

Article

The Influence of Random Element Displacement on DOA Estimates Obtained with (Khatri–Rao-)Root-MUSIC

Veronique Inghelbrecht ^{1,*}, Jo Verhaevert ¹, Tanja van Hecke ¹ and Hendrik Rogier ²

¹ Department of Industrial Technology and Construction, Ghent University, Valentin Vaerwyckweg 1, 9000 Gent, Belgium; E-Mails: Jo.Verhaevert@UGent.be (J.V.); Tanja.Vanhecke@UGent.be (T.H.)

² Department of Information Technology, Ghent University, Sint-Pietersnieuwstraat 41, 9000 Gent, Belgium; E-Mail: Hendrik.Rogier@UGent.be

* Author to whom correspondence should be addressed; E-Mail: Veronique.Inghelbrecht@UGent.be; Tel.: +32-9-243-2592.

External Editor: Leonhard M. Reindl

Received: 5 September 2014; in revised form: 27 October 2014 / Accepted: 27 October 2014 /
Published: 11 November 2014

Abstract: Although a wide range of direction of arrival (DOA) estimation algorithms has been described for a diverse range of array configurations, no specific stochastic analysis framework has been established to assess the probability density function of the error on DOA estimates due to random errors in the array geometry. Therefore, we propose a stochastic collocation method that relies on a generalized polynomial chaos expansion to connect the statistical distribution of random position errors to the resulting distribution of the DOA estimates. We apply this technique to the conventional root-MUSIC and the Khatri–Rao-root-MUSIC methods. According to Monte-Carlo simulations, this novel approach yields a speedup by a factor of more than 100 in terms of CPU-time for a one-dimensional case and by a factor of 56 for a two-dimensional case.

Keywords: DOA-estimation; Khatri–Rao; steering vector error; stochastic collocation method

1. Introduction

DOA estimation is a major application of the sensor array, since there are many real-world problems where an accurate estimate of the source direction is essential: for example, in radar, electroencephalogram, sonar and microphone array systems. The number of sources that can be resolved

by an N -element uniform linear array using traditional subspace-based methods, such as MUSIC, is $N - 1$. Recently, more advanced methods have been presented for DOA estimation, for example, the Khatri–Rao (KR) product approach. By assuming quasi-stationary sources, this concept can identify up to $2N - 1$ sources using an N -element uniform linear array, without computing higher-order statistics [1].

Irrespective of the DOA estimation algorithm applied, deviations between the actual steering vector and the presumed steering vector are unavoidable in most applications that rely on an array of sensors. Errors during fabrication, uncalibrated arrays and arrays undergoing deformations are among the potential factors that can contribute to such errors. Moreover, the arrays may be composed of sensor elements with directional radiation patterns [2–4]. In most papers on DOA estimation, the authors assume that there is no steering vector error or that steering vector deviations are deterministic and constant in time. This has resulted in the development of algorithms that estimate such errors and, subsequently, remove them by a calibration procedure [5–9]. In many cases, however, these errors are random and, therefore, statistical in nature. The motivation of our work is that most calibration methods are computationally expensive and/or time consuming. Moreover, to estimate the modeling errors in the array, often, certain assumptions have to be made, additional experiments have to be carried out and/or specific training sequences must be transmitted. Therefore, these techniques are difficult to apply in real-time applications or in systems where the random errors change quickly. Hence, it is interesting to take into account the effect of uncertainty in the placement of the sensor elements by means of a stochastic framework. Hence, there is a need for stochastic simulation tools that provide the statistical data to quantify the effect of the random displacement of these sensor elements on the DOA estimates.

Conventionally, one may use Monte-Carlo simulations to quantify the effect of position errors of one of the sensor elements to characterize the resulting distribution of the estimated DOAs. However, Monte-Carlo simulations require a large amount of realizations to accurately capture the statistics of the random process. Hence, the method is time consuming.

In this paper, we introduce the stochastic collocation method (SCM) [10] as a more effective way to rapidly model uncertainty in the estimated DOAs, due to variations in the elements of the steering vector. Thanks to this method, we can reduce the CPU time by a factor of more than 100 for a one-dimensional displacement and by a factor of 56 for a two-dimensional displacement of one of the sensor elements.

In this paper, we apply the nominal steering vector of a uniform linear sensor array to estimate the DOAs of a signal impinging on a sensor array where one of the sensors has a random position error. We establish a stochastic framework to find the probability density function (pdf) of the DOA-estimates. To the best of the authors' knowledge, such a framework has not been established in the open literature yet. In [11], a first-order sensitivity analysis is carried out to study the effect of modeling errors on the conventional MUSIC algorithm. This sensitivity analysis results in a sensitivity parameter and failure threshold, but does not provide the complete statistical distribution of the modeling errors. In [12], the authors examine a resolution threshold of the MUSIC algorithm for situations in which the array response is perturbed from its assumed value. This could be a boundary approximation for our framework. However, they assume that the arrays are illuminated with two equal power emitters.

In order to illustrate and validate the SCM method, we derive the statistical distribution on the DOA estimates for the root-MUSIC algorithm and for the KR-root-MUSIC algorithm. The latter algorithm applies the root-MUSIC method to the noise subspace matrix of the KR algorithm, as described in [1]. For the (KR)-root-MUSIC algorithm, we derive the cumulative distribution function (cdf), for one- and two-dimensional displacements of one of the sensor elements.

In earlier work, it has been proven that underdetermined DOA estimation is possible when the source signals are non-Gaussian stationary. In [13], by the use of fourth-order cumulants, a virtual array with an increase in the degrees of freedom is achieved. In [14], a new array geometry, which is capable of increasing the degrees of freedom of linear arrays, is proposed. The structure is obtained by systematically nesting two or more uniform linear arrays. It can provide N^2 degrees of freedom using only N physical sensors and the second-order statistics of the data. Blind source separation, by use of quasi-stationarity, has also received attention. One technique is based on the least squares fitting (LSF) criterion, using parallel factor analysis (PARAFAC) [15]. One can prove that quasi-stationarity enables the identification of sources when the number of sensors is lower than the number of sources. However, the LSF is based on a multi-dimensional non-linear optimization problem. In [16], the authors have used a method called focusing Khatri–Rao subspace (FKR) for wideband array processing. They calculate a focusing matrix using a rotational signal-subspace [17], and then, the covariance matrices of different frequencies are transformed and combined by means of the KR-product. Recently, in [18], the authors used sparse covariance fitting in order to estimate underdetermined DOA estimation, and in [19], a sparse representation of the array covariance vector is used to obtain the DOAs. In [20,21], the Khatri–Rao approach is also extended to a uniform circular array. In [22], the Khatri–Rao approach is used for 2D DOA-estimation. In [23], the authors make use of a maximum likelihood method to obtain the underestimated DOAs for a multiple-input multiple-output radar.

Notations: We denote matrices and vectors by boldfaced capital letters and lower-case letters, respectively. Superscript H denotes the transpose conjugate, whereas superscript T denotes the transpose without the conjugate. For a given vector $\mathbf{x} \in \mathbb{C}$, $\|\mathbf{x}\|$ denotes its Euclidean norm. For a given matrix $\mathbf{A} \in \mathbb{C}^{M \times N}$, the i -th column is denoted by \mathbf{a}_i . The notation $\text{vec}(\mathbf{A})$ stands for vectorization; *i.e.*, if $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_N]$, then $\text{vec}(\mathbf{A}) = [\mathbf{a}_1^T, \dots, \mathbf{a}_N^T]^T$.

The remainder of the paper is organized as follows. In Section 2, we present the signal model and the KR-MUSIC algorithm [1]. Section 3 describes how the steering vector and the manifold array change if there is a displacement in one of the sensor elements. A brief description of the SCM, which is applied to model the uncertainty due to statistical variations of the input parameters, is given in Section 4. The proposed method is validated in Section 5, and conclusions are drawn in the last section.

2. Signal Model and Summary of the KR-MUSIC Algorithm

In this section, we give a short summary of the KR-MUSIC algorithm (see [1]), preceded by the signal model used.

