

Research Article

A Sliding Window Data Compression Method for Spatial-Time DOA Estimation

Pin-Jiao Zhao ¹, Guo-Bing Hu ¹ and Li-Wei Wang²

¹Department of Electronic and Information Engineering, Jinling Institute of Technology, Nanjing 211169, China

²Nanjing Electronic Devices Institute, Nanjing 210007, China

Correspondence should be addressed to Pin-Jiao Zhao; zhaopinjiao@hrbeu.edu.cn

Received 10 September 2021; Accepted 7 October 2021; Published 28 October 2021

Academic Editor: Fangqing Wen

Copyright © 2021 Pin-Jiao Zhao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a sliding window data compression method for spatial-time direction-of-arrival (DOA) estimation using coprime array. The signal model is firstly formulated by jointly using the temporal and spatial information of the impinging sources. Then, a sliding window data compression processing is performed on the array output matrix to realize fast calculation of time average function, and the computational burden has been reduced accordingly. Based on the concept of sum and difference co-array (SDCA), the vectorized conjugate augmented MUSIC is adopted, with which more sources than twice of the physical sensors can be resolved. Additionally, the sparse array robustness to sensor failure has been evaluated by introducing the concept of essential sensors. The theoretical analysis and numerical simulations are provided to confirm the effectiveness performance of the proposed method.

1. Introduction

Direction-of-arrival (DOA) estimation has been a crucial topic in various practical applications, such as radar, navigation, and wireless communication [1–4], where the antenna arrays are utilized for collecting the spatial sampling of impinging electromagnetic waves. In comparison to the typical uniform linear array (ULA) [5, 6], the emerging sparse arrays [7–9] have remarkable advantages in terms of sensor layout, degrees of freedom (DOF), and virtual array aperture, with which more sources than the number of the physical sensors can be resolved.

Nested array (NA) [10] and coprime array (CPA) [11] are the two most typical sparse array configurations, which have closed form expressions for array geometry and achievable DOF as compared with the minimum redundancy array (MRA) and the minimum hole array (MHA). NA is composed of two concatenated subarrays with increasing element spacing, which is capable of providing $\mathcal{O}(L^2)$ DOF with $\mathcal{O}(L)$ physical sensors, but the existing small interelement spacing in the subarray of NA would cause severe mutual coupling. As for the CPA, the mutual

coupling can be alleviated for the configuration is constructed by a pair of coprime ULAs, which can offer $\mathcal{O}(MN)$ DOF with $\mathcal{O}(M + N)$ sensors, but has holes in the virtual co-array. Recently, lots of modified versions have been further developed, such as super NA [12], enhanced NA [13], augmented NA [14], generalized CPA [15], and thinned CPA [16], to increase the consecutive DOF and reduce the mutual coupling. The sparse arrays mentioned above construct virtual co-array from the view of difference co-array (DCA) and realize the multitarget DOA estimation by exploiting vector MUSIC method or compressed sensing (CS) approach; nevertheless, the number of resolvable sources cannot exceed twice of the physical aperture.

Motivated by the sum co-array (SCA) originating from active sensing [17–19], the concept of sum and difference co-array (SDCA) has provided a new perspective for DOA estimation [20], where the vectorized conjugate augmented MUSIC (VCAM) is presented by jointly using the temporal and spatial information of the impinging sources. The CPA configuration based on SDCA is firstly proposed for spatial-time DOA estimation [21]. Following this, a modified NA configuration named as sum-diff NA (SdNA) is proposed in

[22] for the purpose of resolving more sources. In [23], combining with the VCam method, the unfold CPA configuration on the basis of SDCA is developed to provide more DOF and larger array aperture, and the DOA estimation accuracy is improved accordingly. However, the above spatial-time DOA estimation methods based on SDCA involve high-dimensional data processing of multiple pseudo snapshots, which has high computational load. Additionally, another limitation of the above methods is that the case of sensor failure [24–26] always occurring in the actual direction-finding system has been ignored. The failure of some sensors or receiving channels may destroy the virtual array structure of sparse array and reduce the number of consecutive DOF and the effective array aperture, which results in the performance degradation.

To tackle these problems, a sliding window data compression method for spatial-time DOA estimation is proposed in this paper. The coprime array is adopted from the perspective of SDCA, which can provide more DOF and larger effective array aperture. Then, a sliding window data compression processing is performed on array output matrix to realize fast calculation of time average function. Afterwards, the vectorized conjugate augmented MUSIC is adopted by jointly using the temporal and spatial information of the impinging sources. Moreover, the sparse array robustness to sensor failure has been evaluated by introducing the concept of essential sensors.

