

Investigation of the Effect of Different Covariance Estimation Methods on the Performance of Least Squares Compressive Covariance Sensing

Nuha A. S. Alwan¹

Department of Electrical Engineering, College of Engineering, University of Baghdad, Baghdad, Iraq

Abstract. Compressive covariance sensing (CCS) can recover the second-order statistics of a signal that has undergone compression, and this can be achieved without the requirement of sparsity conditions. Instead, certain structure information in the statistical domain is to be captured during compression. In particular, least squares compressive covariance sensing is considered which requires the estimation of the covariance of the available compressed signal in order to recover the covariance matrix of the original signal. Different covariance estimation methods are applied and the CCS performance compared, in the presence of white Gaussian noise, in terms of the normalized mean square error between the true and recovered covariance.

Keywords. compressive covariance sensing, least squares, linear sparse ruler, covariance matching, covariance estimation methods, preprocessing.

1. Introduction

Compressive sensing (CS) [1] is a technique that enables simultaneous signal acquisition and compression to limit sensing, storage and communication costs using sub-Nyquist sampling of the signal. Recovery of the original signal is possible provided the latter has a sparse representation in a known transformed domain. This paradigm of CS has impacted a wide range of applications in communications, networking and signal and image processing. On the other hand, compressive covariance sensing (CCS) [2] can recover the second-order statistics of a signal after compression rather than the signal itself, but without the requirement of the sparsity condition. In this case, structure forms other than sparsity are to be captured whilst the signal undergoes compression. Such structure information is present in the statistical domain enabling the reconstruction of second-order statistics of a wide-sense stationary (WSS) signal, and recently [3], even non-stationary signals via online CCS. Since many signal processing techniques and methodologies are based on second-order statistics, CCS finds applications in power spectrum estimation, frequency estimation and direction-of-arrival (DOA) estimation, among others. In CCS, it is generally assumed that the covariance matrix of the original signal has a Hermitian Toeplitz structure such that the covariance matrix can be reconstructed from the covariance sequence, so that, actually, it is the recovery of the covariance sequence that is aimed at.

¹ Corresponding Author, email: n.alwan@ieee.org

This paper focuses on least squares compressive covariance sensing (LS-CCS) in which compression is achieved by a sparse ruler that identifies the sampling pattern. The latter is such that a ruler of length $N-1$ can measure all integer lags from zero to $N-1$ according to the sparse marks of the ruler [2, 3]. This leads to a development of an overdetermined problem due to the assumed Hermitian Toeplitz structure of the covariance matrix [4, 2]. The overdetermined problem can then be solved by least squares leading to the LS-CCS method which involves the computation of an estimate of the covariance matrix of the compressed signal as a step towards recovering the covariance matrix of the original signal.

Different covariance estimates of the compressed signal exhibit different properties and thereby affect the CCS process. The purpose of the present work is to assess the performance of the LS-CCS method using different covariance estimation methods in the presence of white Gaussian noise (WGN). It is found that the unbiased covariance estimation method yields the optimum results.

The rest of the paper is organized as follows: Section 2 presents covariance and sparse ruler sampling. In Section 3, the LS-CCS method is explained, and different methods of covariance estimation are also discussed. The simulation results are presented in Section 4. Finally, Section 5 concludes the paper.

2. Covariance and Sparse Rule Sampling

The autocorrelation of a WSS random signal $x(n)$ is given by:

$$\mathbf{r}_x(\tau) = E[x(n)x^*(n-\tau)] , \quad (1)$$

where the operator E is the expectation or ensemble average operator, the asterisk represents the complex conjugate, n is the time index and τ is the time shift or lag. Another second-order statistic is the autocovariance given by:

$$\mathbf{c}_x(\tau) = E[(x(n)-\mu)(x^*(n-\tau)-\mu)] , \quad (2)$$

where μ is the statistical mean of the random signal $x(n)$. Clearly, if the mean is zero, autocorrelation and autocovariance, as given by Equations 1 and 2 respectively, are interchangeable. For simplicity, autocovariance will be referred to as covariance throughout the paper. Moreover, if the random process is also ergodic [5], the theoretical ensemble average can be replaced by a time average which is easily calculated as follows:

$$\mathbf{c}_x(\tau) = \frac{1}{T} \sum_{n=1}^T x(n)x^*(n-\tau) \quad (3)$$

where T is the number of terms to be averaged.

