhut, K. Schnass, and P. Vandergheynst, "Compressed sensing and redundant dictionaries," *IEEE Transactions on Information Theory*, vol. 54, no. 5, pp. 2210–2219, 2008.

[18] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM Journal on Computing*, vol. 24, no. 2, pp. 227–234, 1995.

[19] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, 2005.

[20] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1998.

[21] P. L. Søndergaard, P. C. Hansen, and O. Christensen, "Finite discrete Gabor analysis," Ph.D. dissertation, Technical University of Denmark, 2007.

[22] P. L. Søndergaard, B. Torrésani, and P. Balazs, "The linear time frequency analysis toolbox," *International Journal of Wavelets, Multiresolution and Information Processing*, vol. 10, no. 04, p. 1250032, 2012.

[23] P. Balazs, "Basic definition and properties of Bessel multipliers," *Journal of Mathematical Analysis and Applications*, vol. 325, no. 1, pp. 571–585, 2007.

[24] Z. Průša, P. L. Søndergaard, N. Holighaus, C. Wiesmeyr, and P. Balazs, "The Large Time-Frequency Analysis Toolbox 2.0," in *Sound, Music, and Motion*, ser. LNCS. Springer International Publishing, 2014, pp. 419–442.

[25] M. Grant, S. Boyd, and Y. Ye, "CVX: Matlab software for disciplined convex programming (2008)," *Web page and software available at http://stanford. edu/ boyd/cvx*, 2015.

[26] E. G. Birgin, J. M. Martínez, and M. Raydan, "Algorithm 813: SPG—software for convex-constrained optimization," *ACM Transactions on Mathematical Software (TOMS)*, vol. 27, no. 3, pp. 340–349, 2001.

[3] For standard equipment. In our experiments, the storage requirement in MATLAB was about 64GB for a matrix of this size.