Novel Algorithms for Analyzing the Robustness of Difference Coarrays to Sensor Failures

Chun-Lin Liu[a,b,*], P. P. Vaidyanathan[c]

[a]Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan 10617
[b]Graduate Institute of Communication Engineering, National Taiwan University, Taipei, Taiwan 10617
[c]Department of Electrical Engineering, MC 136-93, California Institute of Technology, Pasadena, CA 91125, USA

Abstract

Sparse arrays have drawn attention because they can identify $\mathcal{O}(N^2)$ uncorrelated source directions using N physical sensors, whereas uniform linear arrays (ULA) find at most $N-1$ sources. The main reason is that the difference coarray, defined as the set of differences between sensor locations, has size of $\mathcal{O}(N^2)$ for some sparse arrays. However, the performance of sparse arrays may degrade significantly under sensor failures. In the literature, the k-essentialness property characterizes the patterns of k sensor failures that change the difference coarray. Based on this concept, the k-essential family, the k-fragility, and the k-essential Sperner family provide insights into the robustness of arrays. This paper proposes novel algorithms for computing these attributes. The first algorithm computes the k-essential Sperner family without enumerating all possible k-essential subarrays. With this information, the second algorithm finds the k-essential family first and the k-fragility next. These algorithms are applicable to any 1-D array. However, for robust array design, fast computation for the k-fragility is preferred. For this reason, a simple expression associated with

[⋆]This work was supported in part by the ONR grant N00014-18-1-2390, in part by the NSF grant CCF-1712633, in part by the California Institute of Technology, in part by the Ministry of Education, Taiwan, under Grant Numbers NTU-107V0902 and NTU-108V0902, in part by the Ministry of Science and Technology, Taiwan, under Grant Number MOST 108-2218-E-002-043-MY2, and in part by the National Taiwan University.

[*]Corresponding author

Email addresses: chunlinliu@ntu.edu.tw (Chun-Lin Liu), ppvnath@systems.caltech.edu (P. P. Vaidyanathan)

the k-essential Sperner family is proposed to be a tighter lower bound for the k-fragility than the previous result. Numerical examples validate the proposed algorithms and the presented lower bound.
Keywords: Sparse arrays, difference coarrays, essentialness, fragility, numerical algorithms.

1. Introduction

Array signal processing [12, 14, 35] has been central to many fields in science and engineering, such as communication [8], radar [9, 31], imaging [10], and radio astronomy [5]. In these applications, sparse arrays, where the sensing elements are placed nonuniformly, have recently drawn attention [10, 20, 24, 34]. Some sparse arrays can resolve $\mathcal{O}(N^2)$ uncorrelated sources using N physical sensors, while the ULA identifies at most $N-1$ uncorrelated sources with N sensors. For instance, these sparse arrays include the minimum redundancy arrays (MRA) [20], the nested arrays [24], the coprime arrays [34], and their generalizations [28]. The main reason why $\mathcal{O}(N^2)$ uncorrelated sources are resolvable, is that the difference coarray (the set of differences between sensor locations), has an $\mathcal{O}(N^2)$-long central ULA segment. With this concept, the direction-of-arrival (DOA) of the sources can be estimated by analyzing the samples on the difference coarray [2, 15, 24, 26, 27, 34, 39], and these methods were shown to resolve more uncorrelated sources than sensors.

In the literature, two categories of DOA estimators are reported for resolving more sources than sensors for sparse arrays. The first type of DOA estimators explicitly converts the array measurements onto the difference coarray, to which the MUltiple SIgnal Classification (MUSIC) [30] algorithm can be applied. Algorithms of this type, for example, include the coarray MUSIC algorithm [2, 15, 27], and the spatial smoothing MUSIC [24, 25]. The other family reformulates the DOA estimation as a sparse recovery problem associated with a dense grid of the DOA. Then popular sparse recovery techniques such as ℓ_1 minimization and LASSO can be used for DOA estimation [26, 28].

In practice, *sensor failure* may lead to performance degradation or even breakdown of the overall system [13, 22]. *Empirical results* showed that, for some sparse arrays, such as MRA, even one faulty sensor could shrink the $\mathcal{O}(N^2)$-long ULA segment in the difference coarray significantly. Furthermore, small ULA segments in the difference coarray typically degrade the estimation performance [2, 15, 24, 37]. Therefore, sparse arrays were usually considered not to be robust to sensor failure. However, the impact of faulty sensors on sparse arrays has to be analyzed rigorously, since these observations depend on array configurations.

In array signal processing, sensor failures have been handled with two approaches: 1) developing new algorithms that are functional under sensor failures [21, 23, 36, 38] and 2) analyzing the robustness of array configurations [3, 6, 18, 19]. In this paper, we will focus on the second approach, with an emphasis on the robustness of the difference coarray, since the difference coarray plays a crucial role in the applicability of some coarray-based DOA estimators in [2, 15, 24, 25, 27].

This topic was recently investigated in [18, 19]. To begin with, a sensor is said to be *essential* if its deletion changes the difference coarray. A generalization of this (the k-essentialness property and k-essential subarrays), is then defined in order to study the effect of multiple sensor failures on the difference coarray. The k-essential family is defined as a family of k-essential subarrays. Then, the robustness of the difference coarray is quantified using the notion of k-fragility, defined as the ratio of the number of k-essential subarrays to the number of all subarrays of size k. This quantity ranges from 0 to 1 while an array is more robust if the fragility is closer to 0. Finally, the k-essential Sperner family serves as a compact representation of the k-essential family. These attributes have been studied in [19] for array configurations such as the ULA, the MRA [20], the nested array [24], and the coprime array [34], to name a few. Nevertheless, it is more involved to extend these theoretical analyses to arbitrary sparse arrays, and hence numerical algorithms for the k-essential family, the k-fragility, and the k-essential Sperner family are of considerable interest.

This paper presents novel numerical algorithms for finding the k-essential

Sperner family, the k-essential family, and the k-fragility recursively. The novelty of these algorithms is as follows. It will be observed that the search space of the k-essential Sperner family is smaller than that of the k-essential family. This property is incorporated in the algorithms for a fast computation of the k-essential Sperner family. Once the k-essential Sperner family is available, the k-essential family and the k-fragility can be obtained. For the reader's convenience, sample MATLAB codes of these algorithms can be found in [1].

However, the above-mentioned approach involves the enumeration of the k-essential family before the k-fragility is obtained. For applications such as robust sparse array design [17], it is of interest to compute the k-fragility directly. For this reason, we also provide a lower bound for the k-fragility which simply depends on the cardinality of the k-essential Sperner family. This lower bound is simple to compute and more importantly, is tighter than the previous result in [18].

The outline of this paper is as follows. Section 2.1 reviews the data model of array signal processing and the difference coarrays. Section 2.2 reviews the theory of the k-essentialness property, the k-essential family, the k-fragility, and the k-essential Sperner family. These results have been proposed in [18, 19]. Section 3 proposes novel algorithms for computing the k-essential Sperner family, the k-essential family, and the k-fragility. Section 4 presents a lower bound for the k-fragility. In Section 5, the proposed algorithms and the lower bound are demonstrated through numerical examples. Section 6 concludes this paper.

2. Preliminaries

2.1. The Data Model

Consider D monochromatic sources with common wavelength λ illuminating a one-dimensional (1-D) sensor array. The sensors are located at $n\lambda/2$, where n belongs to an integer set $\mathbb{S}$. The complex amplitude and the DOA of the ith source are denoted by A_i and $\theta_i \in [-\pi/2, \pi/2]$, respectively. The measurement vector on $\mathbb{S}$, denoted by $\mathbf{x}_{\mathbb{S}}$, can be modeled as

$$\mathbf{x}_{\mathbb{S}} = \sum_{i=1}^{D} A_i \mathbf{v}_{\mathbb{S}}(\bar{\theta}_i) + \mathbf{n}_{\mathbb{S}}, \tag{1}$$

where $\mathbf{n}_{\mathbb{S}}$ is the additive noise term and $\mathbf{v}_{\mathbb{S}}(\bar{\theta}_i)$ is the steering vector with respect to the normalized DOA $\bar{\theta}_i \triangleq (\sin\theta_i)/2$. In particular, if the integer set $\mathbb{S} = \{n_1, n_2, \ldots, n_N\}$ with $n_1 < n_2 < \cdots < n_N$, then the steering vector is defined as

$$\mathbf{v}_{\mathbb{S}}(\bar{\theta}_i) \triangleq \begin{bmatrix} e^{j2\pi\bar{\theta}_i n_1} & e^{j2\pi\bar{\theta}_i n_2} & \ldots & e^{j2\pi\bar{\theta}_i n_N} \end{bmatrix}^T. \tag{2}$$