If we assume that the available number of signal samples is N , the covariance can be computed for lag values ranging from 0 to $N-1$. The covariance as a sequence is then defined by:

$$\mathbf{c}_x = [c_x(0), c_x(1), \dots, c_x(N-1)] . \quad (4)$$

In applications, covariance is generally described as a matrix and is characterized by specific structures such as positive semidefinite Hermitian Toeplitz [5]. As such, the covariance matrix can be constructed solely from its first column which is none other than the sequence given by Equation 4. In the present work, only real signals will be considered, and therefore, the covariance matrix structure is reduced to symmetric Toeplitz (ST).

As explained in the introduction, CCS recovers the covariance of a signal from a compressed version of this signal to attain the entailed compression advantages. This is possible if compression is achieved using a sparse ruler. A linear sparse ruler (LSR) is shown in Figure 1. It can be thought of as a ruler with missing marks. The existing remaining marks allow all integer distances between zero and the ruler length to be measured.

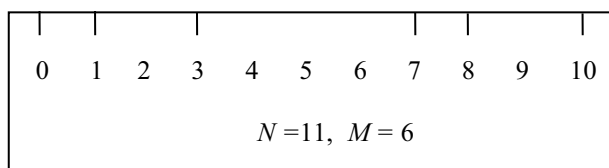


Figure 1. An example of a linear sparse ruler of length $N-1=10$.
 $K=\{0, 1, 3, 7, 8, 10\}$

As shown in Figure 1, a length- $(N-1)$ LSR consists of M marks, usually with $M \ll N$. M takes on integer values between 0 and $N-1$. The ruler can be thought of as a set $K \subset \{0, 1, \dots, N-1\}$ on the condition that for all lag values τ between 0 and $N-1$, there is at least one pair of elements in K , namely (k, k') , such that this pair satisfies $k - k' = \tau$, with $k > k'$ [2]. When this LSR of length $N-1$, represented by K , is synchronized with a signal vector of length N , it will sparsely sample the signal vector at its M marks resulting in a length- M compressed signal vector. Since the LSR can measure all distances from 0 to $N-1$, all possible lags can be derived and the covariance sequence of the original signal can be estimated from the compressed signal.

The compression ratio of the length- $(N-1)$ LSR is given by M/N , and is almost inversely proportional to the length $N-1$. Therefore, long LSRs perform better. There are also circular sparse rulers (CSR) that yield better compressibility than LSRs because of their ability to measure two different distances with each pair of marks [2]. However, the discussion in this work will be concerned with LSRs only.

Although there arise in applications covariance matrix structures other than Toeplitz such as circulant [6] or banded [7], most works on CCS deal with estimating Toeplitz matrices with non-periodic sparse samplers using the LSR, requiring at least a pair of samples in the compressed signal vector for each possible value of lag [4].

3. Least Squares Compressive Covariance Sensing (LS-CCS)

To formulate the CCS problem first, consider the recovery of the covariance of a zero-mean random signal vector $\mathbf{x} \in \mathbf{R}^N$ from the compressed signal vector $\mathbf{y} \in \mathbf{R}^M$ given by:

$$\mathbf{y} = \Phi \mathbf{x} \quad (5)$$

where the matrix $\Phi \in \mathbf{R}^{M \times N}$ is called the compression matrix or sampler, and $M \ll N$. Several realizations of the vector \mathbf{y} may be available.

The matrix Φ performs mathematically the function of the sparse ruler. It is a sparse matrix with at most one non-zero entry in each row or column. Each of the M rows of Φ contains a 'one' at one of the corresponding M positions of the LSR and zeros elsewhere. In contrast to sparse sampling matrices, there are applications that employ dense sampling matrices whose design relies on probabilistic arguments and that have been proven successful [2]. These, however, will not be considered further in this work.

The theoretical covariance matrix described by Equations 1 and 2 will be re-written here in terms of the vector \mathbf{x} , and given a temporary different symbol here (Σ) for notation convenience:

$$\Sigma = E[\mathbf{x} \mathbf{x}^T]. \quad (6)$$

The matrix Σ is assumed to be a linear combination of the ST matrices Σ_s that are elements of the set $\mathfrak{S} = \{\Sigma_0, \Sigma_1, \dots, \Sigma_{S-1}\} \subset \mathbf{R}^{N \times N}$. This implies that there exist real scalars α_s such that:

$$\Sigma = \sum_{s=0}^{S-1} \alpha_s \Sigma_s. \quad (7)$$

When the subspace \mathfrak{S} is assumed to be a linearly independent set of matrices, the decomposition in Equation 7 is unique such that a knowledge of the α_s leads to knowing Σ . The above ST structure for Σ that is characteristic of WSS processes enables a certain degree of compression. If no prior information about Σ is available and the latter is simply considered to be a symmetric positive semidefinite matrix, then no compression is possible that preserves the second-order statistics of the original signal. The condition $S < 2N - 1$ must be satisfied for \mathfrak{S} to be linearly independent [4].