Notations: vectors and matrices are denoted by lowercase and uppercase bold-face letters, respectively. $(\cdot)^T$, $(\cdot)^H$, and $(\cdot)^*$ and $|\cdot|$ denote transpose, conjugate transpose, conjugate, and the norm of the embraced matrix, respectively. $\mathbf{0}_{m \times n}$ represents an $m \times n$ null matrix and \mathbf{I}_m represents an $m \times m$ identity matrix. The symbol \otimes denotes the Kronecker product and \odot denotes the Khatri–Rao product.

2. Problem Formulation

2.1. Sparse Array Configuration. Referring to [11], a prototype CPA is composed of two uniform linear subarrays: one is a N -element ULA with interspacing being Md (d denotes the unit interspacing) and the other is an M -element

ULA with interspacing being Nd , where N and M are coprime integers satisfying $M < N$. Intuitively, an example of 8-element CPA with $M = 4$ and $N = 5$ is illustrated in Figure 1, where white circles and black circles, respectively, denote the sensor locations of subarray 1 and subarray 2 and shaded circles denote the overlapping sensor between subarray 1 and subarray 2. For the reason that the two uniform linear subarrays share the same antenna sensor placing at zero position, the CPA have $M + N - 1$ physical sensors, whose locations are given by

$$\mathbb{P}_{CPA} = \{Mnd | n \in \mathbb{Z}(N-1)\} \cup \{Nmd | m \in \mathbb{Z}(M-1)\}, \quad (1)$$

where $\mathbb{Z}(m)$ denotes a set of positive integers ranging from 1 to m and the unit interspacing d is generally set to be half of the wavelength.

2.2. Signal Model. Assume that K far-field narrowband sources from directions $\{\theta_1, \theta_2, \dots, \theta_K\}$ impinge on an L -element CPA configuration with $\theta_k \in (-\pi/2, \pi/2]$, where $L = M + N - 1$. Then, the antenna array output at time t can be expressed as

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{n}(t) = \sum_{k=1}^K \mathbf{a}(\theta_k)s_k(t) + \mathbf{n}(t), \quad (2)$$

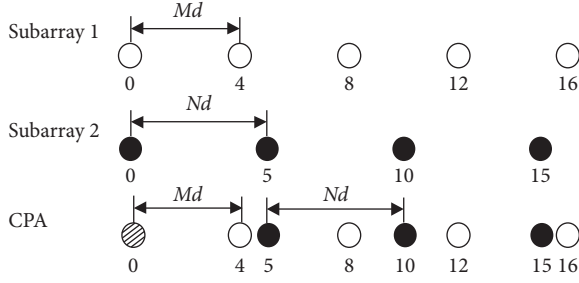
where $\mathbf{a}(\theta_k) = e^{j\pi l \sin \theta_k}$ is the steering vector with respect to the direction θ_k with the sensor location vector being $\mathbf{l} = [l_1, l_2, \dots, l_L]^T$. By stacking all the $\mathbf{a}(\theta_k)$ for $k = 1, 2, \dots, K$, the array manifold matrix $\mathbf{A} = \{\mathbf{a}(\theta_k)\}_{k=1}^K$ can be obtained. $s_k(t) = G_k e^{jw_k t}$ denotes the k th impinging source with G_k and w_k being the deterministic complex amplitude and the small frequency offset, respectively. Accordingly, $\mathbf{s}(t) = \{s_k(t)\}_{k=1}^K$ is the source vector. $\mathbf{n}(t)$ is the Gaussian white noise vector with zero mean and variance being σ_n^2 .

Define the i th and the j th ($1 \leq i, j \leq L$) row of $\mathbf{x}(t)$ as $x_i(t)$ and $x_j(t)$, respectively; then, the time average function for T_p samples can be calculated by

$$R_{x_i^* x_j}(\tau) = \frac{1}{T_p} \sum_{t=1}^{T_p} x_i^*(t) x_j(t + \tau) \approx \sum_{k=1}^K e^{j\pi(l_j - l_i) \sin \theta_k} R_{s_k^* s_k}(\tau) + R_{n_i^* n_j}(\tau), \quad (3)$$

where $\tau \neq 0$ is the time lag ranging from 1 to T_s with T_s being the number of pseudo snapshots, $l_j - l_i$ denotes the generated virtual sensor with $l_i, l_j \in \mathbf{l}$, $R_{s_k^* s_k}(\tau) = |E_k|^2 e^{jw_k \tau}$ can be seen as an equivalent source with amplitude $|E_k|^2$ and frequency offset w_k , which has the similar form as $s_k(t) = G_k e^{jw_k t}$. Since $\mathbf{n}(t)$ is assumed to be the Gaussian white noise, $R_{n_i^* n_j}(\tau) \approx 0$ holds. As compared with the

covariance matrix widely used in spatial DOA estimation methods, $R_{x_i^* x_j}(\tau)$ is constructed by jointly using both temporal and spatial information of the impinging sources, which has potential to expand the effective array aperture and improve the DOA estimation accuracy and angle resolution. Nevertheless, it is worth pointing that (3) involves the sampling of array output matrix with a dimension of

FIGURE 1: Example of 8-element CPA with $M = 4$ and $N = 5$.