2.1. Signal Model

Consider K sources impinging on a uniform linear sensor array, consisting of N omnidirectional sensors and neglecting mutual coupling. For a signal $s_k(t)$ emitted by the k -th source, let $\mathbf{s}(t) = [s_1(t), \dots, s_K(t)]^T$ denote the $K \times 1$ source signal vector. Let $\mathbf{a}(\theta) = [1, e^{j2\pi \frac{d}{\lambda} \sin(\theta)}, \dots, e^{j2\pi \frac{d}{\lambda} (N-1) \sin(\theta)}]^T$ be the $N \times 1$ steering vector corresponding to the direction θ , with d the spacing between the N sensors and λ the signal wavelength. In this paper, we fix the element spacing to $d/\lambda = 1/2$. Our aim is to study the effect of random variations in the array geometry on algorithms estimating the directions of arrival θ_k corresponding to the K source signals, in the presence of measurement noise. Therefore, $\mathbf{A} = [\mathbf{a}(\theta_1) \ \mathbf{a}(\theta_2) \ \dots \ \mathbf{a}(\theta_K)]$ describes the $N \times K$ array manifold matrix and $\mathbf{v}(t) = [v_1(t), \dots, v_N(t)]^T \in \mathbb{C}^{N \times 1}$ represents the spatial and temporal white noise. For the observed signal $x_n(t)$ of the n -th sensor, let $\mathbf{x}(t) = [x_1(t), \dots, x_N(t)]^T$ denote the $N \times 1$ receiver signal vector. This vector equals:

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{v}(t), \quad t = 0, 1, 2, \dots \quad (1)$$

The DOA-estimation algorithms under study are the root-MUSC and KR-root-MUSIC algorithms. These algorithms rely on the following assumptions:

- The noise $\mathbf{v}(t)$ is white Gaussian and zero-mean wide-sense stationary with covariance matrix $\mathbf{C} \triangleq E\{||\mathbf{v}(t)||^2\}$. It is statistically independent of the source signals.
- All source DOAs are distinct: $\theta_k \neq \theta_l$ for $k \neq l$.
- The source signals $s_k(t)$ are mutually uncorrelated and have a zero mean.
- Each source signal is wide-sense quasi-stationary with frame length L , such that:

$$E\{||s_k(t)||^2\} = d_{m,k}, \quad \forall t \in [(m-1)L, mL-1], \\ m = 1, 2, \dots \quad (2)$$

with $d_{m,k}$ the frame-dependent and sensor-dependent average normalized signal power. This assumption means that the second-order statistics of the source signals are time varying, but that they remain static over a short period of time. Under this quasi-stationarity assumption, we define a local covariance matrix:

$$\mathbf{R}_m = E\{||\mathbf{x}(t)||^2\} \in \mathbb{C}^{N \times N}, \\ \forall t \in [(m-1)L, mL-1], \quad m = 1, 2, \dots \quad (3)$$

In the remainder of the paper, we apply the DOA estimation algorithms to synthetic signals generated by the procedure proposed in [1] to construct source signals $s_k(t)$ satisfying the above constraints, with sequence length $T = 25,600$ and an allowable range of the frame periods $[L_{low}, L_{upp}] = [300, 700]$. Algorithm 1 is applied to generate the synthetic signals.

```

Set  $T_{cur} = 0$ .  $T_{cur}$  is the start of a new frame of the signal;
while  $T_{cur} < T$  do
    Randomly generate  $L_f$  following a uniform distribution on  $[L_{low}, L_{upp}]$ ;
    Randomly generate  $\sigma_s$  following a uniform distribution on  $[0,1]$ ;
    for  $t = T_{cur}$  to  $T_{cur} + L_f - 1$  do
        Randomly generate  $s_k(t) = s_R(t) + js_I(t)$ , where  $s_R(t)$  and  $s_I(t)$  are independent and
        identically Laplacian distributed with zero mean and variance  $\sigma_s^2/2$ ;
    end
     $T_{cur} := T_{cur} + L_f$ ;
end

```

Algorithm 1: The algorithm to generate synthetic signals.

Observe that the frame intervals of the different impinging sources are not fixed and not synchronized. We apply the KR subspace methods by choosing a fixed frame period of $L = 512$. The number of frames is set to $M = T/512 = 50$.

2.2. Summary of the KR-MUSIC Algorithm

For a given receiver signal vector $\{\mathbf{x}(t)\}_{t=0}^{T-1}$, with frame length L , where L divides T , the KR-MUSIC algorithm proceeds as follows [1]:

- Compute the local variance estimates

$$\hat{\mathbf{R}}_m = \frac{1}{L} \sum_{t=(m-1)L}^{mL-1} \mathbf{x}(t)\mathbf{x}^H(t) \text{ for } m = 1 \dots M.$$

- Denote an orthogonal complement projector $\mathbf{P}_{\mathbf{I}_M}^\perp = \mathbf{I}_M - \frac{1}{M}\mathbf{1}_M\mathbf{1}_M^T$, with \mathbf{I}_M the $M \times M$ identity matrix and $\mathbf{1}_M$ the $M \times 1$ all-one vector. Perform the projection $\bar{\mathbf{Y}} = \hat{\mathbf{Y}}\mathbf{P}_{\mathbf{I}_M}^\perp$ ($\bar{\mathbf{Y}} \in \mathbb{C}^{N^2 \times M}$), with $\hat{\mathbf{Y}}$ the data matrix formed by vectorize the covariance matrix. This projection eliminates the unknown noise covariance.
- As in [1], perform a dimension reduction on $\bar{\mathbf{Y}}$ obtaining $\tilde{\mathbf{Y}}$.
- To apply MUSIC to $\tilde{\mathbf{Y}}$, perform a singular value decomposition (SVD) on:

$$\tilde{\mathbf{Y}} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H \quad (4)$$

where $\mathbf{U} \in \mathbb{C}^{(2N-1) \times K}$ and $\mathbf{V} \in \mathbb{C}^{M \times K}$ are the left and right singular matrices, respectively, and $\mathbf{\Sigma} \in \mathbb{R}^{K \times K}$ is the diagonal singular values matrix with the singular values being arranged in descending order.

- By extracting the noise subspace matrix $\mathbf{U}_n = [\mathbf{u}_{K+1}, \dots, \mathbf{u}_{2N-1}] \in \mathbb{C}^{(2N-1) \times (2N-1-K)}$ from \mathbf{U} , we compute the DOA spectrum:

$$P_{\text{KR-MUSIC}}(\theta) = \frac{1}{\|\mathbf{U}_n^H \mathbf{W}^{\frac{1}{2}} \mathbf{b}(\theta)\|^2} \quad (5)$$

Define $\mathbf{b}_r(z) \in \mathbb{C}^{1 \times (2N-1)}$ as:

$$\mathbf{b}_r(z) = [z^{-(N-1)}, \dots, z^{-1}, 1, z, z^2, \dots, z^{N-1}]^T \quad (6)$$

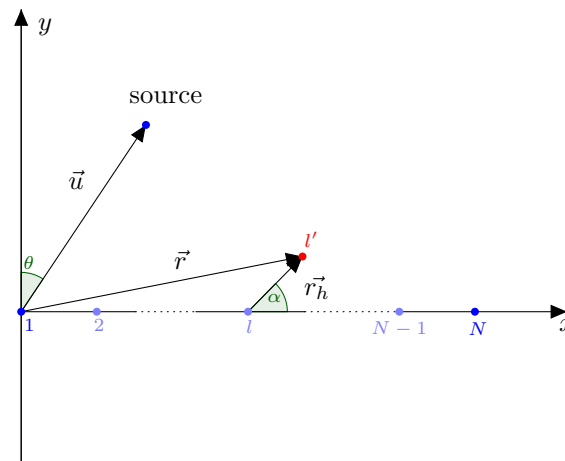
and $\mathbf{b}(\theta) = \mathbf{b}_r(e^{j2\pi\frac{d}{\lambda}\sin(\theta)})$. The K largest local maxima of the DOA spectrum $P_{\text{KR-MUSIC}}(\theta)$ are the DOA estimates.

Instead of searching the peaks of the MUSIC spectrum, we can also apply the root-MUSIC method [24] to $\|\mathbf{U}_n^H \mathbf{W}\|^2$. In the remainder of the paper, we call the resulting method the KR-root-MUSIC method.

3. Displacement of One of the Sensor Elements

In this section, we assume that one of the sensor elements is displaced due to an unintentional movement of an element. Figure 1 depicts an array of N sensor elements, where the first sensor element is placed at the origin of the xy -plane and the other elements along the x -axis of this plane. The l -th sensor element is displaced by a vector \vec{r}_h with respect to its original position.

Figure 1. Displacement of the l -th sensor element.



Denote $r_h = \|\vec{r}_h\|$ and $\alpha = \arg(\vec{r}_h)$. The new coordinates of the displaced l -th sensor are given by $\vec{r} = [(l-1)d + r_h \cos(\alpha), r_h \sin(\alpha)]$. Define $\beta = \arg(\vec{r})$ and $r = \|\vec{r}\|$, such that $\frac{2\pi}{\lambda} \vec{u} \cdot \vec{r} = \frac{2\pi}{\lambda} r \sin(\theta + \beta)$. \vec{u} is the unit vector with the same direction of the position vector of the source, which is located in the array's far field. The l -th row of the array manifold matrix then becomes $[e^{j2\pi\frac{r}{\lambda}\sin(\theta_1+\beta)}, e^{j2\pi\frac{r}{\lambda}\sin(\theta_2+\beta)}, \dots, e^{j2\pi\frac{r}{\lambda}\sin(\theta_K+\beta)}]$.

4. Stochastic Collocation Method

In order to investigate the influence of a stochastic displacement of one of the sensor elements, we need stochastic simulation tools to characterize the statistical distribution of the DOAs due to the distribution of the random displacement vectors.

A Monte-Carlo simulation for such a task requires a large amount of realizations to accurately capture the statistics of the random process. Therefore, we apply the SCM proposed in [10] as a more effective way to rapidly model geometrical uncertainty, due to random variations of the array element positions.

4.1. One-Dimensional Stochastic Input

Assume that an input random variable X (in our case, e.g., the displacement in the x - or y -direction) is given. We want to determine the variation of the output $Y = f(X)$ (in this paper, the DOA estimates), due to the statistical variation of X . The random variable X follows the cumulative distribution P^X and probability density function dP^X in the sample space Ω . To determine the statistics of Y , we rely on the Askey scheme [25] to approximate the transformation $Y = f(X)$ by a polynomial expansion of order P .

$$Y \approx f^P(X) = \sum_{k=0}^P y_k^X \phi_k^X(X), \quad (7)$$

with $\phi_k^X(X)$ the expansion polynomials and y_k^X the weights belonging to the k -th expansion polynomial. $f^P(X)$ is the generalized polynomial chaos expansion (GPoC expansion) of $Y = f(X)$.