The covariance of the vector \mathbf{y} is given by

$$\bar{\Sigma} = E[\mathbf{y} \mathbf{y}^T]. \quad (8)$$

Substituting Equations 5 and 6 in 8 yields:

$$\bar{\Sigma} = \Phi \Sigma \Phi^T = \sum_{s=0}^{S-1} \alpha_s \bar{\Sigma}_s, \quad (9)$$

$$\text{where } \bar{\Sigma}_s = \Phi \Sigma_s \Phi^T, \quad (10)$$

so that $\bar{\Sigma}$ is linear combination of the symmetric matrices that are elements of the set $\bar{\mathcal{S}} = \{\bar{\Sigma}_0, \bar{\Sigma}_1, \dots, \bar{\Sigma}_{S-1}\} \subset \mathbf{R}^{M \times M}$. These symmetric matrices in the subspace $\bar{\mathcal{S}}$ are not necessarily Toeplitz. If compression preserves the second-order statistical information, that is if, for example, it is achieved according to the LSR described in Section 2, then $\bar{\mathcal{S}}$ is linearly independent and knowing Σ_s leads to knowing $\bar{\Sigma}_s$ from Equation 10. This entails knowing the α_s from Equation 9 and finally knowing Σ using Equation 7.

The problem of estimating the α_s from the linear parameterization of Equations 9 and 10 is known in the literature as structured covariance estimation or covariance matching [8, 4], and it plays an important role in CCS as will be shown by way of the following example. The example also clarifies the need for a least squares solution to an overdetermined system of equations.

3.1. CCS Example

We assume the original signal vector \mathbf{x} to have limited dimension ($N=4$) to clarify the CCS concepts discussed so far. Let the vector \mathbf{x} be given by:

$$\mathbf{x} = [x_o \ x_1 \ x_2 \ x_3]^T$$

A LSR of $M=3$ will be used to compress the above vector. Due to the small dimension ($N=4$), M cannot be made smaller than 3 if the second-order statistical information is to be preserved. This LSR is mathematically equivalent to multiplying \mathbf{x} by the following sparse sampling matrix:

$$\Phi = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Using Equation 5, the compressed vector will be:

$$\mathbf{y} = [x_o \ x_2 \ x_3].$$

Therefore, Σ has dimensions of 4×4 , whereas $\bar{\Sigma}$ has dimensions of 3×3 . The covariance matrix of the compressed signal, $\bar{\Sigma}$, can be readily computed from the vector \mathbf{y} using one of the covariance estimation methods to be discussed shortly, one of which is given by the practical computation of Equation 3. The steps for recovering the covariance matrix of the original signal, Σ , are the following;

- Choose the matrices Σ_s such that they are symmetric, Toeplitz and linearly independent. We can choose:

$$\Sigma_o = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad \Sigma_1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad \Sigma_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\Sigma_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

We are taking $S=4$ (or N) since, as explained, it is required that $S < 2N - 1$.

- For each Σ_s , find $\bar{\Sigma}_s$ from Equation 10. The results are:

$$\bar{\Sigma}_o = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad \bar{\Sigma}_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \bar{\Sigma}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad \bar{\Sigma}_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

It is clear that the $\bar{\Sigma}_s$ are symmetric and linearly independent, but not necessarily Toeplitz.

- Compute the covariance matrix of the compressed signal, $\bar{\Sigma}$, using one of the covariance estimation methods to be discussed next. Naturally, the availability of more than a single realization of \mathbf{y} (that is, of \mathbf{x}) gives a more accurate result.
- Find the α_s from Equation 9, knowing $\bar{\Sigma}$ and $\bar{\Sigma}_s$.
- Finally, find Σ from Equation 7.
- The result can be verified by finding the covariance matrix Σ by the same estimation method used to find $\bar{\Sigma}$. If we assume that the covariance sequence vector of the original signal, as in Equation 4, is denoted by \mathbf{c}_x , and the recovered covariance sequence vector computed for this same signal from the above steps is denoted by $\hat{\mathbf{c}}_x$, then the normalized mean square error (NMSE) between the covariance sequence \mathbf{c}_x and its recovered version $\hat{\mathbf{c}}_x$ can be calculated as the ratio of the squared magnitude of the difference vector to that of the original sequence vector, and is given below in dB's:

$$NMSE(dB) = 10 \log_{10} \left[\frac{\|\mathbf{c}_x - \hat{\mathbf{c}}_x\|^2}{\|\mathbf{c}_x\|^2} \right] \quad (11)$$