$L \times T_s T_p$, which is computationally expensive. Toward this end, a data compression processing method based on sliding window principle is proposed in Section 3.

3. DOA Estimation Based on Sliding Window Data Compression

For the purpose of realizing the fast calculation of time average function and the corresponding pseudo data matrix, the sliding window data compression processing is performed on $x_i(t)$ and $x_j(t)$ ($1 \leq i, j \leq L$) with a period of T_p , which is given by

$$\begin{cases} x_o(t) = x_i(t) * [\mathbf{0}_{T_s \times T_p}; \mathbf{I}_{T_p}; \mathbf{0}_{T_s \times T_p}], \\ x_{\tau+}(t) = x_j(t) * [\mathbf{0}_{(T_s+\tau) \times T_p}; \mathbf{I}_{T_p}; \mathbf{0}_{(T_s-\tau) \times T_p}], \\ x_{\tau-}(t) = x_j(t) * [\mathbf{0}_{(T_s-\tau) \times T_p}; \mathbf{I}_{T_p}; \mathbf{0}_{(T_s+\tau) \times T_p}]. \end{cases} \quad (4)$$

With the sliding window data compression processing in (4), the sampling of the array output matrix with a dimension of $L \times T_s T_p$ can be transformed into the sampling of an equivalent one with a dimension of $L \times (T_p + 2T_s)$; thus, the computational burden can be reduced accordingly. Moreover, notice that the data information is not lost during the process of sliding window data compression, which ensures the DOA estimation accuracy and angle resolution. For the convenience of analysis, we choose the first sensor to be the reference, i.e., $i = 1$. Then, (3) can be written as

$$\begin{cases} R_{ij}^*(\tau) = \frac{1}{T_p} \sum_{t=1}^{T_p} x_o^*(t) x_{\tau+}(t), \\ R_{ij}^*(-\tau) = \frac{1}{T_p} \sum_{t=1}^{T_p} x_o^*(t) x_{\tau-}(t). \end{cases} \quad (5)$$

More intuitively, an example of a short data sequence with $T_p = 3$ and $T_s = 4$ is depicted in Figure 2 for the convenience of description, where the data in blue sliding window stands for the reference, the sliding windows on its right side stand for the cases of $\tau > 0$, and the sliding windows on its left side stand for the cases of $\tau < 0$. It should be noted that the proposed sliding window data compression method involves a long data sequence, which has the same processing principle as the given example of short data sequence.

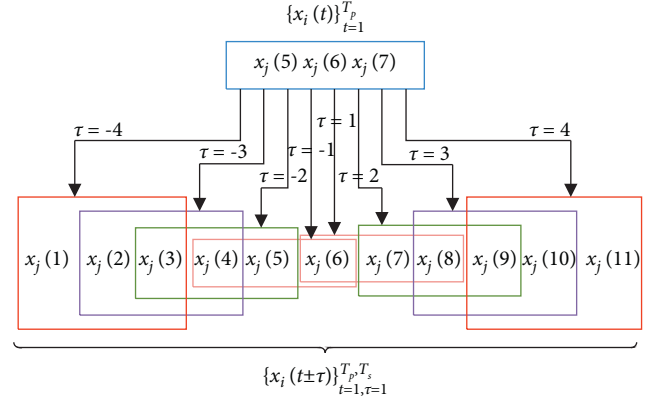


FIGURE 2: Example of sliding window data compression processing.

Since j varies from 1 to L , the conjugate augmented vector can be obtained as

$$\mathbf{g}(\tau) = [\mathbf{g}_x^H(-\tau), \mathbf{g}_x^T(\tau)]^T = [\mathbf{A}^H, \mathbf{A}^T]^T \mathbf{g}_s(\tau), \quad (6)$$

where the time average vector $\mathbf{g}(\tau)$ and its mirrored version $\mathbf{g}_x(-\tau)$ are given as

$$\begin{cases} \mathbf{g}_x(\tau) = \mathbf{A} \mathbf{g}_s(\tau) = [R_{11}^*(\tau), R_{12}^*(\tau), \dots, R_{1L}^*(\tau)]^T, \\ \mathbf{g}_x(-\tau) = \mathbf{A} \mathbf{g}_s(-\tau) = [R_{11}^*(-\tau), R_{12}^*(-\tau), \dots, R_{1L}^*(-\tau)]^T, \end{cases} \quad (7)$$

with $\mathbf{g}_s(\tau) = [R_{s_1^* s_1}(\tau), R_{s_2^* s_2}(\tau), \dots, R_{s_K^* s_K}(\tau)]^T$ and $\mathbf{g}_s(-\tau) = [R_{s_1^* s_1}(-\tau), R_{s_2^* s_2}(-\tau), \dots, R_{s_K^* s_K}(-\tau)]^T$.