An optimal expansion is obtained when the set of expansion polynomials forms a complete orthogonal basis in Ω with orthogonality relation:

$$\begin{aligned} \langle \phi_i^X(x), \phi_j^X(x) \rangle &= \int_{\Omega} \phi_i^X(x) \phi_j^X(x) dP^X(x) \\ &= \langle (\phi_i^X(x))^2 \rangle \delta_{ij}, \end{aligned} \quad (8)$$

with the Kronecker $\delta_{ij} = 0$ if $i \neq j$ and $\delta_{ij} = 1$ if $i = j$.

In this case, the Cameron–Martin [26] convergence theorem ensures exponential convergence to the function $Y = f(X)$ for $P \rightarrow \infty$. In the remainder of the paper, we assume that the statistical variation of X follows a Gaussian distribution in the sample space Ω . Relying on the Askey scheme, we know that the probabilistic Hermite polynomials $H_k(X)$ provide an optimal expansion for the normal Gaussian distribution [25].

To determine the unknown expansion coefficients y_k^X , we apply Galerkin weighting to Equation (7) [27]:

$$y_k^X = E[Y(x) \phi_k^X(x)] = \int_{\Omega} Y(x) \phi_k^X(x) dP^X(x). \quad (9)$$

We can approximate this integral by a Q -point Gauss–Hermite quadrature rule, being [28]:

$$y_k^X \approx \sum_{i=1}^Q w_i Y(x_i) \phi_k^X(x_i) \quad k = 0, 1, \dots, P. \quad (10)$$

where the quadrature points x_i are given by the Q zeros of $\phi_Q^X(x)$ in Ω and with w_i the corresponding weights. In order to evaluate Equation (10), we must determine $Y = f(X)$ for Q realizations of the random variable X , corresponding to the quadrature points. When utilizing a Gaussian quadrature, Equation (10) exactly integrates all polynomials of degree equal or less than $2Q - 1$. Given an expansion order P , the highest order coefficient evaluation of Equation (9) can be assumed to involve integrands of at least polynomial order $2P$, being $\phi_k^X(x)$ of order P and $Y(x)$ modeled to order P , such that a minimal Gaussian quadrature order of $P+1$ will be required to obtain good accuracy in these coefficients. Given the exponential convergence of the expansion (7), using the SCM theory, we need to apply the KR-root-MUSIC algorithm in the $Q = P + 1$ quadrature points. The Monte-Carlo analysis may then

be applied to the computationally cheap expansion (7) instead of to the KR-root-MUSIC. This results in huge savings in CPU-time, as the Monte-Carlo method is only slowly converging with respect to the number of samples taken, being inversely proportional to the square root of the number of samples [29].

4.2. Two-Dimensional Stochastic Input

We assume that the variation of the output $Y = f(X_1, X_2)$ depends on two independent stochastic variables X_1 and X_2 , respectively, in the sample space Ω_1 and Ω_2 , respectively (e.g., a displacement in the x - and y -direction).

We first extend the GPoC expansion (7) to two dimensions. For $Y = f(X_1, X_2)$, we obtain:

$$Y = f(X_1, X_2) = \sum_{i=0}^{P_1} \sum_{j=0}^{P_2} y_{ij}^{X_1 X_2} \phi_i^{X_1}(X_1) \phi_j^{X_2}(X_2) \quad (11)$$

which is a polynomial expansion of degree P_1 in X_1 and P_2 in X_2 . The weight coefficients are given by:

$$y_{ij}^{X_1 X_2} = \int_{\Omega_1} \int_{\Omega_2} Y(x_1, x_2) \phi_i^{X_1}(x_1) \phi_j^{X_2}(x_2) dP^{X_1}(x_1) dP^{X_2}(x_2) \quad (12)$$

or if we approximate Equation (12) by a tensor product quadrature rule:

$$y_{ij}^{X_1 X_2} = \sum_{k=1}^{Q_1} \sum_{l=1}^{Q_2} w_k \phi_i^{X_1}(u_k) f(u_k, u_l) w_l \phi_j^{X_2}(u_l), \quad (13)$$

with the quadrature points u_k and u_l given by the Q_1 zeros of $\phi_{Q_1}^{X_1}(x_1)$ and the Q_2 zeros of $\phi_{Q_2}^{X_2}(x_2)$, respectively. Equation (13) requires $Q_1 \cdot Q_2$ function evaluations. If we only need a small number of quadrature points (whenever we can model Y as a polynomial of low degree), this is a very effective numerical tool. However, the number of function evaluations grows quadratically with the number of quadrature points Q (in the case that $Q = Q_1 = Q_2$).

In the case that X_1 and X_2 are independent Gaussian distributed variables with zero mean and dispersions σ_1 and σ_2 , respectively, we can rewrite Equation (12) as:

$$\begin{aligned} y_{ij}^{X_1 X_2} &= \int_{\Omega_1} \int_{\Omega_2} \frac{e^{-x_1^2 - x_2^2}}{\pi} H_i(\sqrt{2}\sigma_1 x_1) H_j(\sqrt{2}\sigma_2 x_2) \\ &\quad \cdot Y(\sqrt{2}\sigma_1 x_1, \sqrt{2}\sigma_2 x_2) dx_1 dx_2 \\ &= \int_{\Omega_1} \int_{\Omega_2} p(x_1, x_2) e^{-x_1^2 - x_2^2} dx_1 dx_2, \end{aligned} \quad (14)$$

with,

$$p(x_1, x_2) = \frac{1}{\pi} H_i(\sqrt{2}\sigma_1 x_1) H_j(\sqrt{2}\sigma_2 x_2) Y(\sqrt{2}\sigma_1 x_1, \sqrt{2}\sigma_2 x_2). \quad (15)$$

We approximate this integral by using cubature formulas for the plane with weight function e^{-r^2} ([30,31]):

$$y_{ij}^{X_1 X_2} = \sum_{k=1}^Q w_k p(x_{1,k}, x_{2,k}), \quad (16)$$

with $(x_{1,k}, x_{2,k})$ the quadrature points and w_k the corresponding weights. For these formulas, the number of function evaluations is limited to Q . We call d_e the degree of the cubature formula, meaning that Equation (16) is exact for polynomials of degree d_e .

The cubature formulas that we have applied in this paper are those from [32] with $Q = 44$, $d_e = 15$, those from [33] with $Q = 99$, $d_e = 21$ and those from [33] with $Q = 172$, $d_e = 31$.

Given the exponential convergence of the polynomial expansions, for the two-dimensional expansion, we have to apply the KR-root-MUSIC algorithm only $Q_1 \cdot Q_2$ times, when making use of the tensor rule Equation (13). Whereas, if we use the cubature formulas following Equation (16), we need to apply the KR-root-MUSIC algorithm Q times. Once we have estimated the DOAs in the quadrature points, the whole distribution can be obtained by applying the Monte-Carlo method to the computationally cheap polynomial expansion instead of to the DOA estimation algorithms.

5. Results

In all simulation examples below, the signal-to-noise ratio SNR (in dB) is defined as:

$$\text{SNR} = 10 \log_{10} \left(\frac{E \{ \|\mathbf{A}\mathbf{s}(\mathbf{t})\|^2 \}}{E \{ \|\mathbf{v}(\mathbf{t})\|^2 \}} \right) \quad (17)$$

In order to validate the efficiency of the SCM method, we will provide several numerical examples. We first consider the case where the displacement vector \vec{r}_h is a constant vector. Next, we focus on the more general case where the x - and/or the y -coordinates of the displacement vector are random variables.

In Section 5.1, we investigate the influence on the DOA estimates obtained by root-MUSIC and KR-root-MUSIC for a displacement of one sensor element, in absence of noise. In Section 5.2, we use the SCM method to approximate the curves obtained in Section 5.1. In Section 5.3, we analyze the cdf for a displacement of the sensor element in one direction, for noisy data signals. In Section 5.4, we derive the GPoC expansion by means of the SCM method for a two-dimensional shift.

5.1. One Shifted Sensor Element: DOA Estimation in Absence of Noise

Consider the case of $K = 3$ sources and $N = 4$ sensor elements. The true DOAs are assumed to be $[-18^\circ, 5^\circ, 25^\circ]$. We apply the synthetic signal generation procedure of Section 2.1 to generate a random source signal and use this signal vector throughout the simulations. We furthermore assume that no noise has been added to the impinging signal. We assume that one of the sensor elements has moved along the x -axis. For e_x^{rel} , being the relative displacement (with respect to the distance d between two sensor elements) of the sensor element, \vec{r}_h corresponds to coordinates $[e_x^{rel}d, 0]$.