With regard to (1), we assume that the set $\mathbb{S}$ and the number of sources D are fixed and known, while the normalized DOAs $\bar{\theta}_i$ are fixed but *unknown*. We assume that A_i and $\mathbf{n}_{\mathbb{S}}$ are zero-mean and uncorrelated. In other words, if $\mathbf{s} \triangleq \begin{bmatrix} A_1 & A_2 & \ldots & A_D & \mathbf{n}_{\mathbb{S}}^T \end{bmatrix}^T$, then we have

$$\mathrm{E}[\mathbf{s}] = \mathbf{0}, \qquad \mathrm{E}\left[\mathbf{s}\mathbf{s}^H\right] = \begin{bmatrix} p_1 & 0 & \ldots & 0 & \mathbf{0} \\ 0 & p_2 & \ldots & 0 & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & p_D & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} & p_n\mathbf{I} \end{bmatrix}. \tag{3}$$

Here p_i and p_n are the powers of the ith source and the noise, respectively. They are assumed to be fixed but unknown. The notation $\mathrm{E}[\cdot]$ is the expectation operator. Based on these assumptions, the covariance matrix of $\mathbf{x}_{\mathbb{S}}$ is given by

$$\mathbf{R}_{\mathbb{S}} = \sum_{i=1}^{D} p_i \mathbf{v}_{\mathbb{S}}(\bar{\theta}_i)\mathbf{v}_{\mathbb{S}}^H(\bar{\theta}_i) + p_n\mathbf{I}. \tag{4}$$

Next, let us define the difference coarray of the array $\mathbb{S}$ as follows [15, 20, 24, 34]

Definition 1 (Difference coarray)**.** *Let $\mathbb{S}$ be an integer set. The difference coarray is defined as the set of differences between the elements in $\mathbb{S}$. More specifically,*

$$\mathbb{D} \triangleq \{n_1 - n_2 : n_1, n_2 \in \mathbb{S}\}. \tag{5}$$

Then we can construct the autocorrelation vector $\mathbf{x}_{\mathbb{D}}$ by vectorizing $\mathbf{R}_{\mathbb{S}}$ and averaging duplicating elements

$$\mathbf{x}_{\mathbb{D}} = \sum_{i=1}^{D} p_i \mathbf{v}_{\mathbb{D}}(\bar{\theta}_i) + p_n \mathbf{e}_0, \tag{6}$$

where $\mathbf{v}_{\mathbb{D}}(\bar{\theta}_i)$ denotes the steering vector on the difference coarray and the vector $\mathbf{e}_0$ is given by

$$\mathbf{e}_0 = [\ \underbrace{0 \quad 0 \quad \ldots \quad 0}_{(|\mathbb{D}|-1)/2} \quad 1 \quad \underbrace{0 \quad 0 \quad \ldots \quad 0}_{(|\mathbb{D}|-1)/2}\ ]^T. \tag{7}$$

The notation $|\cdot|$ denotes the cardinality of a set. Note that according to Definition 1, the element 0 belongs to $\mathbb{D}$. Furthermore, $\mathbb{D}$ is evenly symmetric. More specifically, $m \in \mathbb{D}$ if and only if $-m \in \mathbb{D}$. Therefore, $|\mathbb{D}|$ is an odd number and $(|\mathbb{D}| - 1)/2$ is an integer.

Eq. (6) can be regarded as the measurement vector defined on the difference coarray $\mathbb{D}$. In particular, if we replace the physical array $\mathbb{S}$ with the difference coarray $\mathbb{D}$, the source amplitude A_i with the source power p_i, and the noise term $\mathbf{n}_{\mathbb{S}}$ with the vector $p_n\mathbf{e}_0$, then the array measurement on the physical array (1) becomes that on the difference coarray (6). This observation also has led to a vast development of sparse arrays in the last few years. For instance, for certain array configurations, quite a few DOA estimators on the difference coarray were proposed to achieve higher resolution than DOA estimators based on the physical array [2, 25, 27, 33, 39]. Furthermore, since the size of the difference coarray could be $\mathcal{O}(N^2)$, where N is the number of physical sensors, it is possible to resolve more uncorrelated sources than sensors [2, 25, 27, 33, 39].

Given a difference m, there could be multiple sensor pairs (n_1, n_2) with $n_1 - n_2 = m$. The number of such pairs is characterized by the weight function $w(m)$:

Definition 2. *The weight function $w(m)$ is defined as the number of sensor pairs with separation $m \in \mathbb{Z}$. This definition can be written as $w(m) = \left|\{(n_1, n_2) \in \mathbb{S}^2 : n_1 - n_2 = m\}\right|$.*

By definition, the weight function $w(m)$ is integer-valued and its support (the set of m's such that $w(m) \neq 0$) is exactly the difference coarray [24]. It can be shown that the weight function is an even function, i.e. $w(-m) = w(m)$. Furthermore, for a 1-D sensor array $\mathbb{S}$ with N physical sensors, the weight function satisfies [24]

$$w(0) = N, \qquad w(\max(\mathbb{S}) - \min(\mathbb{S})) = 1, \qquad \sum_{m \in \mathbb{D}} w(m) = N^2, \tag{8}$$

where $\max(\mathbb{S})$ and $\min(\mathbb{S})$ denote the maximum and the minimum of the set $\mathbb{S}$, respectively.

Next we will define *holes*, which are important in analyzing the performances achieved by the difference coarray [34]. We say an integer h is a hole of the difference coarray $\mathbb{D}$ if $\min(\mathbb{D}) \leq h \leq \max(\mathbb{D})$ but $h \notin \mathbb{D}$.

The central ULA segment of the difference coarray, denoted by the set $\mathbb{U}$, is the largest segment centered around 0 that consists of consecutive integers. In particular, $\mathbb{U} \triangleq \{m : \{0, \pm 1, \ldots, \pm m\} \subseteq \mathbb{D}\}$. The difference coarray is *hole-free* if there are no holes in the difference coarray, which is equivalent to the property that $\mathbb{D} = \mathbb{U}$. Note that the central ULA segment $\mathbb{U}$ plays a significant role in DOA estimators such as spatial smoothing MUSIC [24, 25] and coarray MUSIC [2, 15, 27].

Example 1. *Let us consider a sensor array with* $\mathbb{S} = \{0, 6, 7, 9, 11, 19\}$*, as depicted on the top of Fig. 1. The difference coarray* $\mathbb{D}$*, as defined in Definition 1, is also illustrated in Fig. 1. It can be observed that the sizes of* $\mathbb{S}$ *and* $\mathbb{D}$ *are* 6 *and* 29*, respectively. The holes of the difference coarray are* ± 14*,* ± 15*,* ± 16*,* ± 17*, and* ± 18*, while the central ULA segment of the difference coarray contains consecutive integers from* -13 *to* 13*. The weight function of* $\mathbb{S}$ *is shown on the bottom of Fig. 1. In particular, we have* $w(0) = 6$*,* $w(19) = 1$*, and* $\sum_{m \in \mathbb{D}} w(m) = 36$*, which are consistent with the property in* (8)*.*

The difference coarray $\mathbb{D}$ and the weight function $w(m)$ are utilized to derive optimal sparse array configurations. For instance, the minimum redundancy array (MRA) with N sensors has the largest hole-free difference coarray [20]. The

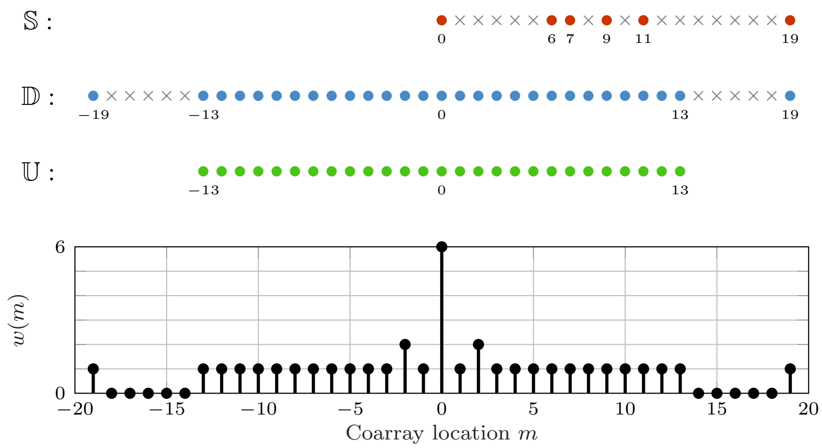

Figure 1: An illustration of the physical array $\mathbb{S}$ (red dots), the difference coarray $\mathbb{D}$ (blue dots), the central ULA segment of the difference coarray $\mathbb{U}$ (green dots), and the weight function $w(m)$ (black).

minimum hole array (MHA) has the smallest number of holes subject to the constraint that each nonzero difference originates from a unique sensor pair [4]. Both the MRA and the MHA have difference coarrays of size $\mathcal{O}(N^2)$. However, finding the sensor locations of the MRA and the MHA is computationally intractable for large number of sensors [4, 11, 20, 29].