Then, for the T_s pseudo snapshots, the pseudo data matrix can be constructed as

$$\mathbf{G} = \{\mathbf{g}(n_s P_s)\}_{n_s=1}^{T_s} = [\mathbf{A}^H, \mathbf{A}^T]^T \mathbf{E} \mathbf{\Omega}, \quad (8)$$

where P_s is the pseudo sampling period satisfying Nyquist sampling principle, \mathbf{E} is a diagonal matrix with main diagonal elements being $|E_k|^2$ and zeros elsewhere, and $\mathbf{\Omega}$ is a $K \times T_s$ matrix with the (k, n) th element being $e^{j\omega_k n P_s}$. By calculating the covariance matrix of \mathbf{G} and then vectorizing it, we have

$$\mathbf{v}_G = (\bar{\mathbf{A}}^H \odot \bar{\mathbf{A}}^T) \mathbf{q}_s, \quad (9)$$

where $\bar{\mathbf{A}} = [\mathbf{A}^H, \mathbf{A}^T]$ is the generated array manifold matrix and \mathbf{q}_s can be seen as an equivalent impinging source vector $K \times 1$ with the k th element being $|E_k|^4$. Additionally, the k th column of $(\bar{\mathbf{A}}^H \odot \bar{\mathbf{A}}^T)$ can be denoted as

$$\tilde{\mathbf{a}}(\theta_k) = \begin{bmatrix} \mathbf{a}(\theta_k) \otimes \mathbf{a}^*(\theta_k) \\ \mathbf{a}(\theta_k) \otimes \mathbf{a}(\theta_k) \\ \mathbf{a}^*(\theta_k) \otimes \mathbf{a}^*(\theta_k) \\ \mathbf{a}^*(\theta_k) \otimes \mathbf{a}(\theta_k) \end{bmatrix}, \quad (10)$$

where the term $\tilde{\mathbf{a}}(\theta_k)$ in (10) behaves like a longer virtual steering vector that can provide more DOF for DOA estimation. More specifically, the union of $\mathbf{a}(\theta_k) \otimes \mathbf{a}^*(\theta_k)$ and $\mathbf{a}^*(\theta_k) \otimes \mathbf{a}(\theta_k)$, respectively, are corresponding to the DCA

TABLE 1: Main steps of the proposed method.

-
- (1) Calculate the time average function $R_{x_i^* x_j}(\tau)$ using (3)
 - (2) Perform sliding window data compression processing on $x_i(t)$ and $x_j(t)$ via (4) and rewrite (3) using (5)
 - (3) Construct the conjugate augmented vector $\mathbf{g}(\tau)$ using (6)
 - (4) Construct the pseudodata matrix \mathbf{G} using (8) and calculate its covariance matrix
 - (5) Vectorize the covariance matrix of \mathbf{G} using (9)
 - (6) Perform spatial smoothing MUSIC method or sparse construction techniques on \mathbf{q}_s
-

and its mirror version and the union of $\mathbf{a}(\theta_k) \otimes \mathbf{a}(\theta_k)$ and $\mathbf{a}^*(\theta_k) \otimes \mathbf{a}^*(\theta_k)$, respectively, are corresponding to the SCA and its mirror version. Then, the spatial smoothing MUSIC method or sparse construction techniques can be performed for DOA estimation. To conclude, the main steps of the proposed method are given in Table 1.

4. Array Robustness Analysis

The array robustness to sensor failure directly affects the DOA estimation performance in the practical direction-finding system and the relevant analysis is discussed in detail in this section. For a sparse array with known array configuration, if the distribution of SDCA changes when one or more sensors are deleted from the physical array (PA), then these sensors are termed as essential sensors. Mathematically, for the sparse array \mathcal{A} and the corresponding SDCA \mathcal{S} , when the l th sensor is deleted from \mathcal{A} , the remaining array becomes $\mathcal{A}_{-l} = \mathcal{A} \setminus \{l\}$ and the corresponding SDCA becomes \mathcal{S}_{-l} . If $\mathcal{S} \neq \mathcal{S}_{-l}$ holds, then the l th sensor is essential for the sparse array \mathcal{A} , but not vice versa.

An example of detecting essential sensor with $\mathcal{A}_1 = \{0, 1, 2, 3, 4, 6\}$ is given in Figure 3, where the red circles and black circles, respectively, denote the locations of PA and SDCA. As can be seen from Figure 3, after removing the sensor 1, the PA becomes $\mathcal{A}_2 = \{0, 2, 3, 4, 6\}$; then, the corresponding SDCA has holes at ± 1 , which implies that sensor 1 is essential. On the contrary, after removing the sensor 2, the PA becomes $\mathcal{A}_3 = \{0, 1, 3, 4, 6\}$; then, the corresponding SDCA is the same as the original one; hence, the sensor 2 of PA is inessential.