We notice from the above example that $S=4$, so we have four values of α_s , that is, four unknowns. However, Equation 9 from which these unknowns can be computed provides nine equations as all involved matrices have nine elements each. This is an overdetermined system of equations that can be solved by least squares. In MATLAB, any of the two commands, `lsqr(.)` or `pinv(.)`, can be used.

3.2. Estimation Methods for the Covariance Sequence of the Compressed Signal

As can be seen from Equations 1 and 2, covariance involves averaging. LS-CCS can utilize an additional average as preprocessing to obtain more accurate covariance estimation [2, 9] that can be explained as follows. The NMSE computed from Equation 11 becomes larger as compression increases since the total number of available signal samples is reduced. This is true for any estimation, not just LS estimation. Therefore, to increase accuracy of the estimate, several realizations of the vector \mathbf{y} are needed over which an average is taken. This does not contradict the concept of compression since the average sampling rate is reduced anyway, and it is this reduction in sampling rate that entails the compression advantage of reduced hardware cost [2].

In accordance with Equation 4, the covariance matrix of the compressed signal, $\bar{\Sigma}$, has a covariance sequence, derived from its first column, that can be denoted by \mathbf{c}_y . Regardless of the additional averaging or preprocessing just discussed, common estimation methods to obtain \mathbf{c}_y are as follows:

- Unscaled estimation: This is exemplified by the MATLAB command `xcorr(.)`. This can be expressed as

$$\mathbf{c}_y(\tau) = \sum_{n=1}^T y(n) y(n-\tau) \quad (12)$$

where T is the number of terms to be summed and is equal to $M - \tau$.

- Biased estimation. This is given by:

$$\mathbf{c}_y(\tau) = \frac{1}{M} \sum_{n=1}^T y(n) y(n-\tau) \quad (13)$$

- Unbiased estimation. This is given by:

$$\mathbf{c}_y(\tau) = \frac{1}{T} \sum_{n=1}^T y(n) y(n-\tau) \quad (14)$$

The covariance estimates of Equations 12-14 can all be made more accurate by preprocessing averaging.

4. Simulation Results and Discussion

Simulations are carried out in MATLAB. A length-33 signal vector \mathbf{x} is chosen with the unscaled covariance shown in Figure 2. This has been computed and plotted using the MATLAB function `xcorr(.)` whose argument is set to \mathbf{x} , yielding the covariance for lag values ranging from -32 to 32 that correspond to $\mathbf{c}_x(-32)$ to $\mathbf{c}_x(32)$. The signal vector \mathbf{x} is divided into three parts or three realizations, each of length 11, denoted by \mathbf{x}_1 , \mathbf{x}_2 and \mathbf{x}_3 . Compression and CCS will be performed using one realization (\mathbf{x}_1) and thereafter the three realizations (\mathbf{x}_1 , \mathbf{x}_2 and \mathbf{x}_3) to highlight the increased accuracy advantage of preprocessing discussed in Section 3.2.

Working with length-11 realizations, the CCS results are expected to be $\hat{\mathbf{c}}_x(0)$ to $\hat{\mathbf{c}}_x(10)$. Thus, the original values of \mathbf{c}_x considered for comparison or verification will be limited to $\mathbf{c}_x(0)$ to $\mathbf{c}_x(10)$ only out of the 65 values of Figure 2.