Recently, this issue has been addressed by sparse arrays with large difference coarrays. In particular, with N physical sensors, the size of the difference coarray is up to $\mathcal{O}(N^2)$. More importantly, these sparse arrays have closed-form expressions for the sensor locations, which are simple to compute. The arrays with these properties include the nested array [24], the coprime array [34] the generalized coprime array [28], and the super nested array [16], to name a few.

2.2. Robustness of Difference Coarrays of Sparse Arrays

The structure of the difference coarray plays an important role in the applicability of many coarray-based DOA estimators, such as the coarray MUSIC algorithm [2, 15, 24, 25, 27]. In [18], the influence of a sensor failure on the difference coarray is described by the essentialness property:

Definition 3. *The sensor at $n \in \mathbb{S}$ is essential if the removal of that sensor changes the difference coarray.*

To be more specific, let n belong to $\mathbb{S}$, and $\overline{\mathbb{S}} \triangleq \mathbb{S}\backslash\{n\}$ be the array configuration after the failure of the sensor at n. The difference coarrays of $\mathbb{S}$ and $\overline{\mathbb{S}}$ are denoted by $\mathbb{D}$ and $\overline{\mathbb{D}}$, respectively. We say the sensor at n is essential if $\overline{\mathbb{D}} \neq \mathbb{D}$. In addition, n is said to be *inessential* if $\overline{\mathbb{D}} = \mathbb{D}$.

The essentialness property can be utilized to quantify the robustness of array configurations. For instance, the more essential sensors in an array, the more likely the difference coarray changes under sensor failures. In this scenario, coarray MUSIC is more likely to fail. Thus the array is not robust.

The essentialness property is also useful in assigning sensing devices of different reliability and cost to different locations. For example, suppose there are two sensing devices, Device I and Device II, with different costs and qualities. Device I is costly but is durable, i.e. with low failure probability. Device II is less expensive but is easily broken. Therefore, to strike a balance between the cost and the robustness of an array, Device I can be used as an essential sensor, and Device II as an inessential sensor.

The essentialness property can also be utilized to assess the economy of an array [18]:

Definition 4. *A sensor array $\mathbb{S}$ is said to be* maximally economic *if all the sensors in $\mathbb{S}$ are essential.*

These arrays are also called maximally economic sparse arrays (MESA). It was proved in [19, Theorem 1] that the MESA family includes the MRA, the MHA, parts of the nested array, and the Cantor array. However, the ULA and the coprime arrays are not maximally economic [19].

The essentialness property is closely related to one sensor failure in an array. However, multiple sensor failures are more realistic. The k-essentialness property, a generalization of Definition 3, is defined for this purpose [18]:

Definition 5. *A subarray $\mathbb{A}$ of $\mathbb{S}$ is said to be k-essential with respect to an array $\mathbb{S}$ if it has the following properties.*

1. $\mathbb{A}$ *has size exactly* k.
2. *The difference coarray changes when* $\mathbb{A}$ *is removed from* $\mathbb{S}$.

Note that the k-essentialness is an attribute of a subarray $\mathbb{A}$ of $\mathbb{S}$, while the essentialness is an attribute of a sensor at n in $\mathbb{S}$. In particular, the essentialness of $n \in \mathbb{S}$ is equivalent to the k-essentialness of $\{n\} \subseteq \mathbb{S}$ with $k = 1$. The collection of the k-essential subarrays constitutes the k-essential family [18]:

Definition 6. *The* k*-essential family* $\mathcal{E}_k$ *with respect to a sensor array* $\mathbb{S}$ *is defined as*

$$\mathcal{E}_k \triangleq \{\mathbb{A} : \mathbb{A} \text{ is } k\text{-essential with respect to } \mathbb{S}\}. \tag{9}$$

Here $k = 1, 2, \ldots, |\mathbb{S}|$.

The larger the size of $\mathcal{E}_k$ is, the more likely that the difference coarray tends to change under k faulty sensors. This concept leads to the definition of the k-fragility [18]:

Definition 7. *The fragility or* k*-fragility of a sensor array* $\mathbb{S}$ *is defined as*

$$F_k \triangleq \frac{|\mathcal{E}_k|}{\binom{|\mathbb{S}|}{k}}, \tag{10}$$

where $k = 1, 2, \ldots, |\mathbb{S}|$.

The k-fragility can be regarded as a scalar attribute to quantify the robustness of an array, in the sense that the difference coarray changes under k sensor failures. The larger F_k is, the more fragile (or the less robust) the array is.

The following present some properties regarding the k-fragility F_k [18]:

Theorem 1. *Let* $\mathbb{S}$ *be an integer set denoting the sensor locations. The* k*-fragility* F_k *with respect to* $\mathbb{S}$ *has the following properties:*

1. $F_k \leq F_{k+1}$ *for all* $1 \leq k \leq |\mathbb{S}|-1$. *The equality holds if and only if* $F_k = 1$.

2. $F_k = 1$ *for all* k *such that* $Q \leq k \leq |\mathbb{S}|$, *where* $Q = \min\{Q_1, Q_2\}$. *The parameters* Q_1 *and* Q_2 *are given by*

$$Q_1 = |\mathbb{S}| - |\mathcal{E}_1| + 1, \tag{11}$$

$$Q_2 = \left\lceil |\mathbb{S}| - \frac{\sqrt{8|\mathbb{S}| - 11} + 1}{2} \right\rceil, \qquad \text{for } |\mathbb{S}| \geq 2. \tag{12}$$

3. *Let* $F_{\min}$ *denote* $\min\{1, 2/|\mathbb{S}|\}$. *Then* $F_{\min} \leq F_k \leq 1$ *for all* $1 \leq k \leq |\mathbb{S}|$.

Theorem 1 imposes some constraints on the k-fragility F_k. Property 1 indicates that F_k is an increasing function, which is consistent with the concept that, as the number of faulty sensors increases, the difference coarray tends to change. Property 2 is equivalently saying that, if there are many faulty sensors (i.e. $k \geq Q$), then the difference coarray changes. Property 3 shows that F_k is bounded between $\min\{1, 2/|\mathbb{S}|\}$ and 1, which provides a numerical range for comparing the robustness of arrays.

It may be computationally difficult to find $\mathcal{E}_k$ and F_k for any array $\mathbb{S}$. The reason is that, first there are as many as $\binom{|\mathbb{S}|}{k}$ subarrays of size k. Second, for each subarray, we need to compare the corresponding difference coarray $\overline{\mathbb{D}}$ and then compare it with the original difference coarray $\mathbb{D}$, as in Definition 5. These steps become computationally difficult for large $|\mathbb{S}|$ and $k \approx |\mathbb{S}|/2$. Furthermore, the memory storage of $\mathcal{E}_k$ is also challenging, due to the large size of $\mathcal{E}_k$.

To mitigate these issues, the k-essential Sperner family was proposed in [18] as a compact representation of $\mathcal{E}_k$. The formal definition is as follows [18]:

Definition 8. *Let* $\mathcal{E}_k$ *be the* k*-essential family with respect to the array* $\mathbb{S}$*, where the integer* k *satisfies* $1 \leq k \leq |\mathbb{S}|$. *The* k*-essential Sperner family* $\mathcal{E}'_k$ *is defined as follows:*

$$\mathcal{E}'_k = \begin{cases} \mathcal{E}_1, & \text{if } k = 1, \\ \{\mathbb{A} \in \mathcal{E}_k : \forall \mathbb{B} \in \mathcal{E}_{k-1},\ \mathbb{B} \not\subset \mathbb{A}\}, & \text{if } k = 2, \ldots, |\mathbb{S}|. \end{cases} \tag{13}$$

where $\mathbb{B} \not\subset \mathbb{A}$ *denotes that* $\mathbb{B}$ *is not a proper subset of* $\mathbb{A}$.

The k-essential Sperner family with $k = 1$ is defined as the k-essential family with $k = 1$. For $2 \leq k \leq |\mathbb{S}|$, the k-essential Sperner family extracts the portions of the $\mathcal{E}_k$ which are not supersets of any elements in $\mathcal{E}_{k-1}$. By doing so, it was demonstrated in [18] that the size of the k-essential Sperner family $\mathcal{E}'_k$ can be much smaller than that of the k-essential family $\mathcal{E}_k$. This attribute makes it possible to reduce the computational complexity, as we shall elaborate in Section 3.

The term *Sperner* stems from the *Sperner family* in discrete mathematics [7, 32]. A Sperner family is a family of sets in which none of the elements is a subset of the other [7, 32]. Based on Definition 8, it can be shown that $\mathcal{E}'_k$ themselves form a Sperner family [19].