In Figures 2 and 3, we plot the estimated DOAs, as a function of e_x^{rel} for a relative displacement of each of the sensor elements for the root-MUSIC algorithm and the MUSIC algorithm. As a reference, the figures also show the true DOAs as dashed lines. We observe that even without any noise, the MUSIC algorithm is sensitive to a shift of one of the sensor elements and that if that shift is too large, in some cases, we do not find peaks in the MUSIC spectrum, such as in Figure 3 for $e_x^{rel} > 0.2$. On the same figure, we observe that the root-MUSIC algorithm [24] estimates all DOAs, even in these cases where the conventional MUSIC algorithm does not find enough peaks. Furthermore, we observe that the DOAs

estimated by the root-MUSIC algorithm and those estimated by the MUSIC algorithm are almost the same. The figures for a displacement of the first and fourth array element and those for a displacement of the second and third array element will almost be symmetric.

Figure 2. The estimated DOAs by means of (root)-MUSIC as a function of a shift of the first sensor element along the x -axis for true DOAs = $[-18^\circ, 5^\circ, 25^\circ]$ and $(N, K) = (4, 3)$.

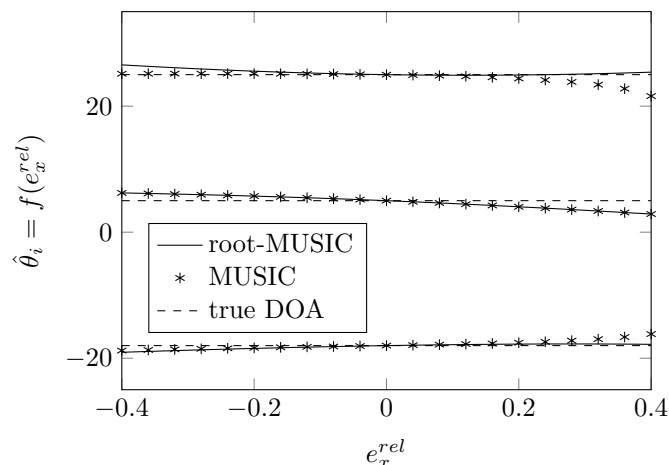
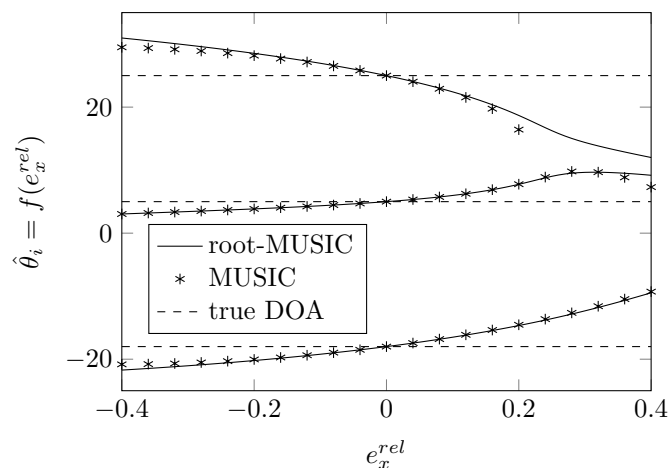


Figure 3. The estimated DOAs by means of (root)-MUSIC as a function of a shift of the second sensor element along the x -axis for true DOAs = $[-18^\circ, 5^\circ, 25^\circ]$ and $(N, K) = (4, 3)$.



In Figures 4 and 5, we have plotted the estimated DOAs, as a function of e_x^{rel} , for a random relative displacement of the first and second sensor element for the KR-MUSIC and KR-root-MUSIC algorithm. In contrast to MUSIC, the KR-MUSIC algorithm is more robust to shifts in one of the sensor elements. The Khatri–Rao transformation expands the dimension of the vector on which we perform the SVD and the extracted noise subspace matrix. This increases the probability of finding K peaks. We also observe that in the KR-MUSIC case, the estimated curves are almost linear as a function of the shifts. Furthermore, we observe that DOAs estimated by using the KR-MUSIC algorithm are almost the same as those estimated by the KR-root-MUSIC algorithm.

Figure 4. The estimated DOAs by means of Khatri–Rao (KR)-(root)-MUSIC as a function of a shift of the first sensor element along the x -axis for true DOAs = $[-18^\circ, 5^\circ, 25^\circ]$ and $(N, K) = (4, 3)$.

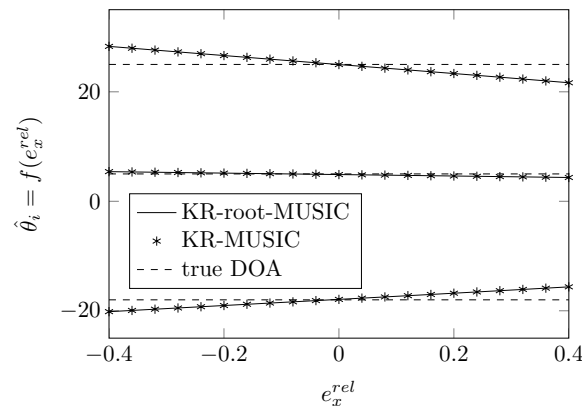
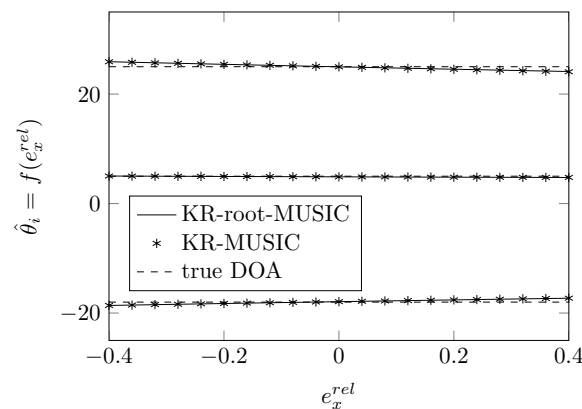


Figure 5. The estimated DOAs by means of KR-(root)-MUSIC as a function of a shift of the second sensor element along the x -axis for true DOAs = $[-18^\circ, 5^\circ, 25^\circ]$ and $(N, K) = (4, 3)$.



5.2. One-Dimensional GPoC Expansion

In this section, we start from the same assumptions as in Section 5.1, concerning the impinging signal and the sensor array.

Let $f(e_x^{rel})$ be the estimation of the second DOA (true DOAs = $[-18^\circ, 5^\circ, 25^\circ]$) as a function of the relative displacement of the second sensor element along the x -axis e_x^{rel} . As plotted in the top figure of Figure 6, the relative displacement e_x^{rel} is assumed to be Gaussian distributed with dispersion $\sigma_{pl} = 0.08$. As outlined in Section 4, we therefore apply Hermite polynomials in the GPoC expansion of $f(e_x^{rel})$. We use the SCM method (with Q quadrature points and a polynomial expansion of order $P = Q - 1$) to determine the expansion coefficients for the GPoC expansion. In the bottom figure of Figure 6, we have plotted the function $f(e_x^{rel})$ and the GPoC expansion of $f(e_x^{rel})$. We remark that, for e_x^{rel} between -0.16 and 0.16 , all plotted curves coincide within 1%. We also observe that, if we make use of a higher expansion order for the polynomials, the GPoC expansions better approximate the curve representing $f(e_x^{rel})$. For a polynomial expansion of order 16, the GPoC curve coincides with the $f(e_x^{rel})$ curve within 8% in the observed interval.

Figure 6. (Top) The Gaussian distribution with zero mean and $\sigma_{pl} = 0.08$; (Bottom) root-MUSIC DOA estimate, approximated by the generalized polynomial chaos (GPoC) of order P , with Q quadrature points as a function of e_x^{rel} , for a displacement of the second sensor element, true DOA = 5° and $(N, K) = (4, 3)$.

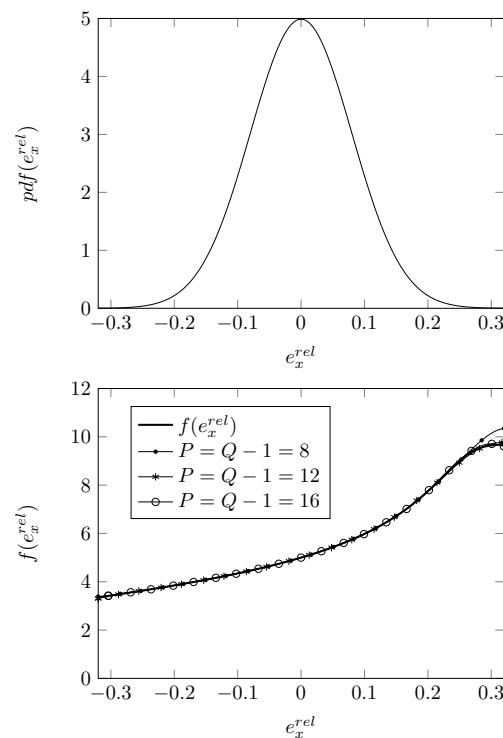


Figure 7. (Top) The Gaussian distribution with zero mean and $\sigma_{pl} = 0.2$; (Bottom) root-MUSIC DOA estimate, approximated by the GPoC of order P , with Q quadrature points as a function of e_x^{rel} , for a displacement of the second sensor element, true DOA = 5° and $(N, K) = (4, 3)$.