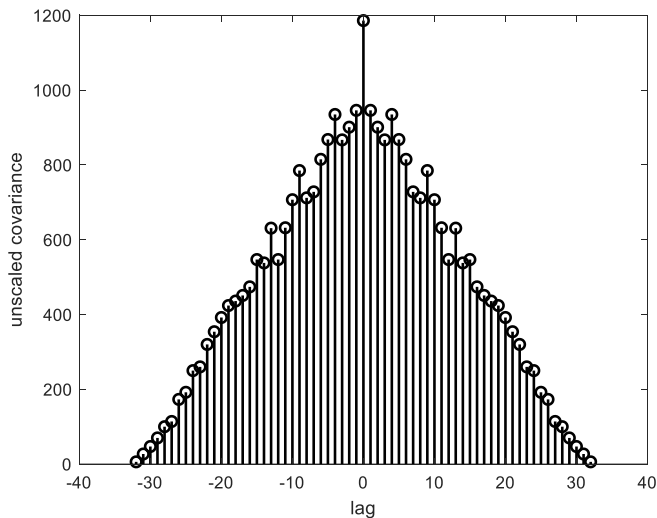


Figure 2. Unscaled covariance of the vector \mathbf{x} of length 33 obtained by the MATLAB command `xcorr(x)`.

For each length-11 realization, the LSR of length $N=11$ shown in Figure 1 is used for compression to obtain a length- M \mathbf{y} vector where $M=6$. For the first realization, CCS is carried out exactly as outlined in the example of Section 3.1, but with respect to the different vectors and matrices dimensions, and the NMSE is calculated using Equation 11.

Preprocessing and averaging over three realizations is done as follows: The LSR of Figure 1 is applied to each of \mathbf{x}_1 , \mathbf{x}_2 and \mathbf{x}_3 , and the $\hat{\mathbf{c}}_x$ values are computed and averaged over the three realizations for $\tau = 0$ to $\tau = 10$ using LS-CCS. Thereafter the NMSE is calculated.

To incorporate WGN, the noise is added to the observed compressed signal vector \mathbf{y} . The compressed signal power is computed for each compressed realization denoted by \mathbf{y}_1 , \mathbf{y}_2 or \mathbf{y}_3 . Each compressed realization is of length $M=6$ corresponding to the length-11 original realizations. The total concatenated compressed vector \mathbf{y} is of length 18 corresponding to the length-33 original signal \mathbf{x} . The standard deviation (SD) of the noise is found by specifying the values of signal-to-noise-ratio (SNR) in dB's and substituting them in:

$$SD = \sqrt{\frac{\text{compressed signal power}}{10^{0.1 \times \text{SNR(dB)}}}}. \quad (15)$$

In the simulations, the above value of SD is computed and added to the compressed realizations. CCS is performed and NMSE is computed in dB from Equation 11. All simulations are averaged over 1000 independent runs.

To compare between the different estimation methods of the covariance \mathbf{c}_y , all the afore-mentioned explained procedure is repeated for each of the estimation methods presented in Section 3.2, and $\hat{\mathbf{c}}_x$ is found in each case by LS-CCS. Note that the computed original \mathbf{c}_x needed in Equation 11 must be found by the same estimation method used for \mathbf{c}_y .

Figure 3 demonstrates the results of the CCS process as a plot of the NMSE versus the SNR when each of the three estimation methods are used for the computation of the covariance of the compressed signal vector. Preprocessing is employed so that the three realizations are used. We notice that the unbiased scaled estimation results in the best CCS performance for moderate and high SNR values whereas the unscaled method is best for low SNR. This is explained as follows. In general, the smaller the value of lag τ , the higher the quality of the estimate because the number of averaging terms is larger for small lags as can be observed from Equations 12-14. Moreover, we note that the biased and unbiased estimation methods in Equations 13 and 14 reduce the covariance magnitudes compared to the unscaled estimation of Equation 12, thereby reducing the importance of the high-lag covariance values even further and making the low-lag values the prominent or dominant ones. It is also noteworthy that the white uncorrelated noise variance or power is actually added only to $\mathbf{c}_y(0)$ and is absent from the rest of the covariance values. Thus, for low SNR the noise level is high and therefore the dominant low-lag covariance values are the noisy ones when the second and third estimation methods are used, leading to poor overall CCS performance. In the first method, the noisy low-lag values are not that dominant because the covariance values for other lags are also quite considerable.