The k-essential family can be uniquely recovered from the k-essential Sperner family. More specifically, given the complete information of $\mathcal{E}'_1, \mathcal{E}'_2, \ldots, \mathcal{E}'_k$, the k-essentialness property can be readily verified by examining the inclusion between sets, as in the following lemma [18]:

Lemma 1. *Let $\mathcal{E}'_k$ be the k-essential Sperner family of $\mathbb{S}$ with $1 \leq k \leq |\mathbb{S}|$. Then the k-essential family $\mathcal{E}_k$ satisfies*

$$\mathcal{E}_k = \begin{cases} \mathcal{E}'_1, & \text{if } k = 1, \\ \{\mathbb{A} \cup \mathbb{B} : \mathbb{A} \in \mathcal{E}'_\ell,\ 1 \leq \ell \leq k, \\ \qquad \mathbb{B} \subseteq \mathbb{S} \backslash \mathbb{A},\ |\mathbb{B}| = k - \ell\}, & \text{otherwise.} \end{cases}$$

Lemma 1 is particularly useful in numerically evaluating the k-essential family $\mathcal{E}_k$ and the k-fragility F_k. Given the k-essential Sperner family $\mathcal{E}'_k$ for all k, first $\mathcal{E}_k$ can be constructed and the F_k can be found. Another advantage is that if there are only few sensor failures, for example $k = 1, 2, \ldots, K_{\max}$, then Lemma 1 can be applied recursively for these k, which alleviates the computational burden of finding all the $\mathcal{E}_k$'s.

The k-fragility can be utilized in designing new sparse array configurations with enhanced robustness. For instance, the robust minimum redundancy array (RMRA) [17] has the largest hole-free difference coarray and the property that $F_1 = F_{\min}$. Here $F_{\min}$ is given in Property 3 of Theorem 1. It was shown that RMRA has the property that $|\mathbb{D}| = \mathcal{O}(|\mathbb{S}|^2)$, which is as good as the MRA.

Furthermore, RMRA with N sensors is more robust than MRA with N sensors, where the fragility F_1 is $2/N$ for RMRA and 1 for MRA.

3. Numerical Algorithms

3.1. The k-Essential Sperner Family $\mathcal{E}'_k$

In general, it is very involved to derive closed-form expressions for the k-essential family, the k-fragility, and the k-essential Sperner family [19]. For arbitrary array configurations, closed forms of $\mathcal{E}'_k$ and F_k for all k are not available, except for MESA and coprime arrays [19]. Fortunately however, it is still possible to *compute* the k-essential family, the k-fragility, and the k-essential Sperner family *numerically*.

A straightforward approach to the numerical algorithm is as follows. For a given k in $1 \leq k \leq |\mathbb{S}|$, we first enumerate all $\binom{|\mathbb{S}|}{k}$ subarrays to find $\mathcal{E}_k$. For each subarray, the k-essentialness property is examined, which requires the computation of $\overline{\mathbb{D}}$ and a comparison between $\mathbb{D}$ and $\overline{\mathbb{D}}$ (Definitions 3 and 5). Once $\mathcal{E}_k$ is known, the k-fragility F_k and the k-essential Sperner family $\mathcal{E}'_k$ can be obtained according to Definitions 7 and 8, respectively.

A drawback of this straightforward method is the large search space for $\mathcal{E}_k$. In principle, the k-essentialness property needs to be examined over $\binom{|\mathbb{S}|}{k}$ subarrays. However, if we focus on the k-essential Sperner family first, the search space could be much smaller. Therefore, in this paper, we will first propose a method to finding $\mathcal{E}'_k$ *without the complete information of* $\mathcal{E}_k$.

The complete algorithm for the k-essential Sperner family is summarized in Algorithm 1, whose objective is to compute $\mathcal{E}'_k$ for $k = 1, 2, \ldots, K_{\max}$. Here $K_{\max}$ is a pre-defined integer in the range $1 \leq K_{\max} \leq |\mathbb{S}|$. This parameter indicates that sensor failures up to size $K_{\max}$ are of interest.

Algorithm 1 is divided into two stages. The first stage is to compute $\mathcal{E}'_1$ while the second stage is to find $\mathcal{E}'_k$ for $k \geq 2$. In the first stage, the essentialness property of all the sensors in $\mathbb{S}$ is examined, as shown in lines 3 to 7 in Algorithm 1.

Algorithm 1 The k-essential Sperner family $\mathcal{E}'_k$

Require: The physical array $\mathbb{S}$

Require: An integer $K_{\max}$ with $1 \le K_{\max} \le |\mathbb{S}|$

```
function k-ESSENTIAL-SPERNER-FAMILY($\mathbb{S}$, $K_{\max}$)
    $\mathcal{E}'_k \leftarrow \varnothing$ for $k = 1, 2, \ldots, K_{\max}$
    for all $n \in \mathbb{S}$ do
        if ($n$ is essential) then
            $\mathcal{E}'_1 \leftarrow \mathcal{E}'_1 \cup \{n\}$
        end if
    end for
    $\mathbb{I} \leftarrow$ the set of inessential sensors in $\mathbb{S}$
    $Q \leftarrow \min(Q_1, Q_2)$                          ▷ Equations (11) and (12)
    for $k \leftarrow 2, 3, \ldots, \min(K_{\max}, Q)$ do
        for all $\mathbb{A} \subseteq \mathbb{I}$ and $|\mathbb{A}| = k$ do
            if ($\mathbb{A}$ is a superset of some $\mathbb{B} \in \mathcal{E}'_\ell$ for $2 \le \ell \le k-1$) then
                Continue                                   ▷ Skip to another $\mathbb{A}$
            else if ( $\mathbb{A}$ is $k$-essential ) then
                $\mathcal{E}'_k \leftarrow \mathcal{E}'_k \cup \{\mathbb{A}\}$
            end if
        end for
    end for
    return $\mathcal{E}'_1, \mathcal{E}'_2, \ldots, \mathcal{E}'_{K_{\max}}$
end function
```

The second stage focuses on computing $\mathcal{E}'_k$ for $k = 2, \ldots, K_{\max}$. It is not necessary to search over all possible subarrays $\mathbb{A} \subseteq \mathbb{S}$ of size k due to the following reasons.

1. It suffices to compute $2 \leq k \leq \min\{K_{\max}, Q\}$. It was shown in [18, Lemma 8] that $\mathcal{E}'_k$ is empty if $k > Q = \min\{Q_1, Q_2\}$. Here Q_1 and Q_2 are given in (11) and (12), respectively. By definition, Q_1 and Q_2 can be readily computed from the number of sensors $|\mathbb{S}|$ and the number of essential sensors $|\mathcal{E}'_1|$.
2. We only need to consider all subarrays consisting of *inessential sensors* in finding $\mathcal{E}'_k$. The reason is as follows. Suppose $\mathbb{A} \triangleq \{n_1, n_2, \ldots, n_k\} \subseteq \mathbb{S}$ is a subarray of size k and without loss of generality, $n_1 \in \mathbb{S}$ is essential. By definition, we have $\{n_1\} \in \mathcal{E}_1$, $\{n_1, n_2\} \in \mathcal{E}_2$, $\{n_1, n_2, \ldots, n_{k-1}\} \in \mathcal{E}_{k-1}$. Since $\{n_1, n_2, \ldots, n_{k-1}\} \subset \mathbb{A}$, the set $\mathbb{A}$ does not belong to $\mathcal{E}'_k$, due to Definition 8. As a result, n_1 is inessential with respect to $\mathbb{S}$. This chain of arguments shows that the set $\mathbb{A}$ consists of inessential sensors.
3. It is not necessary to compute the difference coarrays for all the subarrays in Reason 2. Due to Lemma 1, the k-essential subarrays can be expressed as $\mathbb{A} \cup \mathbb{B}$, where $\mathbb{A} \in \mathcal{E}'_\ell$ for some $1 \leq \ell \leq k$. In other words, if a subarray of size k is a superset of some ℓ-essential subarray, then that subarray is k-essential. Therefore, we can use the contents of $\mathcal{E}'_\ell$ to accelerate the computation.

These points are also taken into consideration in lines 10, 11, and 12 of Algorithm 1.

If the k-fragility with $k = 1$ is close to 1, Algorithm 1 is much faster than computing the k-essential family directly. The reason is that in Algorithm 1, the subarrays are enumerated over the inessential sensors (line 11), which are very few when F_1 is close to 1. In particular, if an array is maximally economic, Algorithm 1 is able to detect the scenario in line 8 and then skip the enumeration in lines 10 to 18. This mechanism reduces the computational time significantly for MESA.