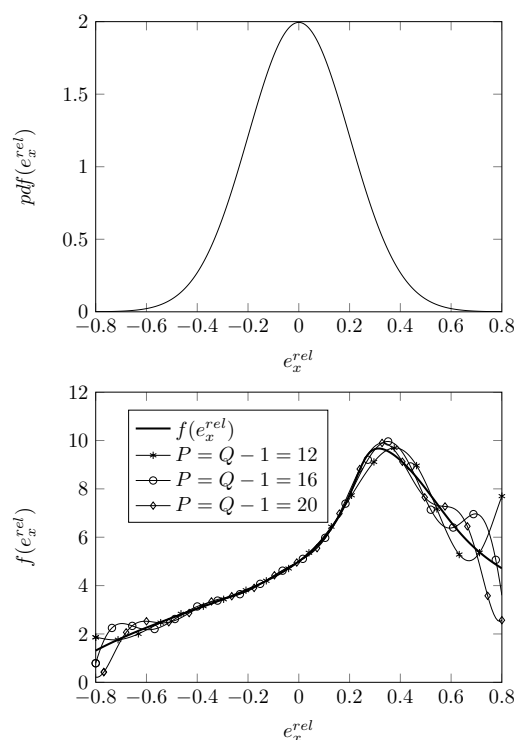
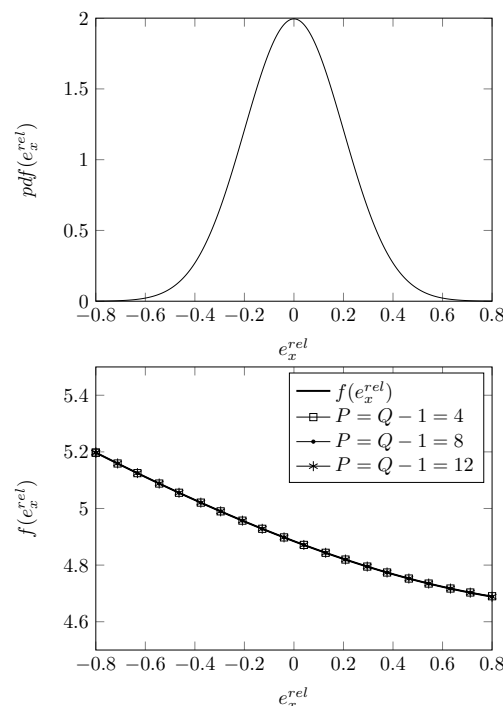


Figure 8. (Top) The Gaussian distribution with zero mean and $\sigma_{pl} = 0.2$; (Bottom) KR-root-MUSIC DOA estimate, approximated by the GPoC of order P , with Q quadrature points as a function of e_x^{rel} , for a displacement of the second sensor element, true DOA = 5° and $(N, K) = (4, 3)$.



In Figure 7, we apply the SCM method with $P = Q - 1$ for e_x^{rel} Gaussian distributed with dispersion $\sigma_{pl} = 0.2$. We observe that, in the 95% confidence interval $[-0.4, 0.4]$ for e_x^{rel} , the GPoC based on the Hermite polynomials of order 12 or higher provides an accurate approximation for $f(e_x^{rel})$. However, beyond the 95% confidence interval, the Hermite polynomials rapidly oscillate, and they do not yield a good approximation for the $f(e_x^{rel})$ curve.

In Figure 8, we have made the same assumptions as in Figure 7, but we have estimated the DOA using the KR-root-MUSIC algorithm. The curves coincide, even when restricting the GPoC to only $P = 4$.

5.3. One Shifted Element: Statistical Distribution of DOA Estimates

In this section, we again apply the synthetic signal generation procedure of Section 2.1 to generate a source signal. We construct an arbitrary complex Gaussian distributed noise vector $\mathbf{v}(t) = v_R(t) + v_I(t)$. $v_R(t)$ and $v_I(t)$ are both Gaussian distributed with zero mean and dispersion $\sigma = 0.3$ providing an SNR = 7.78 dB. Throughout the simulations, we use this source signal vector and noise vector. We also assume that the first sensor element is displaced along the x -axis and that the relative displacement e_x^{rel} is Gaussian distributed with zero mean and dispersion $\sigma_{pl} = 0.12$. To obtain the cdfs, we use two different methods: Monte-Carlo (MC) simulation and SCM. We carry out an MC simulation (where we vary e_x^{rel}) with 10,000 realizations. The CPU-time to carry out this simulation equals about 126 s for the root-MUSIC algorithm and 157 s for the KR-root-MUSIC algorithm. The simulations were performed on a Dell Latitude E6520PC with an Intel(R) Core(TM)i5-2520M 2.5 GHz processor and 4 GB RAM. The MATLAB version used is 8.0.0.783 (R2012b).

Table 1. The measured value $D_{n,n'}$ of the Kolmogorov–Smirnov (KS) test (critical value = 0.0192), one-dimensional shift along the x -axis of first sensor element, root-MUSIC algorithm, true DOAs = $[-18^\circ, 5^\circ, 25^\circ]$; the last column is the simulation time.

root-MUSIC, $\sigma_{pl} = 0.12$					
$D_{n,n'}$ for					
P	Q	θ_1	θ_2	θ_3	time
4	5	0.0088	0.0004	0.0190	0.16 s
8	9	0.0002	0.0002	0.0006	0.17 s
12	13	0.0002	0.0001	0.0005	0.25 s
16	17	0.0001	0.0001	0.0002	0.39 s

Table 2. The measured value $D_{n,n'}$ of the KS test (critical value = 0.0192), one-dimensional shift along the x -axis of first sensor element, KR-root-MUSIC algorithm, true DOAs = $[-18^\circ, 5^\circ, 25^\circ]$; the last column is the simulation time.

KR-root-MUSIC, $\sigma_{pl} = 0.12$					
$D_{n,n'}$ for					
P	Q	θ_1	θ_2	θ_3	time
4	5	0.0002	0.0001	0.0003	0.09 s
8	9	0.0001	0.0001	0.0001	0.33 s
12	13	0.0001	0.0001	0.0001	0.36 s
16	17	0.0001	0.0001	0.0001	0.55 s

We also apply the SCM theory for different values of the quadrature points Q and a polynomial expansion of Y of order $P = Q - 1$. We perform a two-sample Kolmogorov–Smirnov test (KS test) [34], to compare the samples obtained by the MC method with those obtained by the SCM method. The two-sample KS test verifies the null hypothesis that the two samples come from the same distribution. The test statistic is defined as:

$$D_{n,n'} = \sup_x (|F_{1,n}(x) - F_{2,n'}(x)|), \quad (18)$$

with $\sup_x(f(x))$ the supremum of the function $f(x)$. $F_{1,n}(x)$ is the empirical cdf obtained with the MC and $F_{2,n'}(x)$ the empirical cdf obtained with SCM. $n = n' = 10,000$ are the number of samples used for the MC and SCM method. The null hypothesis is rejected at level α if $D_{n,n'} > c(\alpha)\sqrt{\frac{n+n'}{nn'}}$, where, for example, $\alpha = 0.05$ corresponds to $c(\alpha) = 1.36$. In our case, the critical value is $D_{n,n'} = 0.0192$. In Tables 1 and 2, we show the values of the test statistic $D_{n,n'}$, for different values of the number of quadrature points Q and orders of polynomial expansion $P = Q - 1$. We consider the root-MUSIC algorithm and the KR-root-MUSIC algorithm, respectively. The two-sample KS test accepts, with a confidence of 95%, that, for the root-MUSIC algorithm and for $P = Q - 1 = 4$, the SCM samples and the MC samples have the same distribution. We also show the time necessary to obtain the cdf using the SCM method. Note that the simulation time to obtain the samples for the SCM algorithm is a factor more than 100 smaller than the simulation time to calculate the samples obtained by the MC method.

For the KR-root-MUSIC algorithm, both empirical distributions are the same for $P = Q - 1 = 4$. In Figures 9–11, we have plotted the cdfs both obtained by the SCM method (with $Q = 9$ quadrature points and a polynomial expansion of order $P = 8$) and the MC method. We observe that, even for a small number of quadrature points, we see almost no difference between the curves obtained by the SCM theory and the curves obtained by MC simulation, as indicated by the true sample KS test. Moreover, for the first and third estimated DOA, the root-MUSIC algorithm estimates the DOA very accurately. For the estimation of the second DOA, however, the KR-root-MUSIC algorithm performs better.

Figure 9. $P = Q - 1 = 8$, the cdf of $\hat{\theta}_1$ when the first sensor element is displaced in the x -direction, $\sigma_{pl} = 0.12$, true DOA = -18° and $(N, K) = (4, 3)$.

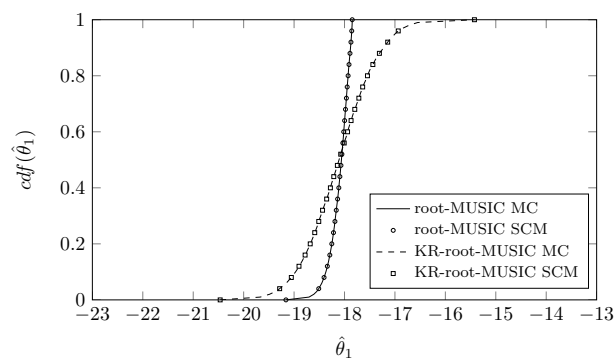


Figure 10. $P = Q - 1 = 8$, the cdf of $\hat{\theta}_2$ when the first sensor element is displaced in the x -direction, $\sigma_{pl} = 0.12$, true DOA = 5° and $(N, K) = (4, 3)$.