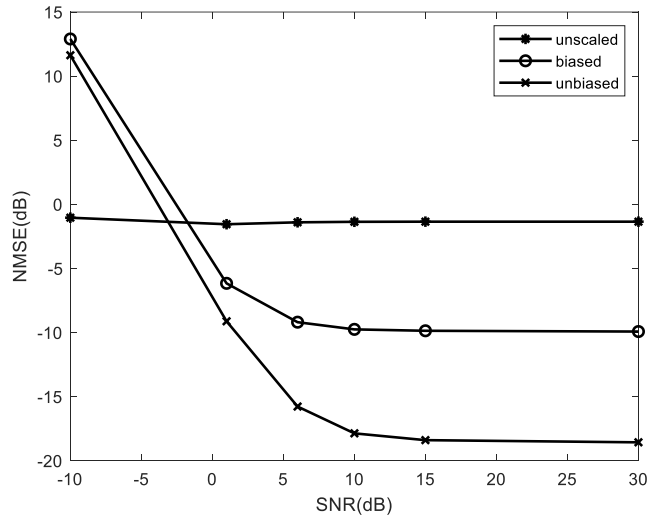


Figure 3. LS-CCS performance as a plot of NMSE (dB) vs. SNR (dB) using the unscaled, biased and unbiased covariance estimation methods. Three realizations are used.

To demonstrate the accuracy advantage of preprocessing, Figure 4 is a plot of NMSE vs. SNR using only the third estimation method (scaled unbiased) with LS-CCS carried out for one and then three realizations. Clearly, the latter case gives better results, that is, smaller NMSE.

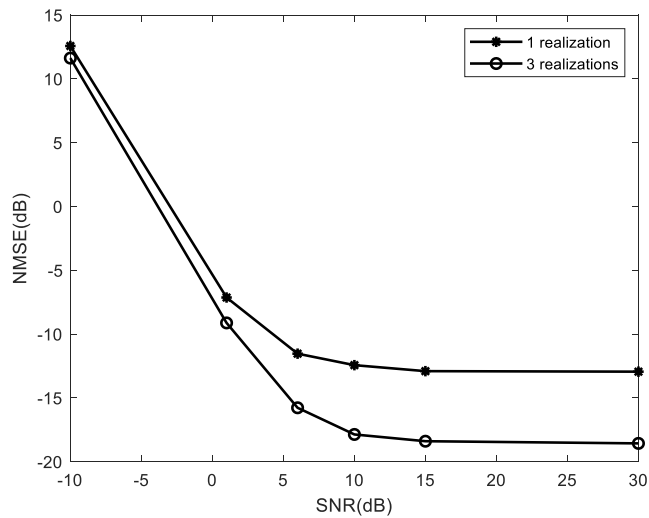


Figure 4. LS-CCS performance as a plot of NMSE (dB) vs. SNR (dB) using the scaled unbiased covariance estimation method, and for the cases of one and three realizations.

5. Conclusion

Least squares CCS is simulated using different covariance estimation methods in computing the compressed signal covariance in order to recover the covariance of the original signal through covariance matching. It is found that the scaled biased and especially the unbiased covariance estimation methods, in achieving LS-CCS, outperform the unscaled method for moderate and high SNR's. This improvement is manifested in terms of the normalized mean square error between the true and recovered covariance values. Additional averaging by preprocessing is found to result in even better CCS performance. As future work, applications that necessitate computation of correlation values from compressed measurements will be researched, such as the important field of single-tone frequency estimation in white and colored noise.

References

- [1] E. J. Candes and M. B. Wakin, An introduction to compressive sampling, *IEEE Signal Processing Magazine* **25** (2008), 21-30.
- [2] D. Romero, D. D. Ariananda, Z. Tian and G. Leus, Compressive covariance sensing: structure-based compressive sensing beyond sparsity, *IEEE Signal Processing Magazine* **33** (2016), 78-93.
- [3] C. Park and B. Lee, Online compressive covariance sensing, *Signal Processing* **162** (2019), 1-9.
- [4] D. Romero and R. Lopez-Valcarce, Compression limits for random vectors with linearly parameterized second-order statistics, *IEEE Transactions on Information Theory* **61** (2015), 1410-1425.
- [5] V. Ingle, S. Kogon and D. Manolakis, Statistical and Adaptive Signal Processing, Artech, 2005.
- [6] C. P. Yen, Y. Tsai and X. Wang, Wideband spectrum sensing based on sub-Nyquist sampling, *IEEE Transactions on Signal Processing* **61** (2013), 3028-3040.
- [7] D. D. Ariananda and G. Leus, Compressive wideband power spectrum estimation, *IEEE Transactions on Signal Processing* **60** (2012), 4775-4789.
- [8] B. Ottersten, P. Stoica and R. Roy, Covariance matching estimation techniques for array signal processing applications, *Digital Signal Processing* **8** (1998), 185-210.
- [9] D. Romero and G. Leus, Wideband spectrum sensing from compressed measurements using spectral prior information, *IEEE Transactions on Signal Processing* **61** (2013), 6232-6246.