Algorithm 2 The k-essential family $\mathcal{E}_k$ and the k-fragility F_k

Require: The physical array $\mathbb{S}$

Require: An integer $K_{\max}$ with $1 \leq K_{\max} \leq |\mathbb{S}|$

Require: The k-essential Sperner family $\mathcal{E}'_1, \mathcal{E}'_2, \ldots, \mathcal{E}'_{K_{\max}}$

function k-ESSENTIAL-FAMILY-FRAGILITY($\mathbb{S}$, $K_{\max}$, $\mathcal{E}'_1, \mathcal{E}'_2, \ldots, \mathcal{E}'_{K_{\max}}$)
  $\mathcal{E}_1 \leftarrow \mathcal{E}'_1$
  $F_1 \leftarrow |\mathcal{E}_1|/|\mathbb{S}|$
  $Q \leftarrow \min(Q_1, Q_2)$ ▷ Equations (11) and (12)
  for $k \leftarrow 2, 3, \ldots, K_{\max}$ **do**
    if $(k < Q)$ **then**
      for all $\mathbb{A} \subseteq \mathbb{S}$ and $|\mathbb{A}| = k$ **do**
        if ($\mathbb{A}$ is a superset of some $\mathbb{B} \in \mathcal{E}'_\ell$ for $1 \leq \ell \leq k$) **then**
          $\mathcal{E}_k \leftarrow \mathcal{E}_k \cup \{\mathbb{A}\}$
        end if
      end for
    else
      $\mathcal{E}_k \leftarrow$ the family of all subarrays of $\mathbb{S}$ of size k
    end if
    $F_k \leftarrow |\mathcal{E}_k|/\binom{|\mathbb{S}|}{k}$
  end for
  return $\mathcal{E}_1, \mathcal{E}_2, \ldots, \mathcal{E}_{K_{\max}}, F_1, F_2, \ldots, F_{K_{\max}}$
end function

Next we will discuss the algorithm for computing the k-essential family and the k-fragility, as shown in Algorithm 2. Suppose the physical array $\mathbb{S}$ and an integer $K_{\max}$ with $1 \leq K_{\max} \leq |\mathbb{S}|$ are given. The k-essential Sperner family is first evaluated by using Algorithm 1, and then the k-essential family and the k-fragility can be obtained.

Due to the properties of the k-essential family, the following two points are used to accelerate the computation:

1. The k-essential family $\mathcal{E}_k$ has all the subarrays of size k for $k \geq Q$ [19, Theorem 1].
2. According to Lemma 1, the k-essential family comprises the supersets of the elements in the k-essential Sperner family.

These points are in lines 6 to 14 of Algorithm 2. Finally, the k-fragility F_k can be obtained from the sizes of $\mathcal{E}_k$, as in line 15 of Algorithm 2.

Summarizing, the proposed algorithms (Algorithms 1 and 2) are able to find the k-essential Sperner family, the k-essential family, and the k-fragility of any 1-D array configuration. In Algorithm 1, the essential sensors are found first, and then the k-essential Sperner family $\mathcal{E}'_k$ for $k \geq 2$ is constructed from the inessential sensors. Based on the results of Algorithm 1, Algorithm 2 finds the k-essential family and the k-fragility, by comparing the inclusion between sets. These algorithms are more flexible than the closed forms in [19], which are only applicable to certain array configurations.

4. Lower Bounds for the k-Fragility

In the design of sparse arrays with enhanced robustness, it is of primary interest to specify the level of robustness in terms of the k-fragility, rather than the explicit expressions for the k-essential family $\mathcal{E}_k$ [17]. As a result, in these applications, it is redundant to first construct the k-essential family in line 13 of Algorithm 2, and then to determine the size the k-essential family in line 15 of Algorithm 2.

This issue can be addressed by studying the *lower bounds* for F_k. These bounds have to be tight and simple to compute. A candidate for the lower bounds is $F_{\min}$ in Property 3 of Theorem 1. This quantity is simple to compute. However, it is a loose bound, since only the information of the number of sensors is utilized in $F_{\min}$.

In principle, as long as we have the information of the k-essential Sperner family, it is possible to derive a lower bound for the k-fragility which can be readily computed and tighter than $F_{\min}$. The following lemma states the improved lower bound L_k associated with the sizes of the k-essential Sperner family:

Lemma 2. *For an array $\mathbb{S}$, consider the k-essential Sperner family $\mathcal{E}'_k$ and the k-fragility F_k defined in Definitions 8 and 7, respectively. Let $\mathbb{I}$ be the set of inessential sensors in $\mathbb{S}$. Then F_k satisfies*

$$F_k \geq L_k, \qquad \text{for } k = 1, 2, \ldots, |\mathbb{S}| - |\mathcal{E}'_1| - 2, \tag{14}$$

where the lower bound L_k is given by

$$L_k = \begin{cases} \frac{|\mathcal{E}'_1|}{|\mathbb{S}|}, & \text{if } k = 1, \\ \frac{|\mathcal{E}'_1|}{|\mathbb{S}|}\left(2 - \frac{|\mathcal{E}'_1|-1}{|\mathbb{S}|-1}\right) + \frac{|\mathcal{E}'_2|}{\binom{|\mathbb{S}|}{2}}, & \text{if } k = 2, \\ 1 - \frac{1}{\binom{|\mathbb{S}|}{k}}\left[\binom{|\mathbb{I}|}{k} - \binom{|\mathbb{I}|-2}{k-2}C - |\mathcal{E}'_k|\right], & \text{if } 3 \leq k \leq |\mathbb{S}| - |\mathcal{E}'_1| - 2. \end{cases}$$

The cardinality of $\mathbb{I}$ is $|\mathbb{I}| = |\mathbb{S}| - |\mathcal{E}'_1|$ and the parameter $C \triangleq \min\{|\mathcal{E}'_2|, 1\}$. Furthermore, the equality in (14) *holds true for $k = 1$ and $k = 2$.*

Proof. If $k = 1$, then according to Definitions 7 and 8, we have $F_1 = |\mathcal{E}_1|/|\mathbb{S}| = |\mathcal{E}'_1|/|\mathbb{S}|$, so $F_1 = L_1$. If $k = 2$, then the k-essential subarray $\mathbb{A}$ can be divided into two non-overlapping categories:

1. The subarray $\mathbb{A}$ has at least one essential sensor. This case is the complement of the event that all sensors in $\mathbb{A}$ are inessential. As a result, there are $\binom{|\mathbb{S}|}{2} - \binom{|\mathbb{I}|}{2}$ choices of $\mathbb{A}$.
2. The subarray $\mathbb{A}$ belongs to $\mathcal{E}'_2$. There are $|\mathcal{E}'_2|$ subarrays.

As a result, the cardinality of $\mathcal{E}_2$ can be expressed as $|\mathcal{E}_2| = \binom{|\mathbb{S}|}{2} - \binom{|\mathbb{I}|}{2} + |\mathcal{E}'_2|$ and the k-fragility with $k = 2$ becomes

$$F_2 = \frac{|\mathcal{E}_2|}{\binom{|\mathbb{S}|}{2}} = 1 - \frac{\binom{|\mathbb{I}|}{2}}{\binom{|\mathbb{S}|}{2}} + \frac{|\mathcal{E}'_2|}{\binom{|\mathbb{S}|}{2}} = L_2,$$

where the last equality is due to $|\mathbb{I}| = |\mathbb{S}| - |\mathcal{E}'_1|$.

Let us consider the case when $3 \leq k \leq |\mathbb{S}| - |\mathcal{E}'_1| - 2$. Assume that $\mathbb{A}$ is a k-essential subarray of $\mathbb{S}$. According to Lemma 1, the k-essential family $\mathcal{E}_k$ can be decomposed into four components $\mathcal{G}_1$, $\mathcal{G}_2$, $\mathcal{H}$, and $\mathcal{E}'_k$

$$\mathcal{E}_k = \underbrace{\{\mathbb{A}\cup\mathbb{B} : \mathbb{A}\in\mathcal{E}'_1,\ \mathbb{B}\subseteq\mathbb{S}\backslash\mathbb{A},\ |\mathbb{B}| = k-1\}}_{\mathcal{G}_1} \tag{15}$$

$$\cup \underbrace{\{\mathbb{A}\cup\mathbb{B} : \mathbb{A}\in\mathcal{E}'_2,\ \mathbb{B}\subseteq\mathbb{S}\backslash\mathbb{A},\ |\mathbb{B}| = k-2\}}_{\mathcal{G}_2} \tag{16}$$

$$\cup \underbrace{\{\mathbb{A}\cup\mathbb{B} : \mathbb{A}\in\mathcal{E}'_\ell,\ \mathbb{B}\subseteq\mathbb{S}\backslash\mathbb{A},\ |\mathbb{B}| = k-\ell,\ 2<\ell<k\}}_{\mathcal{H}} \cup \mathcal{E}'_k. \tag{17}$$

Then the cardinality of $\mathcal{E}_k$ can be written as

$$\begin{aligned} |\mathcal{E}_k| &= |\mathcal{G}_1\cup\mathcal{G}_2\cup\mathcal{H}\cup\mathcal{E}'_k| \\ &= |\mathcal{G}_1| + |\mathcal{G}_2\backslash\mathcal{G}_1| + |(\mathcal{H}\backslash\mathcal{G}_1)\backslash\mathcal{G}_2| + |((\mathcal{E}'_k\backslash\mathcal{G}_1)\backslash\mathcal{G}_2)\backslash\mathcal{H}|. \end{aligned} \tag{18}$$

Next let us analyze the four terms in (18) separately.