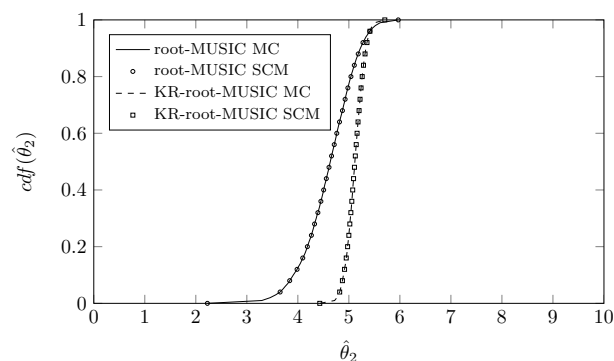
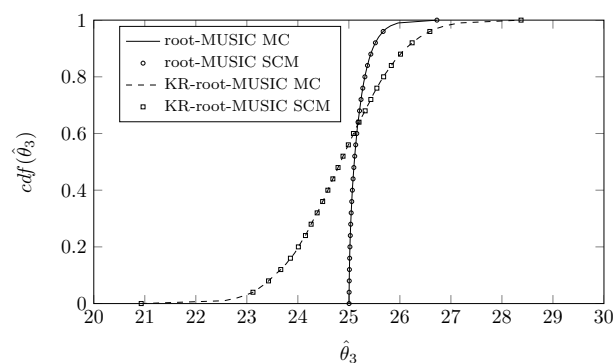


Figure 11. $P = Q - 1 = 8$, the cdf of $\hat{\theta}_3$ when the first sensor element is displaced in the x -direction, $\sigma_{pl} = 0.12$, true DOA = 25° and $(N, K) = (4, 3)$.



5.4. Two-Dimensional Displacement of One Sensor Element

In this section, we assume that the first sensor element is displaced by $\vec{r}_h = [e_x^{rel}d, e_y^{rel}d]$ (Figure 1). The relative displacements in both the x -direction e_x^{rel} and y -direction e_y^{rel} are independently Gaussian distributed, with zero mean and dispersion σ_{pl} . This means that the relative length of the displacement vector is Rayleigh distributed with mean $\sigma_{pl}\sqrt{\frac{\pi}{2}}$ and variance $\frac{4-\pi}{2}\sigma_{pl}^2$ and that the argument of the displacement vector is uniformly distributed in $[-\pi, \pi]$.

Table 3. The measured value $D_{n,n'}$ of the KS test (critical value = 0.0192), two-dimensional displacement of first sensor element, root-MUSIC, true DOAs = $[-18^\circ, 5^\circ, 25^\circ]$; upper part of table: DOAs obtained by the two-dimensional stochastic collocation method (SCM) method Equation (13); bottom part of table: DOAs obtained by cubature formulas (16); the last column is the simulation time.

root-MUSIC, $\sigma_{pl} = 0.12$					
$D_{n,n'}$ for					
$P_1 = P_2$	$Q_1 = Q_2$	θ_1	θ_2	θ_3	time
7	8	0.0251	0.0039	0.0731	4.14 s
10	11	0.0102	0.0025	0.0255	5.63 s
13	14	0.0038	0.0033	0.0112	6.40 s
P	Q	θ_1	θ_2	θ_3	time
7	44	0.0134	0.0052	0.0486	0.76 s
10	99	0.0061	0.0021	0.0309	1.54 s
15	172	0.0044	0.0021	0.0219	2.22 s

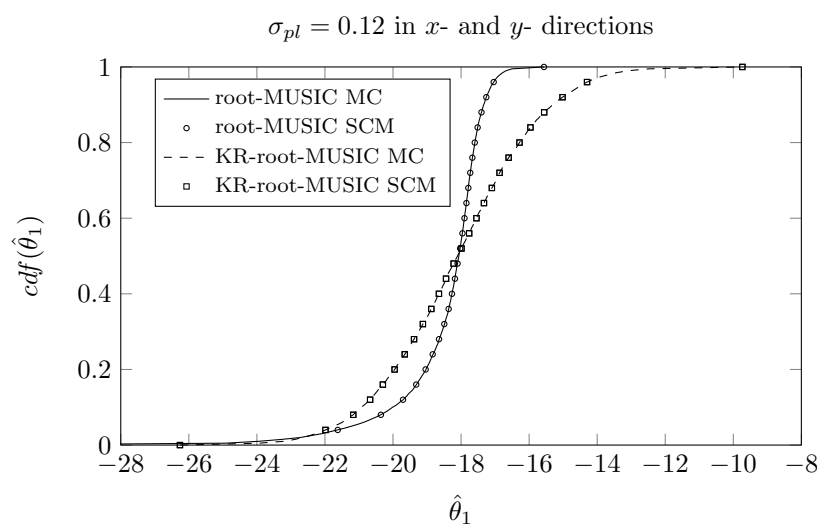
Table 4. The measured value $D_{n,n'}$ of the KS test (critical value = 0.0192), KR-root-MUSIC, true DOAs = $[-18^\circ, 5^\circ, 25^\circ]$; upper part of table: DOAs obtained by two-dimensional SCM method Equation (13); bottom part of table: DOAs obtained by cubature formulas (16); the last column is the simulation time.

KR-root-MUSIC, $\sigma_{pl} = 0.12$					
$D_{n,n'}$ for					
$P_1 = P_2$	$Q_1 = Q_2$	θ_1	θ_2	θ_3	time
7	8	0.0002	0.0002	0.0002	4.74 s
10	11	0.0001	0.0002	0.0001	6.01 s
13	14	0.0001	0.0001	0.0001	7.33 s
P	Q	θ_1	θ_2	θ_3	time
7	44	0.0002	0.0002	0.0002	0.83 s
10	99	0.0003	0.0003	0.0004	1.62 s

For the same impinging signal as in Section 5.3, we have plotted the cdfs of the different estimated DOAs, for $\sigma_{pl} = 0.12$. We have carried out an MC simulation with 10,000 realizations. The simulation time is about 128 s for the root-MUSIC algorithm and 157 s for the KR-root-MUSIC algorithm. We also apply the two-dimensional GPoC expansion (11) with degree P in both variables to obtain the cdfs for the DOAs. We calculate the weight coefficients using $Q_1 = Q_2 = P - 1$ quadrature points in the x - and y -directions using Equation (13). Tables 3 and 4 show the KS test to compare the samples obtained by the SCM method and the MC method. Furthermore, we compare this with the two-dimensional SCM theory that relies on the cubature formulas with $Q = 44$, $Q = 99$ and $Q = 172$, respectively, and with polynomial expansions of order $P = 7$, $P = 10$ and $P = 15$, respectively. The measured value of the KS test, which compares the samples of the MC method with those obtained with the cubature formulas, is also presented in Tables 3 and 4. Furthermore, the simulation times are shown in the last column of the tables. We observe that now, for the root-MUSIC algorithm, we need 14 quadrature points in each direction to satisfy the null hypothesis of the KS test ($\alpha = 0.05$). Using cubature formulas, only for the estimates of the third DOA, 172 quadrature points are insufficient to satisfy the null hypothesis. For the KR-root-MUSIC algorithm, four quadrature points in each direction are sufficient. If we use cubature formulas, $Q = 44$ quadrature points are sufficient to satisfy the null hypothesis. We also observe that, using the SCM theory, the simulation time is reduced by a factor of about 20, and using the cubature formulas, we can reduce the simulation time by a factor of 56.

In Figures 12–14, we have plotted the empirical cdf for the MC simulation and for the SCM method $P = Q - 1 = 13$. We observe that both curves almost coincide.

Figure 12. $P = Q - 1 = 13$, the cdf of $\hat{\theta}_1$ for a two-dimensional displacement of first sensor element, true DOAs = $[-18^\circ, 5^\circ, 25^\circ]$ and $(N, K) = (4, 3)$.



Define Q_p as the p -th percentile of the estimated DOA, meaning that $p\%$ of the estimates for the true DOA are below Q_p . Define $S = \frac{Q_{84} - Q_{16}}{2}$, such that 68% of the estimates lie in an interval of length $2S$ around the true DOA. For a Gaussian distribution, the value S will correspond to the dispersion. In Figures 15 and 16, we have plotted the value S as a function of σ_{pl} , for a two-dimensional displacement of the first sensor element. Again, we observe that the SCM method provides a good approximation for the estimates, both for the root-MUSIC algorithm as for the KR-root-MUSIC algorithm. Observe

that for values σ_{pl} less than 0.4, the root-MUSIC yields better estimates for the first and third DOA than the KR-root-MUSIC algorithm. However, the KR-root-MUSIC estimates the second DOA better. We furthermore observe that, for the KR-root-MUSIC algorithm, the S -values are straight lines as a function of σ_{pl} .

Figure 13. $P = Q - 1 = 13$, the cdf of $\hat{\theta}_2$ for a two-dimensional displacement of first sensor element, true DOAs = $[-18^\circ, 5^\circ, 25^\circ]$ and $(N, K) = (4, 3)$.