Then, the evaluation function of sparse array robustness is calculated as

$$\mathcal{F}_k = |\mathcal{E}_k|/|\mathcal{A}|, \quad (11)$$

where $|\mathcal{E}_k|$ is the number of essential sensors for sparse array \mathcal{A} and $|\mathcal{A}|$ is the number of the whole sparse array \mathcal{A} . According to (10), the values of \mathcal{F}_k ranges from 0 to 1. More specifically, if $\mathcal{F}_k \rightarrow 0$, then the sparse array has strong robustness, but the number of essential sensors is small; if $\mathcal{F}_k \rightarrow 1$, almost all the sensors in sparse array are essential, which is economic, but the array robustness is low. Therefore, several strategies, such as reducing the failure probability of essential sensors and introducing a certain number of inessential sensors, can be adopted to improve the robustness of sparse array and meanwhile ensure the economic benefits.

5. Numerical Simulations

In this section, numerical simulations are performed to evaluate the performance of the proposed method. Consider $K=20$ narrowband sources uniformly distributed between -60° and 60° and impinge on an 8-sensor CPA with SNR being 0 dB and pseudo snapshot number $T_s = 400$, where the sensor locations of CPA is $\mathbb{P}_{CPA} = \{0, 4, 5, 8, 10, 12, 15, 16\}$. To demonstrate the computational efficiency of the proposed method, the averaged CPU times of the proposed sliding window data compression (SWDC) and VCAM [21] *versus* snapshot number T_p over 200 independent Monte Carlo trials are compared in Table 2, where the software used for implementation is MATLAB R2014a (version 8.3) and executed in PC Intel(R) Core(TM) i7-8550U processor with 8.0 GB RAM. It can be observed from Table 2 that the operation time of the proposed SWDC method is significantly reduced in comparison to VCAM, which is mainly attributed to reason that sliding window data compression can transform the antenna received matrix with a dimension of $L \times T_s T_p$ into an $L \times (T_p + 2T_s)$ equivalent one, and the computational load is reduced accordingly.

Then, array robustness to sensor failure is discussed in this simulation. For the 8-element CPA with $\mathbb{P}_{CPA} = \{0, 4, 5, 8, 10, 12, 15, 16\}$, we model the sensor failure by deleting the sensors of CPA one by one; then, the remaining array location, holes of SDCA, and the consecutive DOF are listed in Table 3. As can be seen from Table 3 that if sensor failure occurs to sensor 0, the distribution of SDCA remains unchanged, which implies that sensor "0" is an inessential sensor for the 8-element CPA. Apart from sensor "0," if sensor failure occurs to other sensors, the corresponding SDCA has more holes and less consecutive DOF. Thus, all the sensors in the 8-element CPA are essential, except for sensor "0." Moreover, the importance of each essential sensor is different, e.g., the failure of sensor "5," "10," "12," or "15" would cause more "holes" than the other essential sensors.

The third simulation investigates the DOA estimation performance via MUSIC spectrum. All the conditions are the same as the first simulation except that the snapshot number T_p is set to be 300. In addition, the searching step of MUSIC spectrum is set to be 0.5° . Figure 4 depicts the MUSIC spectrum of the proposed method, where the blue solid line represents the angle estimates and the red dotted line represents incident sources. As can be seen from Figure 4, the proposed method can resolve 20 impinging sources with only 8 sensors and the MUSIC spectrum has sharp and high power peaks in the vicinity of the true impinging sources.

In the last simulation, the DOA estimation performance of the proposed SWDC, VCAM [21], and CPA-MUSIC [11] *versus* SNR and the number of snapshots are compared via 200 independent Monte Carlo trials, where the root mean square error (RMSE) is chosen for evaluating the DOA estimation performance:

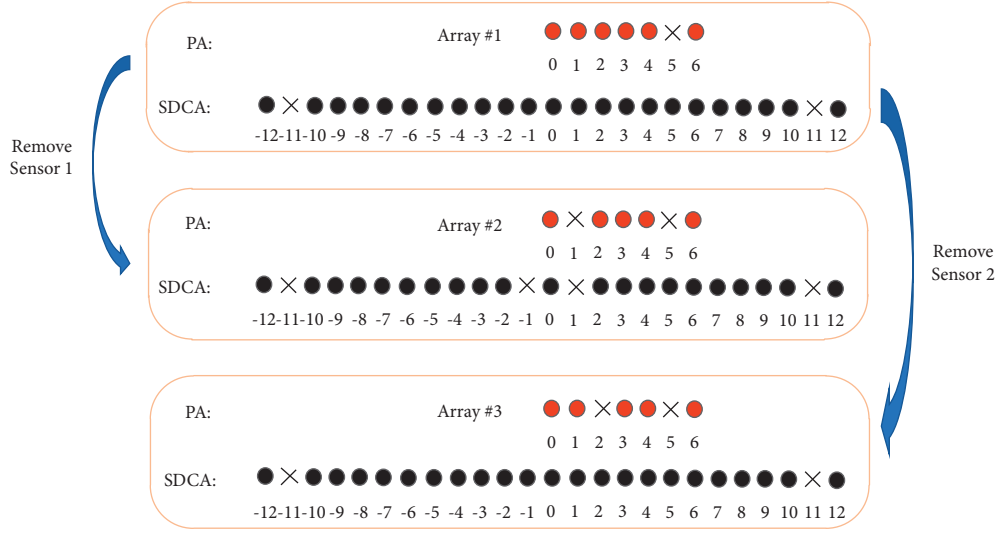


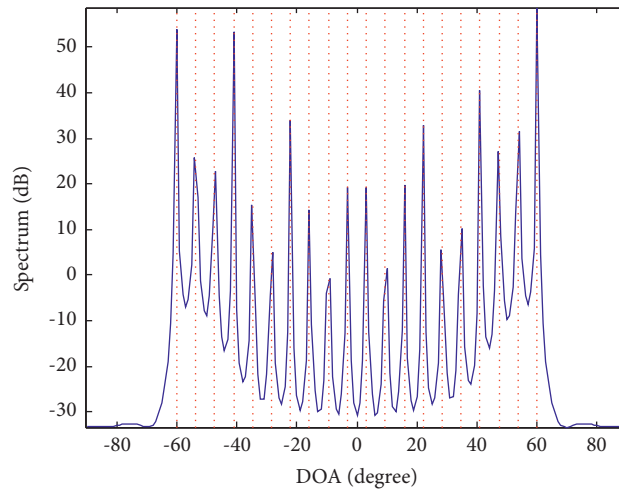
FIGURE 3: Example of detecting essential sensors.

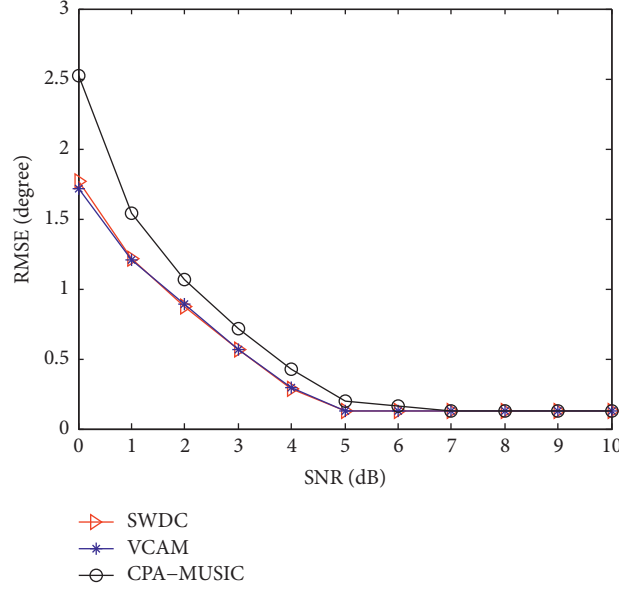
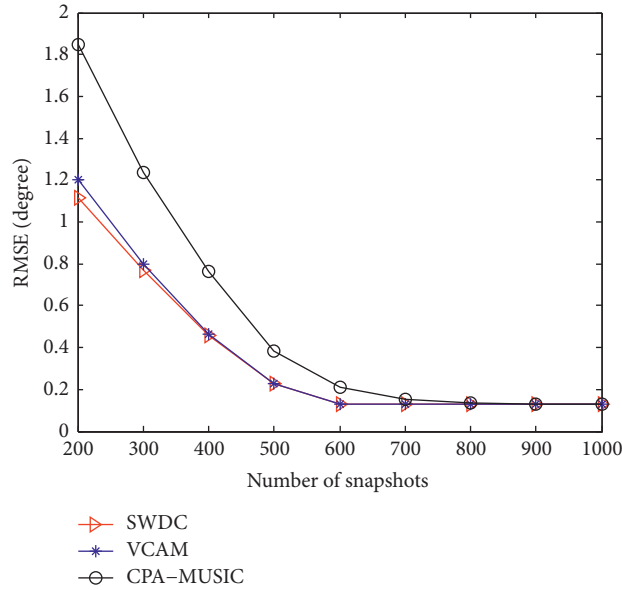
TABLE 2: Averaged CPU times.

Time(sec)	$T_p = 200$	$T_p = 300$	$T_p = 400$	$T_p = 500$	$T_p = 600$
SWDC	0.0544	0.0666	0.0761	0.0852	0.0962
VCAM	0.0995	0.1368	0.1747	0.1950	0.1990

TABLE 3: The array robustness to sensor failure.