1. The elements in $\mathcal{G}_1$ correspond to the subarrays with at least one essential sensor. Similar to the first category in the analysis of F_2, we have $|\mathcal{G}_1| = \binom{|\mathbb{S}|}{k} - \binom{|\mathbb{I}|}{k}$.
2. The cardinality $\mathcal{G}_2\backslash\mathcal{G}_1$ can be simplified as follows. First, if $\mathcal{E}'_2$ is empty, then $|\mathcal{G}_2\backslash\mathcal{G}_1| = 0$. Second, if $\mathcal{E}'_2$ is not empty, then there exists a set $\mathbb{X}\in\mathcal{E}'_2$. Based on $\mathbb{X}$, let us define a family $\mathcal{X}$ as

$$\mathcal{X} \triangleq \{\mathbb{X}\cup\mathbb{Y} : \mathbb{Y}\subseteq\mathbb{S}\backslash\mathbb{X},\ |\mathbb{Y}| = k-2,\quad \mathbb{Y} \text{ consists of inessential sensors}\}. \tag{19}$$

   According to (15), (16), and (19), it can be shown that $\mathcal{X}\subseteq\mathcal{G}_2\backslash\mathcal{G}_1$ so $|\mathcal{G}_2\backslash\mathcal{G}_1| \geq |\mathcal{X}| = \binom{|\mathbb{I}|-2}{k-2}$. Therefore, combining the cases of empty $\mathcal{E}'_2$ and non-empty $\mathcal{E}'_2$ leads to the following relation

$$|\mathcal{G}_2\backslash\mathcal{G}_1| \geq \binom{|\mathbb{I}|-2}{k-2} C, \tag{20}$$

   where $C \triangleq \min\{|\mathcal{E}'_2|, 1\}$.
3. The cardinality of a family is nonnegative so we have $|(\mathcal{H}\backslash\mathcal{G}_1)\backslash\mathcal{G}_2| \geq 0$.
4. According to Definition 8, the families $\mathcal{G}_1\cup\mathcal{G}_2\cup\mathcal{H}$ and $\mathcal{E}'_k$ are disjoint. Therefore $|((\mathcal{E}'_k\backslash\mathcal{G}_1)\backslash\mathcal{G}_2)\backslash\mathcal{H}| = |\mathcal{E}'_k|$.

Based on these cases, for $3 \leq k \leq |\mathbb{S}| - |\mathcal{E}'_1| - 2$, we have $|\mathcal{E}_k| \geq \binom{|\mathbb{S}|}{k} - \binom{|\mathbb{I}|}{k} + \binom{|\mathbb{I}|-2}{k-2}C + |\mathcal{E}'_k|$, so that $F_k \geq L_k$. □

The lower bound $F_{\min}$ and the lower bound L_k represent different tradeoffs between the computational complexity and tightness of the bounds. The lower bound $F_{\min}$ has lower computational complexity than L_k. The reason is as follows. Evaluating $F_{\min}$ only requires the number of sensors, but for L_k, the sizes of the k-essential Sperner family have to be known, so Algorithm 1 has to be executed first. As a trade-off, L_k is a tighter lower bound than $F_{\min}$, as demonstrated in Section 5.3 later.

5. Numerical Examples

5.1. Robustness of ULA

In this subsection, the robustness of ULA with N sensors will be studied using the proposed algorithms.

In the first experiment, we assume that $N = 10$ so the array configuration is $\mathbb{S}_{\text{ULA}} = \{0, 1, \ldots, 9\}$. Using Algorithm 1 with $K_{\max} = 10$ leads to the k-essential Sperner family of $\mathbb{S}_{\text{ULA}}$:

$$\mathcal{E}'_1 = \{\{0\}, \{9\}\}, \tag{21}$$

$$\mathcal{E}'_2 = \{\{1, 8\}\}, \tag{22}$$

$$\mathcal{E}'_3 = \{\{1, 2, 7\}, \{2, 7, 8\}\}, \tag{23}$$

$$\begin{aligned}\mathcal{E}'_4 = \{&\{1, 2, 3, 6\}, \{2, 3, 4, 5\}, \{2, 3, 6, 7\}, \{2, 4, 5, 7\},\\ &\{3, 4, 5, 6\}, \{3, 6, 7, 8\}, \{4, 5, 6, 7\}\},\end{aligned} \tag{24}$$

$$\mathcal{E}'_5 = \{\{1, 3, 4, 5, 7\}, \{1, 3, 5, 6, 7\}, \{2, 3, 4, 6, 8\}, \{2, 4, 5, 6, 8\}\}, \tag{25}$$

and $\mathcal{E}'_k = \varnothing$ for $6 \leq k \leq 10$. It can be observed that (21) to (23) are in accordance with the closed forms in [19, Theorem 2]. Furthermore, Algorithm 1 is able to find $\mathcal{E}'_4$ and $\mathcal{E}'_5$, as in (24) and (25), respectively, which are not given in the closed forms of [19, Theorem 2].

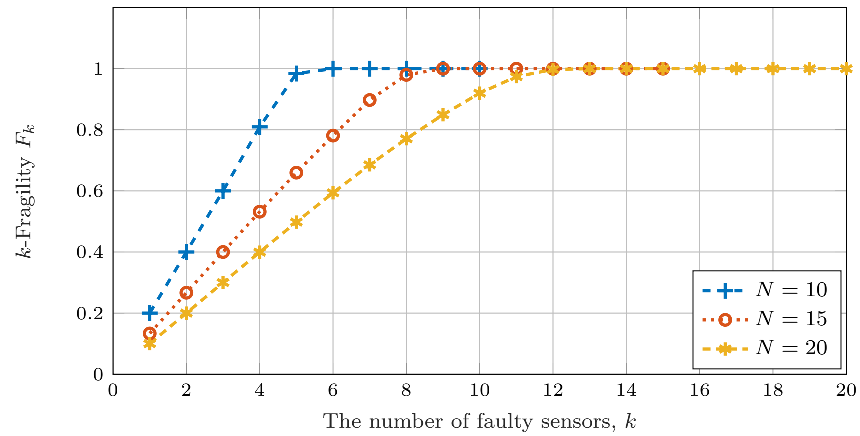

Figure 2: The k-fragility of ULA with $N = 10$, 15, and 20 physical sensors.

The second experiment considers the k-fragility of ULA with N physical sensors. Fig. 2 depicts the k-fragility F_k of ULAs with $N = 10$, 15, and 20 physical sensors. These F_k's are numerically computed by using Algorithms 1 and 2. Several observations can be drawn from Fig. 2. First, the fragility F_1 decreases as the number of sensors N increases, which is in accordance with the explicit expression $F_1 = 2/N$ for the ULA with $N \geq 4$ [19, (30)]. Furthermore, for a fixed k, F_k reduces as N increases (assuming that F_k is well-defined for that k). For instance, for $k = 6$, the k-fragility F_k is 1 for $N = 10$, approximately 0.8 for $N = 15$, and around 0.6 for $N = 20$.

Fig. 3 shows the running time of Algorithms 1 and 2 for ULA with N sensors. The simulation program is implemented on a workstation with Intel® Core™ i7-8700 CPU 3.20GHz, 32GB RAM, Ubuntu 16.04.6 LTS, and MATLAB® R2018b.

It is observed in Fig. 3 that for $N < 15$, Algorithm 2 takes less time, while for $N > 15$, Algorithm 1 is less time-consuming. This phenomenon is due to the following:

1. First let us compare the number of subarrays $\mathbb{A}$ to be examined in these algorithms. According to lines 3 to 18 of Algorithm 1 and lines 5 to 16 of

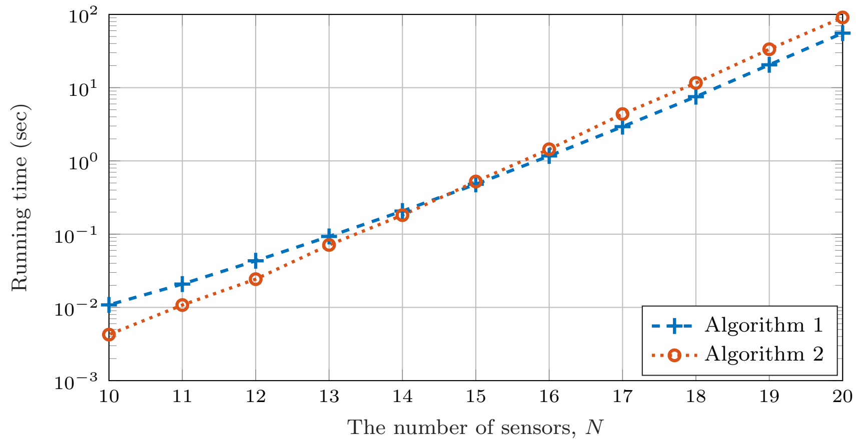

Figure 3: The dependence of the running time for Algorithms 1 and 2 on the number of physical sensors N in ULAs.