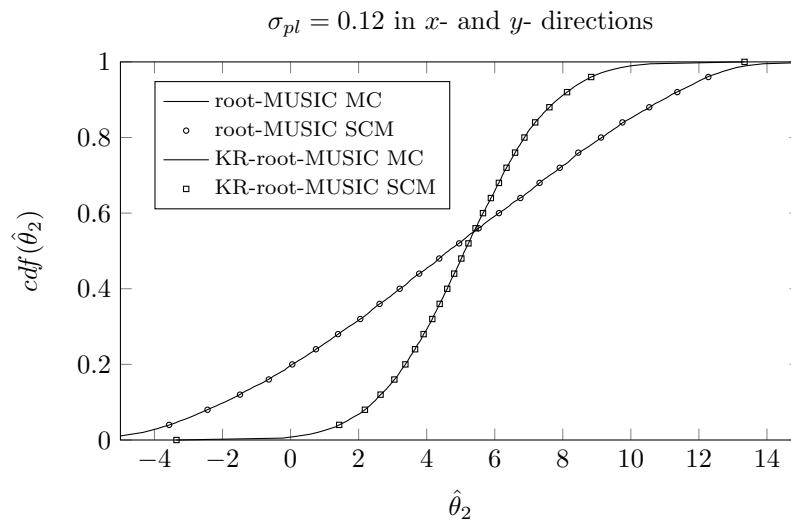


Figure 14. $P = Q - 1 = 13$, the cdf of $\hat{\theta}_3$ for a two-dimensional displacement of first sensor element, true DOAs = $[-18^\circ, 5^\circ, 25^\circ]$ and $(N, K) = (4, 3)$.

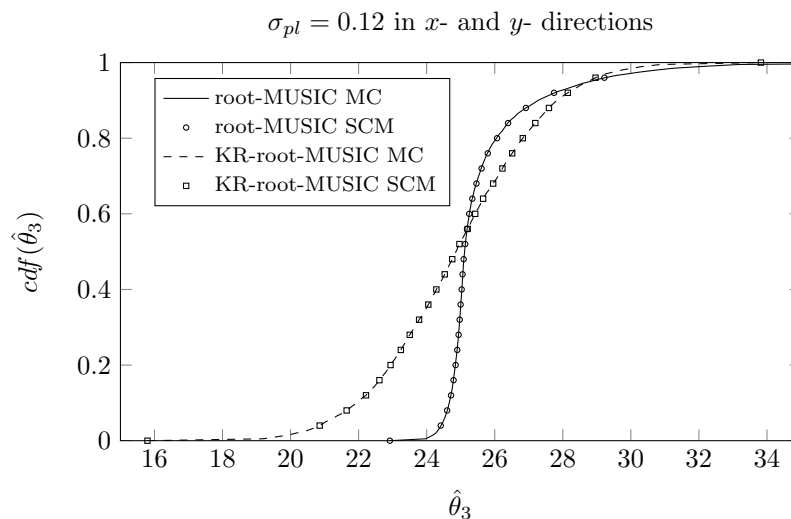


Figure 15. $(Q_{84} - Q_{16})/2$ as a function of σ_{pl} for a two-dimensional displacement of the sensor element and when applying root-MUSIC, true DOAs = $[-18^\circ, 5^\circ, 25^\circ]$ and $(N, K) = (4, 3)$.

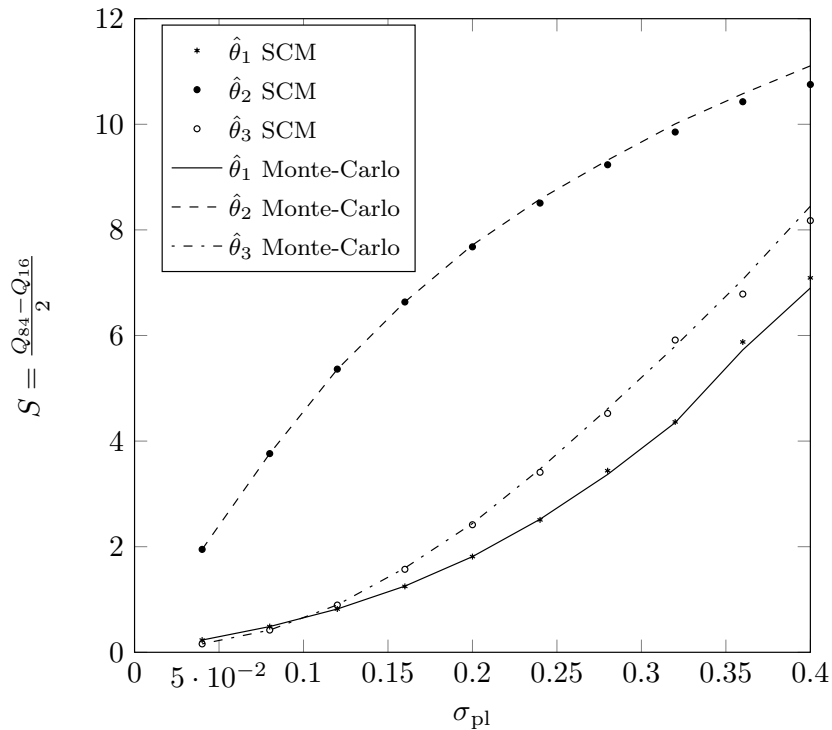


Figure 16. $\frac{Q_{84} - Q_{16}}{2}$ as a function of σ_{pl} for a two-dimensional displacement of the sensor element and when applying KR-root-MUSIC, true DOAs = $[-18^\circ, 5^\circ, 25^\circ]$ and $(N, K) = (4, 3)$.

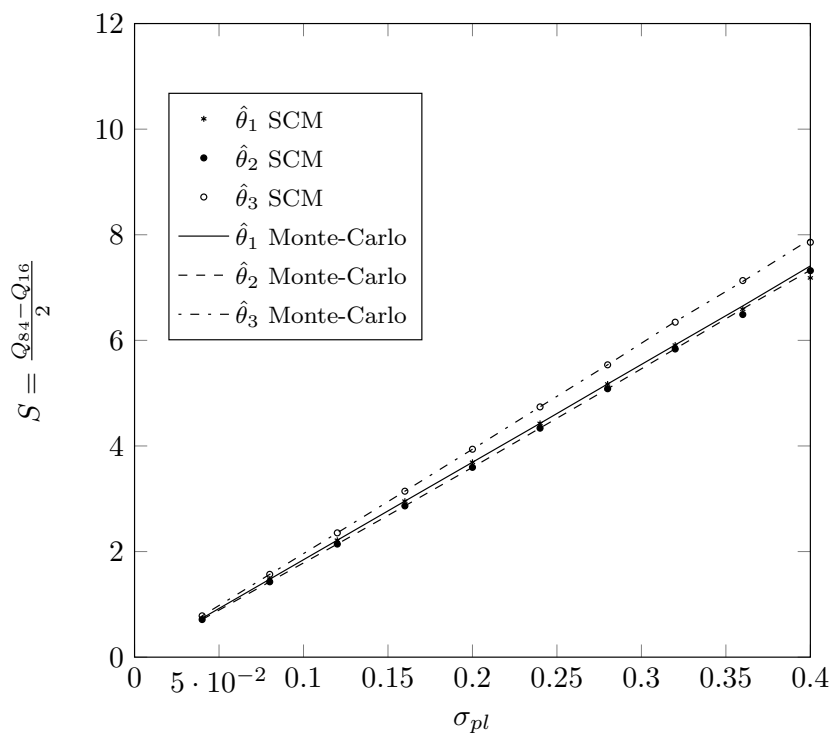
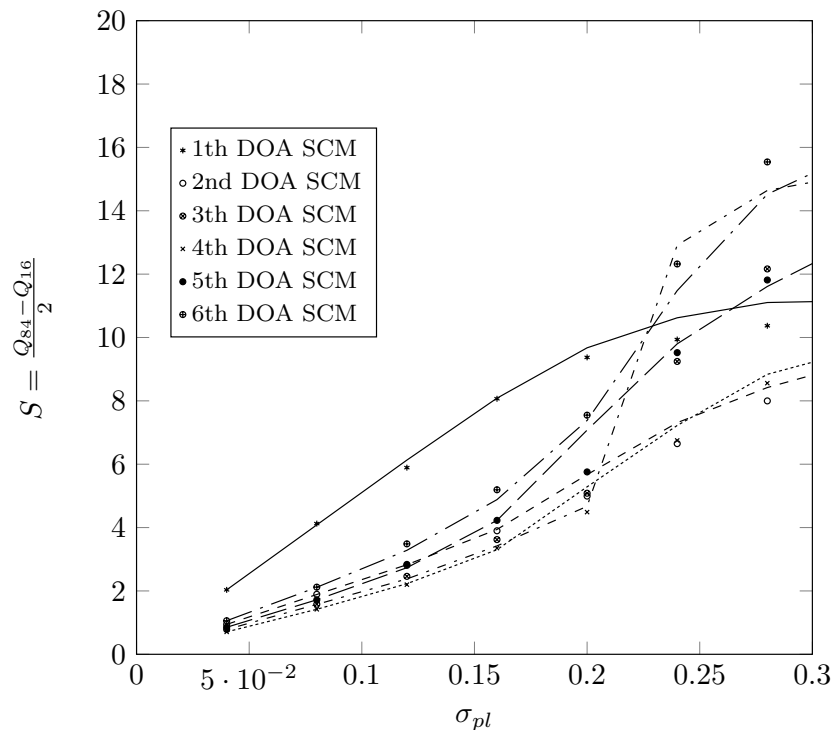


Figure 17. $\frac{Q_{84}-Q_{16}}{2}$ as a function of σ_{pl} for a two-dimensional displacement of the sensor element and when applying KR-root-MUSIC, true DOAs = $[-65^\circ, -40^\circ, -20^\circ, 10^\circ, 25^\circ, 42^\circ]$ and $(N, K) = (4, 6)$.