Sensor failure	Remaining array	Holes in SDCA	Consecutive DOF
Null	{0, 4, 5, 8, 10, 12, 15, 16}	{-29, 29}	57
"0"	{4, 5, 8, 10, 12, 15, 16}	{-29, 29}	57
"4"	{0, 5, 8, 10, 12, 15, 16}	{-29, -19, -14, 14, 19, 29}	27
"5"	{0, 4, 8, 10, 12, 15, 16}	{-29, -21, -17, -13, -9, 9, 13, 17, 21, 29}	17
"8"	{0, 4, 5, 10, 12, 15, 16}	{-29, -23, -18, -13, 13, 18, 23, 29}	25
"10"	{0, 4, 5, 8, 12, 15, 16}	{-29, -26, -25, -22, -18, -14, -6, -2, 2, 6, 14, 18, 22, 25, 26, 29}	9
"12"	{0, 4, 5, 8, 10, 15, 16}	{-29, -28, -27, -22, -17, 17, 22, 27, 28, 29}	33
"15"	{0, 4, 5, 8, 10, 12, 16}	{-29, -27, -25, -23, -19, 19, 23, 25, 27, 29}	37
"16"	{0, 4, 5, 8, 10, 12, 15}	{-29, -28, -26, -21, 21, 26, 28, 29}	41

FIGURE 4: MUSIC spectrum with SNR=0 dB, $T_p = 300$, and $T_s = 400$.

FIGURE 5: RMSE *versus* SNR with the number of snapshots being 600.FIGURE 6: RMSE *versus* the number of snapshots with SNR being 5 dB.

$$\text{RMSE} = \sqrt{\frac{1}{200K} \sum_{i=1}^{200} \sum_{k=1}^K (\hat{\theta}_{k,i} - \theta_k)^2}, \quad (12)$$

where $\hat{\theta}_{k,i}$ is the estimate of θ_k for the i th Monte Carlo trial. The source distribution in this simulation is the same as that in the third simulation. The results from Figures 5 and 6 show that the DOA estimation accuracy of the proposed SWDC method is similar to that of VCAM, which is superior to the CPA-MUSIC. The reason is that the data information is not lost for the sliding window data compression processing, which implies that the proposed SWDC method and the VCAM method

exploit the same data for DOA estimation, and the estimation accuracy is almost the same accordingly. By contrast, the CPA-MUSIC is performed based on DCA, where the available DOF and virtual array aperture are reduced.

6. Conclusions

In this paper, we have proposed a sliding window data compression method to reduce the computational burden of high-dimensional data processing for the spatial-time DOA estimation. By jointly using the temporal and spatial information of the impinging sources, the signal model is firstly formulated based on CPA from the perspective of

SDCA. Then, a sliding window data compression processing is applied to the array output vector. Afterwards, we resort to the VCAM approach for DOA estimation. In addition, the sparse array robustness to sensor failure has been evaluated by introducing the concept of essential sensors. Simulation results have confirmed that the proposed method can resolve more sources than twice of physical sensors and has notable performance advantages in terms of computational load and DOA estimation accuracy.

Data Availability

No data were used in this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China, under Grant no. 62101223, and Natural Science Foundation of the Jiangsu Higher Education Institutions of China, under Grant nos. 20KJB510027 and 20KJA510008.