Algorithm 2, the number of subarrays $\mathbb{A}$ is *upper bounded* by

$$\text{Algorithm 1}: \qquad |\mathbb{S}| + \sum_{k=2}^{K_{\max}} \binom{|\mathbb{I}|}{k}, \tag{26}$$

$$\text{Algorithm 2}: \qquad \sum_{k=2}^{K_{\max}} \binom{|\mathbb{S}|}{k}. \tag{27}$$

Here the first term in (26) results from $\mathcal{E}_1'$, where $|\mathbb{S}|$ comparisons are needed. The second term in (26) represents the total number of subarrays from sizes $k = 2$ to $K_{\max}$. Similarly, the number of subarrays $\mathbb{A}$ in Algorithm 2 is given in (27). It was known in [19, Theorem 2] that the ULA has exactly two essential sensors if the number of sensors $N \geq 4$, implying that $|\mathbb{I}| = |\mathbb{S}| - 2$. Therefore, for sufficiently large $|\mathbb{S}|$, the quantity in (26) is smaller than that in (27).

2. Next let us consider the worst-case complexity for each subarray $\mathbb{A}$ in these algorithms. In Algorithm 1, each $\mathbb{A}$ involves the comparison of sets (lines 12 and 14) and the evaluation of the difference sets (line 14). However, in Algorithm 2, only the comparison of sets (line 8) is required. Therefore, for each subarray $\mathbb{A}$, Algorithm 2 typically enjoys less complexity than Algorithm 1, in the worst-case scenario.

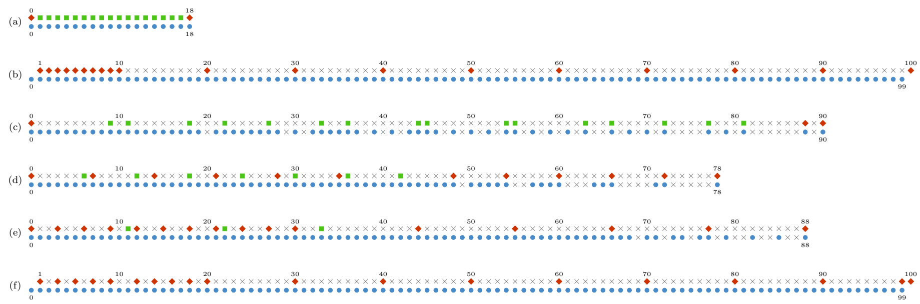

Figure 4: The array configurations of (a) ULA, (b) the nested array, (c) the prototype coprime array, (d) the (extended) coprime array, (e) CACIS with $p = 3$, and (f) the super nested array with $Q = 2$. The essential sensors and the inessential sensors are shown in red diamonds and green squares, respectively. The nonnegative parts of the difference coarray are illustrated in blue dots while crosses denote empty space.

Array	(a)	(b)	(c)	(d)	(e)	(f)
Aperture	18	99	90	78	88	99
$\lvert\mathbb{D}\rvert$	37	199	117	127	157	199
$\lvert\mathbb{U}\rvert$	37	199	39	97	137	199
Hole-free	Yes	Yes	No	No	No	Yes
$\lvert\mathcal{E}_1'\rvert$	2	19	3	12	16	19

Table 1: The profiles of the arrays (a) to (f) in Fig. 4.

The overall complexity is the combination of both factors. The results in Fig. 3 indicate that

1. For small N, the complexity for each subarray $\mathbb{A}$ is more prominent than the number of subarrays.
2. For large N, the number of subarrays is more crucial than the complexity for each $\mathbb{A}$, due to the growth of the gap between (27) and (26).

5.2. Numerical Comparison of the Robustness of Sparse Arrays

The proposed algorithms also facilitate the study of the robustness of a variety of array configurations. In this subsection, the following array configurations

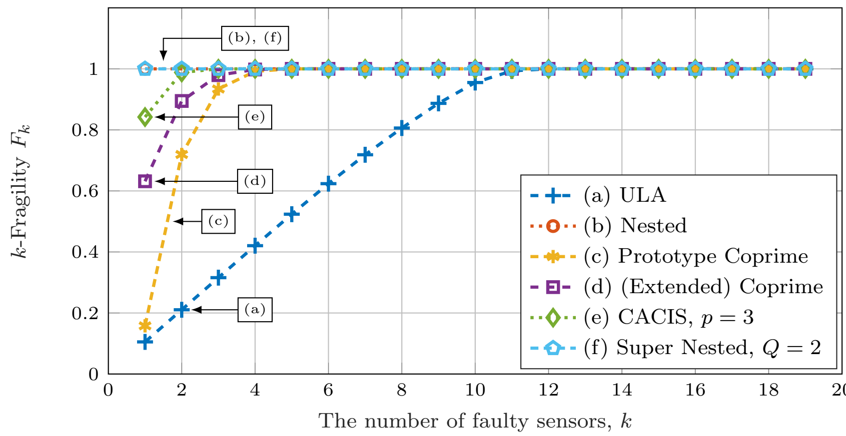

Figure 5: The k-fragility of several array configurations with 19 physical sensors.

will be considered:

(a) ULA with 19 sensors [35].

(b) The nested array with $N_1 = 9, N_2 = 10$ [24].

(c) The prototype coprime array with $M = 9$ and $N = 11$ [34].

(d) The (extended) coprime array with $M = 7$ and $N = 6$ [34].

(e) The coprime array with compressed inter-element spacing (CACIS). The parameters are $M = 9$, $N = 11$, and $p = 3$ [28].

(f) The super nested array with $N_1 = 9$, $N_2 = 10$, and $Q = 2$ [16].

All these array configurations have 19 physical sensors. Fig. 4 illustrates the essential sensors in red diamonds, the inessential sensors in green squares, and the nonnegative parts of the difference coarrays in blue dots. The essential sensors and inessential sensors in Fig. 4 are found according to Algorithm 1. The negative parts of the difference coarrays are not shown due to symmetry. Note that among these arrays, *the robustness of the prototype coprime array, the CACIS, and the super nested array has not been fully studied in the literature. With the proposed algorithms, it is possible to analyze the robustness of all these arrays numerically.*

Table 1 lists the profiles of these arrays, including the aperture ($\max(\mathbb{S}) - \min(\mathbb{S})$), the size of the difference coarray $|\mathbb{D}|$, the size of the central ULA segment of the difference coarray $|\mathbb{U}|$, the hole-free property, and the number of essential sensors $|\mathcal{E}'_1|$. Among these arrays in Fig. 4, the ULA has the smallest aperture and the fewest essential sensors. The nested array and the super nested array have the largest aperture and the most essential sensors (in particular, the nested array with $N_2 \geq 2$ was shown to be maximally economic [19, Theorem 1]). The ULA, the nested array, and the super nested array have hole-free difference coarrays. On the other hand, the prototype coprime array, the (extended) coprime array, and the CACIS have holes, but they are not maximally economic. Among these three arrays, the prototype coprime array has the smallest difference coarray, followed by the (extended) coprime array, and finally the CACIS. The smallest number of essential sensors is owned by the prototype coprime array, followed by the (extended) coprime array, and finally the CACIS.

Finally let us compare the k-fragility F_k of these arrays, by using Algorithms 1 and 2. It can be observed that the ULA is the most robust array, since it has the smallest F_k for all k. The nested array and the super nested array are maximally economic ($F_k = 1$ for all $1 \leq k \leq |\mathbb{S}|$). The prototype coprime array, the (extended) coprime array, and CACIS are less robust than the ULA, but more robust than the nested array and the super nested array.

5.3. The k-Fragility and the Lower Bounds

In this example, we consider the ULA with 19 sensors. Fig. 6 compares the k-fragility with the lower bound $F_{\min}$ in Property 3 of Theorem 1 and the lower bound L_k in Lemma 2. The following observations can be drawn. First, both $F_{\min}$ and L_k are lower bounds for F_k, which are in accordance with Property 3 of Theorem 1 and Lemma 2. Second, L_k is a tighter lower bound than $F_{\min}$ for $2 \leq k \leq N$. The reason is that the information of the k-essential Sperner family is utilized in the bound L_k. On the other hand, only the number of sensors is needed in the lower bound $F_{\min}$. Therefore the lower bound L_k follows the k-fragility F_k more closely than $F_{\min}$.

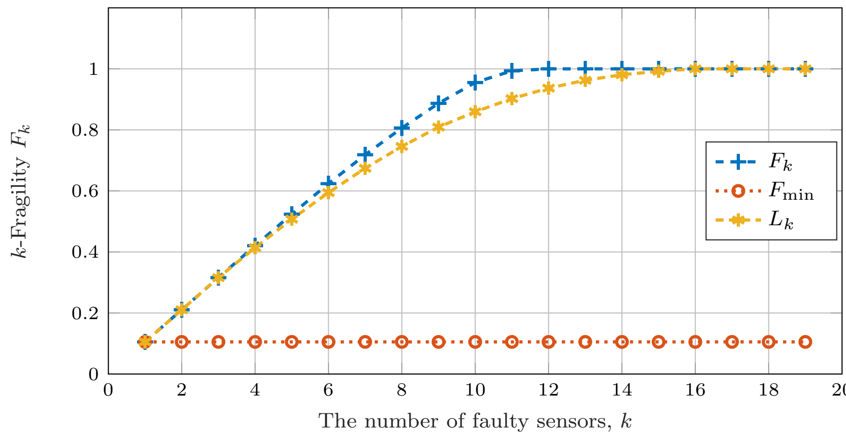

Figure 6: The k-fragility F_k, the lower bound in Property 3 of Theorem 1, and the lower bound L_k in Lemma 2 for the ULA with $N = 19$ sensors.