In Figure 17, we consider the $(N, K) = (4, 6)$ case with a displacement of the first sensor element and with the KR-root-MUSIC algorithm. The true DOAs to be estimated are $[-65^\circ, -40^\circ, -20^\circ, 10^\circ, 25^\circ, 42^\circ]$. Observe that, with the same array configuration as in Figure 16, we estimate twice as many DOAs. Looking at the $S = \frac{Q_{84}-Q_{16}}{2}$, we observe that the KR-root-MUSIC algorithm finds the first estimate for $\hat{\theta}_1 = -65^\circ$ with a large error, even for small values of σ_{pl} . The estimation error for the remainder of the DOAs starts to increase dramatically for values of σ_{pl} larger than 0.2.

6. Conclusions

In this paper, we have investigated the effect of a displacement of one of the sensor elements. We have applied the SCM formalism to efficiently predict the cdfs of the estimated DOAs, in the case that the relative displacement is Gaussian distributed. Comparison of the SCM method to the Monte-Carlo method demonstrates that SCM is an accurate method to predict the cdfs of the DOAs. For the one-dimensional case, the SCM method decreases the simulation time by a factor of more than 100. For the two-dimensional case and using cubature formulas, we can reduce the simulation time by a factor of 56. In the future, we will extend the applied method to array configurations in which more sensor elements are displaced. Moreover, the SCM will be applied to other DOA estimation algorithms and for other array configurations. The flexibility and non-intrusive character of the proposed method only requires that DOA-estimates need to be computed in the quadrature points, by use of the appropriate algorithm.

Acknowledgments

This research was partially funded by the Inter-University Attraction Poles Program initiated by the Belgian Science Policy Office.

Author Contributions

All authors contributed extensively to the study presented in this manuscript. Veronique Inghelbrecht designed the main idea, carried out the simulation, interpreted the results and wrote the paper. Jo Verhaevert, Hendrik Rogier and Tanja Van Hecke supervised the main idea and edited the manuscript. They provided many valuable suggestions to this study: Jo Verhaevert and Hendrik Rogier especially in the domain of sensors; Tanja Van Hecke especially in the domain of statistical analysis.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Ma, W.K.; Hsieh, T.H.; Chi, C.Y. DOA estimation of quasi-stationary signals with less sensors than sources and unknown spatial noise covariance: A Khatri-Rao subspace approach. *IEEE Trans. Signal Process.* **2010**, *58*, 2168–2180.
2. Brown, G.; McClellan, J. Performance bound analysis of a hybrid array. *IEEE Trans. Signal Process.* **1996**, *44*, 74–82.
3. Tam, P.; Wong, K. Cramer-Rao bounds for direction finding by an acoustic vector sensor under nonideal gain-phase responses, noncollocation, or nonorthogonal orientation. *IEEE Sens. J.* **2009**, *9*, 969–982.
4. Tam, P.; Wong, K. An hybrid Cramer-Rao bound in closed form for direction-of-arrival estimation by an “acoustic vector sensor” with gain-phase uncertainties. *IEEE Trans. Signal Process.* **2014**, *62*, 2504–2015.
5. Friedlander, B.; Weiss, A. Effects of model errors on waveform estimation using the MUSIC algorithm. *IEEE Trans. Signal Process.* **1994**, *42*, 147–155.
6. Lo, J.; Marple, S. Observability conditions for multiple signal direction finding and array sensor localization. *IEEE Trans. Signal Process.* **1992**, *40*, 2641–2650.
7. Ratnarajah, T.; Manikas, A. An H^∞ approach to mitigate the effects of array uncertainties on the MUSIC algorithm. *IEEE Signal Process Lett.* **1998**, *5*, 185–188.
8. Viberg, M.; Swindlehurst, A. Analysis of the combined effects of finite samples and model errors on array processing performance. *IEEE Trans. Signal Process.* **1994**, *42*, 3073–3083.
9. Weiss, A.; Friedlander, B. “Almost blind” steering vector estimation using second-order moments. *IEEE Trans. Signal Process.* **1996**, *44*, 1024–1027.

10. Rogier, H.; Boeykens, F. An efficient technique based on polynomial chaos to model the uncertainty in resonance frequency of textile antennas due to bending. *IEEE Trans. Antennas Propag.* **2014**, *AP-62*, 1253–1260.
11. Friedlander, B. A sensitivity analysis of the MUSIC algorithm. *IEEE Trans. Acoust. Speech Signal Process.* **1990**, *38*, 1740–1751.
12. Weiss, A.; Friedlander, B. Effects of modeling errors on the resolution threshold of the Music algorithm. *IEEE Trans. Signal Process.* **1994**, *42*, 1519–1526.
13. Chevalier, P.; Ferreol, A. On the virtual array concept for the fourth-order direction finding problem. *IEEE Trans. Signal Process.* **1999**, *47*, 2592–2595.
14. Pal, P.; Vaidyanathan, P. Nested arrays: A novel approach to array processing with enhanced degrees of freedom. *IEEE Trans. Signal Process.* **2010**, *58*, 4167–4181.
15. Sidiropoulos, N.; Bro, R. Parallel factor analysis in sensor array processing. *IEEE Trans. Signal Process.* **2000**, *48*, 2377–2388.
16. Feng, D.; Bao, M.; Ye, Z.; Guan, L.; Li, X. A novel wideband DOA estimator based on Khatri-Rao subspace approach. *Signal Process.* **2011**, *91*, 2415–2419.
17. Valaee, S.; Kabal, P. Wideband array processing using a two-sided correlation transformation. *IEEE Trans. Signal Process.* **1995**, *43*, 160–172.
18. He, Z.Q.; Shi, Z.P.; Huang, L.; So, H. Underdetermined DOA estimation for wideband signals using robust sparse covariance fitting. *IEEE Signal Process. Lett.* **2013**, *22*, 435–439.
19. He, Z.Q.; Shi, Z.P.; Huang, L. Covariance sparsity-aware DOA estimation for nonuniform noise. *Digit. Signal Process.* **2014**, *28*, 75–81.
20. Cao, M.Y.; Huang, L.; Qian, C.; Xue, J.J.; So, H. Underdetermined DOA estimation of quasi-stationary signal via Khatri-Rao structure for uniform circular array. *Signal Process.* **2015**, *106*, 41–48.
21. Guo, X.; Li, B.; Chu, L.; Wang, D. Near-Field source localization in complex indoor environment uniform circular array. In Proceedings of IEEE China Summit & International Conference on Signal and Information Processing, Xi'an, China, 9–13 July 2014; pp. 412–415.
22. Palanisamy, P.; Kishore, C. 2-D DOA estimation of quasi-stationary signals based on Khatri-Rao subspace approach. In Proceedings of IEEE-International Conference on Recent Trends in Information Technology, Chennai, India, 3–5 June 2011; pp. 798–803.
23. Chan, F.; So, H.; Huang, L.; Huang, L.T. Underdetermined direction-of-departure and direction-of-arrival estimation in bistatic multiple-input multiple-output radar. *Signal Process.* **2014**, *104*, 284–290.
24. Barabell, A. Improving the resolution performance of eigenstructured-based direction-finding algorithms. In Proceedings of IEEE International Conference on ICASSP '83 Acoustics, Speech, and Signal Processing, Boston, MA, USA, 14–16 April 1983; pp. 336–339.
25. Xiu, D. Fast numerical methods for stochastic computations: A review. *Commun. Comput. Phys.* **2009**, *5*, 242–272.
26. Cameron, R.; Martin, W. Transformations of Wiener integrals under translations. *Ann. Math.* **1944**, *45*, 386–396.

27. De Staelen, R.; Crevecoeur, G. A sensor sensitivity and correlation analysis through polynomial chaos in the EEG problem. *IMA J. Appl. Math.* **2014**, *79*, 163–174.
28. Eldred, M.S.; Webster, C.G. Evaluation of non-intrusive approaches for Wiener-Askey generalized polynomial chaos. In Proceedings of 49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Schaumburg, IL, USA, 7–10 April 2008.
29. Caflisch, R. Monte Carlo and quasi-Monte Carlo methods. *Acta Numer.* **1998**, *7*, 1–49.
30. Bagci, H.; Yucel, A.; Hesthaven, J.; Michielssen, E. A fast Stroud-based collocation method for statistically characterizing EMI/EMC phenomena on complex platforms. *IEEE Trans. Electromagn. Compat.* **2009**, *51*, 301–311.
31. Cools, R. An encyclopaedia of cubature formulas. *J. Complex.* **2003**, *19*, 445–453.
32. Stroud, A. *Approximate Calculation of Multiple Integrals*; Prentice-Hall: Upper Saddle River, NJ, USA, 1971.
33. Cools, R.; Pluym, L.; Laurie, D. Cubpack++: A C++ package for automatic two-dimensional cubature. *ACM Trans. Math. Softw. (TOMS)* **1997**, *23*, 1–15.
34. Marsaglia, G.; Tsang, W.; Wang, J. Evaluating Kolmogorov's distribution. *J. Stat. Softw.* **2003**, *8*, 1–4.

© 2014 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/4.0/>).