References

- [1] H. Krim and M. Viberg, "Two decades of array signal processing research: the parametric approach," *IEEE Signal Processing Magazine*, vol. 13, no. 4, pp. 67–94, 1996.
- [2] B. Ottersten, M. Viberg, P. Stoica, and A. Nehorai, *Exact and Large Sample Maximum Likelihood Techniques for Parameter Estimation and Detection in Array Processing*, Springer, Berlin, Germany, 1993.
- [3] L. Wan, M. Zhang, L. Sun, and X. Wang, "Machine learning empowered IoT for intelligent vehicle location in smart cities," *ACM Transactions on Internet Technology*, vol. 21, no. 3, pp. 1–25, 2021.
- [4] J. He, L. Li, and T. Shu, "Sparse nested arrays with spatially spread orthogonal dipoles: high accuracy passive direction finding with less mutual coupling," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 4, pp. 2337–2345, 2021.
- [5] Z. Zhongfu Ye, J. Jisheng Dai, X. Xu Xu, and fnm Xiaopei Wu, "DOA estimation for uniform linear array with mutual coupling," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 1, pp. 280–288, 2009.
- [6] R. Cao, B. Liu, F. Gao, and X. Zhang, "A low-complex one-snapshot DOA estimation algorithm with massive ULA," *IEEE Communications Letters*, vol. 21, no. 5, pp. 1071–1074, 2017.
- [7] C.-L. Liu and P. P. Vaidyanathan, "Cramér-Rao bounds for coprime and other sparse arrays, which find more sources than sensors," *Digital Signal Processing*, vol. 61, pp. 43–61, 2017.
- [8] Z. Zheng, W.-Q. Wang, Y. Kong, and Y. D. Zhang, "MISC array: a new sparse array design achieving increased degrees of freedom and reduced mutual coupling effect," *IEEE Transactions on Signal Processing*, vol. 67, no. 7, pp. 1728–1741, 2019.
- [9] A. M. Elbir, "Two-dimensional DOA estimation via shifted sparse arrays with higher degrees of freedom," *Circuits, Systems, and Signal Processing*, vol. 38, no. 12, pp. 5549–5575, 2019.
- [10] P. Pal and P. P. Vaidyanathan, "Nested arrays: a novel approach to array processing with enhanced degrees of freedom," *IEEE Transactions on Signal Processing*, vol. 58, no. 8, pp. 4167–4181, 2010.
- [11] P. P. Vaidyanathan and P. Pal, "Sparse sensing with co-prime samplers and arrays," *IEEE Transactions on Signal Processing*, vol. 59, no. 2, pp. 573–586, 2011.
- [12] C.-L. Liu and P. P. Vaidyanathan, "Super nested arrays: linear sparse arrays with reduced mutual coupling-part I: Fundamentals," *IEEE Transactions on Signal Processing*, vol. 64, no. 15, pp. 3997–4012, 2016.
- [13] P. Zhao, G. Hu, Z. Qu, and L. Wang, "Enhanced nested array configuration with hole-free co-array and increasing degrees of freedom for DOA estimation," *IEEE Communications Letters*, vol. 23, no. 12, pp. 2224–2228, 2019.
- [14] J. Liu, Y. Zhang, Y. Lu, S. Ren, and S. Cao, "Augmented nested arrays with enhanced DOF and reduced mutual coupling," *IEEE Transactions on Signal Processing*, vol. 65, no. 21, pp. 5549–5563, 2017.
- [15] S. Qin, Y. D. Zhang, and M. G. Amin, "Generalized coprime array configurations for direction-of-arrival estimation," *IEEE Transactions on Signal Processing*, vol. 63, no. 6, pp. 1377–1390, 2015.
- [16] A. Raza, W. Liu, and Q. Shen, "Thinned coprime array for second-order difference co-array generation with reduced mutual coupling," *IEEE Transactions on Signal Processing*, vol. 67, no. 8, pp. 2052–2065, 2019.
- [17] L. Wan, K. Liu, Y.-C. Liang, and T. Zhu, "DOA and polarization estimation for non-circular signals in 3-D millimeter wave polarized massive MIMO systems," *IEEE Transactions on Wireless Communications*, vol. 20, no. 5, pp. 3152–3167, 2021.
- [18] W. Zheng, X. Zhang, Y. Wang, M. Zhou, and Q. Wu, "Extended coprime array configuration generating large-scale antenna co-array in massive MIMO system," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 7841–7853, 2019.
- [19] F. Wen, J. Shi, and Z. Zhang, "Closed-form estimation algorithm for EMVS-MIMO radar with arbitrary sensor geometry," *Signal Processing*, vol. 186, p. 108117, 2021.
- [20] X. Wang, X. Wang, and X. Lin, "Co-prime array processing with sum and difference co-array," in *2015 49th Asilomar Conference on Signals, Systems and Computers*, pp. 380–384, 2015.
- [21] X. Wang, Z. Chen, S. Ren, and S. Cao, "DOA estimation based on the difference and sum coarray for coprime arrays," *Digital Signal Processing*, vol. 69, pp. 22–31, 2017.
- [22] W. Si, Z. Peng, C. Hou, and F. Zeng, "Improved nested arrays with sum-difference coarray for DOA estimation," *IEEE Sensors Journal*, vol. 19, no. 16, pp. 6986–6997, 2019.
- [23] P. Zhao, K. Wang, G. Hu, and L. Wan, "Underdetermined DOA estimation using unfold coprime array from the perspective of sum-difference co-array," *IEEE Access*, vol. 7, pp. 168557–168564, 2019.
- [24] A. Alexiou and A. Manikas, "Investigation of array robustness to sensor failure," *Journal of the Franklin Institute*, vol. 342, no. 3, pp. 255–272, 2005.
- [25] C.-L. Liu and P. P. Vaidyanathan, "Robustness of difference coarrays of sparse arrays to sensor failures-Part I: a theory

- motivated by coarray MUSIC,” *IEEE Transactions on Signal Processing*, vol. 67, no. 12, pp. 3213–3226, 2019.
- [26] C.-L. Liu and P. P. Vaidyanathan, “Robustness of difference coarrays of sparse arrays to sensor failures-Part II: array geometries,” *IEEE Transactions on Signal Processing*, vol. 67, no. 12, pp. 3227–3242, 2019.