6. Concluding Remarks

This paper proposed numerical algorithms for evaluating certain characterizations of the robustness of the difference coarrays to sensor failures. The robustness of arrays was built upon the theory of the k-essentialness property, the k-essential family, the k-fragility, and the k-essential Sperner family. These attributes can be numerically evaluated by the newly proposed algorithms. The first algorithm efficiently finds the k-essential Sperner family, while the second algorithm computes the k-essential family and the k-fragility. We also presented a new lower bound for the k-fragility. This lower bound is not only simple to compute, but also tighter than the lower bound in [18].

In the future, it is of considerable interest to study the robustness of arrays not covered in [19], with the help of the proposed numerical algorithms. Detailed computational analyses of the proposed algorithms are of future interest as well.

7. Acknowledgments

This work was supported in part by the ONR grant N00014-18-1-2390, in part by the NSF grant CCF-1712633, in part by the California Institute of

Technology, in part by the Ministry of Education, Taiwan, under Grant Numbers NTU-107V0902 and NTU-108V0902, in part by the Ministry of Science and Technology, Taiwan, under Grant Number MOST 108-2218-E-002-043-MY2, and in part by the National Taiwan University.

References

[1] http://homepage.ntu.edu.tw/~chunlinliu/Robustness/Robustness.zip.

[2] Y. Abramovich, D. Gray, A. Gorokhov, N. Spencer, Positive-definite Toeplitz completion in DOA estimation for nonuniform linear antenna arrays–Part I: Fully augmentable arrays, IEEE Trans. Signal Process. 46 (9) (1998) 2458–2471.

[3] A. Alexiou, A. Manikas, Investigation of array robustness to sensor failure, Journal of the Franklin Institute 342 (3) (2005) 255 – 272.

[4] G. S. Bloom, S. W. Golomb, Applications of numbered undirected graphs, Proc. IEEE 65 (4) (1977) 562–570.

[5] R. N. Bracewell, Radio Astronomy Techniques, vol. 54, Berlin: Springer, 1962.

[6] M. Carlin, G. Oliveri, A. Massa, On the robustness to element failures of linear ADS-thinned arrays, IEEE Trans. Antennas Propag. 59 (12) (2011) 4849–4853.

[7] K. Engel, Sperner Theory, 1st ed., Cambridge University Press, 1997.

[8] L. Godara, Application of antenna arrays to mobile communications, Part II: Beam-forming and direction-of-arrival considerations, Proc. IEEE 85 (8) (1997) 1195–1245.

[9] J. R. Guerci, Space-Time Adaptive Processing for Radar, Artech House, 2003.

[10] R. T. Hoctor, S. A. Kassam, The unifying role of the coarray in aperture synthesis for coherent and incoherent imaging, Proc. IEEE 78 (4) (1990) 735–752.

[11] M. Ishiguro, Minimum redundancy linear arrays for a large number of antennas, Radio Science 15 (6) (1980) 1163–1170.

[12] D. H. Johnson, D. E. Dugeon, Array Signal Processing: Concepts and Techniques, Addison Wesley Pub. Co. Inc., 1993.

[13] A. Kleyner, P. O'Connor, Practical Reliability Engineering, John Wiley and Sons Ltd, 2012.

[14] H. Krim, M. Viberg, Two decades of array signal processing research: The parametric approach, IEEE Signal Process. Mag. 13 (4) (1996) 67–94.

[15] C.-L. Liu, P. P. Vaidyanathan, Remarks on the spatial smoothing step in coarray MUSIC, IEEE Signal Process. Lett. 22 (9) (2015) 1438–1442.

[16] C.-L. Liu, P. P. Vaidyanathan, Super nested arrays: Linear sparse arrays with reduced mutual coupling–Part I: Fundamentals, IEEE Trans. Signal Process. 64 (15) (2016) 3997–4012.

[17] C.-L. Liu, P. P. Vaidyanathan, Optimizing minimum redundancy arrays for robustness, in: Proc. IEEE Asil. Conf. on Sig., Sys., and Comp., 2018.

[18] C.-L. Liu, P. P. Vaidyanathan, Robustness of difference coarrays of sparse arrays to sensor failures – Part I: A theory motivated by coarray MUSIC, IEEE Trans. Signal Process., in pr 67 (12) (2019) 3213–3226.

[19] C.-L. Liu, P. P. Vaidyanathan, Robustness of difference coarrays of sparse arrays to sensor failures – Part II: Array geometries, IEEE Trans. Signal Process. 67 (12) (2019) 3227–3242.

[20] A. T. Moffet, Minimum-redundancy linear arrays, IEEE Trans. Antennas Propag. AP-16 (2) (1968) 172–175.

[21] M. Muma, Y. Cheng, F. Roemer, M. Haardt, A. M. Zoubir, Robust source number enumeration for r-dimensional arrays in case of brief sensor failures, in: Proc. IEEE Int. Conf. Acoust., Speech, and Sig. Proc., Kyoto, Japan, 2012, pp. 3709–3712.

[22] A. Myers, Complex System Reliability, Springer-Verlag GmbH, 2010.

[23] G. Oliveri, P. Rocca, A. Massa, Reliable diagnosis of large linear arrays–A Bayesian compressive sensing approach, IEEE Trans. Antennas Propag. 60 (10) (2012) 4627–4636.

[24] P. Pal, P. P. Vaidyanathan, Nested arrays: A novel approach to array processing with enhanced degrees of freedom, IEEE Trans. Signal Process. 58 (8) (2010) 4167–4181.

[25] P. Pal, P. P. Vaidyanathan, Coprime sampling and the MUSIC algorithm, in: Proc. IEEE Dig. Signal Process. Signal Process. Educ. Workshop, Sedona, AZ, USA, 2011, pp. 289–294.

[26] P. Pal, P. P. Vaidyanathan, Pushing the limits of sparse support recovery using correlation information, IEEE Trans. Signal Process. 63 (3) (2015) 711–726.

[27] S. Pillai, Y. Bar-Ness, F. Haber, A new approach to array geometry for improved spatial spectrum estimation, Proc. IEEE 73 (10) (1985) 1522–1524.

[28] S. Qin, Y. D. Zhang, M. G. Amin, Generalized coprime array configurations for direction-of-arrival estimation, IEEE Trans. Signal Process. 63 (6) (2015) 1377–1390.

[29] C. Ruf, Numerical annealing of low-redundancy linear arrays, IEEE Trans. Antennas Propag. 41 (1) (1993) 85–90.

[30] R. Schmidt, Multiple emitter location and signal parameter estimation, IEEE Trans. Antennas Propag. 34 (3) (1986) 276–280.

[31] M. I. Skolnik, Introduction to Radar Systems, 3rd ed., McGraw Hill, 2001.

[32] E. Sperner, Ein satz über untermengen einer endlichen menge, Mathematische Zeitschrift 27 (1) (1928) 544–548.

[33] Z. Tan, Y. Eldar, A. Nehorai, Direction of arrival estimation using co-prime arrays: A super resolution viewpoint, IEEE Trans. Signal Process. 62 (21) (2014) 5565–5576.

[34] P. P. Vaidyanathan, P. Pal, Sparse sensing with co-prime samplers and arrays, IEEE Trans. Signal Process. 59 (2) (2011) 573–586.

[35] H. L. Van Trees, Optimum Array Processing: Part IV of Detection, Estimation, and Modulation Theory, Hoboken, NJ, USA: Wiley, 2002.

[36] S. Vigneshwaran, N. Sundararajan, P. Saratchandran, Direction of arrival (DoA) estimation under array sensor failures using a minimal resource allocation neural network, IEEE Trans. Antennas Propag. 55 (2) (2007) 334–343.

[37] M. Wang, A. Nehorai, Coarrays, MUSIC, and the Cramér-Rao bound, IEEE Trans. Signal Process. 65 (4) (2017) 933–946.

[38] M. Wang, Z. Zhang, A. Nehorai, Direction finding using sparse linear arrays with missing data, in: Proc. IEEE Int. Conf. Acoust., Speech, and Sig. Proc., New Orleans, LA, USA, 2017, pp. 3066–3070.

[39] Y. D. Zhang, M. G. Amin, B. Himed, Sparsity-based DOA estimation using co-prime arrays, in: Proc. IEEE Int. Conf. Acoust., Speech, and Sig. Proc., 2013, pp. 3967–3971.