TR-WEL-0001 　　　　　　　　　　　　OO3

Direction-of Arrival Estimation with a Seven-Element Regular-Hexagonal Shaped Espar Antenna Employing the ESPRIT Algorithm

Emmanuelle Chu, Eddy Taillefer
and Takashi Ohira

2004.4.2

Abstract

The electronically steerable parasitic array radiator (ESPAR) antenna is a single-output array antenna. As the output signal of each element of the antenna cannot be observed, direct application of most of the algorithms designed for the conventional array antenna is impractical. In this report, a technique for Direction-of-Arrival (DoA) estimation for the ESPAR antenna based on the ESPRIT (Estimation of Signal Parameters via Rotational Invariance Techniques) algorithm combined with the reactance domain technique is proposed. ESPRIT is a popular subspace-based approach for high-resolution signal parameters with applications to DoA estimation. By exploiting the displacement invariance provided by the 7-element regular-hexagonal shaped ESPAR antenna, DoA estimations can be performed with the ESPRIT algorithm. Simulations show that the performance of the algorithm is linked to the value of the angle of arrival. Because of the subarray division imposed on the ESPAR antenna by the ESPRIT algorithm, DoA estimations are restricted to half of the full azimuthal plane sector. Nevertheless, by employing the proposed method, up to 3 signals impinging simultaneously can be estimated. Moreover, good results can be obtained: for one signal impinging on the ESPAR antenna, simulations show that there is a probability of more than 80 % of achieving DoA estimations within $\pm 2°$ accuracy for an angle in the range of 50° to 130°.

Contents

List of Figures

List of Tables

Glossary and Keywords

Glossary

ATR: Advanced Telecommunication Research

DoA: Direction of Arrival

WACNet : Wireless Ad-hoc Community Network

ESPAR: Electronically Steerable Parasitic Array Radiator

Reactance-domain technique

MUSIC: MUltiple SIgnal Classification

ESPRIT: Estimation of Signal Parameters via Rotational Invariance Techniques

RF: Radio Frequency

SNR: Signal to Noise Ratio

IEEE: Institute of Electrical and Electronics Engineers

IEICE: Institute of Electronics, Information and Communication Engineers

Key words

Antenna, Reactance, ESPRIT, MUSIC, High-resolution, Parameters Estimation, invariance, Signal processing, *ad hoc* networks.

Notations

In this part the mathematical notations used in this report for all equations and calculus, are settled.

A scalar or complex value is noted: e, e_n, E.

A vector is noted: $\mathbf{e}$.

A matrix is noted: $\mathbf{E}$.

The identity matrix of order M is noted: $\mathbf{I}_M$.

The superscript $*$ is the conjugate operation.

The superscript T is the transpose operation for vector or matrix.

The superscript H is the complex conjugate transpose operation for vector or matrix.

The $\mathbf{E}[\ ]$ operator realizes the stochastic mean of a process.

Chapter 1

Introduction

With the development of multimedia and mobile communications, the variety of networks and terminals is increasing and communications environments are becoming more complex. The Adaptive Communication Research (ACR) laboratories are conducting research on key technologies for the construction and evaluation of adaptive systems with the aim of making systems adaptive to various operating and user environment.

Recently, the concept of the Wireless Ad-hoc community Network (WACNet) has shown good promise. These kind of networks consist of mobile nodes with wireless transceivers and are expected to support communication anytime and anywhere.

The application of advanced adaptive techniques has contributed to the development of high-performance receiving adaptive antennas with the capability of automatically eliminating surrounding interferences. Conventionally, it has been thought that only digital technology could provide these advantages. ATR has proposed Electronically Steerable Parasitic Array Radiator (ESPAR) antennas as a means of reviving analog adaptive beamforming, which provides a dramatically simplified architecture that results in significantly lower power dissipation and fabrication cost than are common with digital beamforming antennas.

This report presents a solution to apply the conventional Estimation of signal parameters via rotational invariance technique (ESPRIT) to the ESPAR antennas.

After introducing the main topics related to this work, the WACNet concept, and then the ESPAR antenna features will be described. The new method we proposed is combining the reactance-domain approach and the ESPRIT algorithm. So, before the explanation of the accomplished work on the ESPRIT algorithm, the reactance-domain technique will be detailed. Finally, simulations for performance evaluation of this new technique will be conducted.

Chapter 2

ATR Adaptive Communications Research Laboratories

2.1 Introduction to ATR

Advanced Telecommunications Research Institute International[1] (ATR) was established in 1986 with support from industry, academia and government. It operates with the following objectives:

- to undertake basic, innovative research on telecommunications,

- to coordinate joint research project involving industry, academia and government,

- to implement ATR research activities on a global basis,

- to play a guiding role in the Kansai Science City based on experience gained as the first research institute.

Since its foundation, ATR conducted research activities with financial contributions from KTC, the Japan Key Technology Center, and many private companies to its research laboratories. In the middle of fiscal year 2001, the research funding system was shifted from KTC to a governmental organization called TAO, the Telecommunications Advancement Organization of Japan, which funds contracted research topics in telecommunications for a better society in the 21st century.

The reorganized structure leads to the four following actual departments:

- Adaptive Communications Research Laboratories (ACR),

- Human Information Science Laboratories (HIS),

- Media Information Science Laboratories (MIS),

- Spoken Language Translation Research Laboratories (SLT).

---

[1] http://www.atr.co.jp

2.2 ACR Laboratories

ACR laboratories are engaged in research on technologies for the construction and evaluation of adaptive systems with the aim of making systems adaptive to various operating and user environments.

ACR is researching segmentation algorithms for broader use of wireless resources, routing, media access control, and adaptive antennas with the aim of realizing "WACNet", an autonomous distributed network which does not need infrastructure.

"Networking Anywhere" is technology components for autonomous distributed networks, aimed at supporting free communication by data transmission in ad-hoc gatherings of people.

2.2.1 Organisation

The aim of ACR laboratories is to develop key technologies for wireless ad hoc networks and to confirm network performance in real world environment. Here after is depicted the organisation of the laboratories.
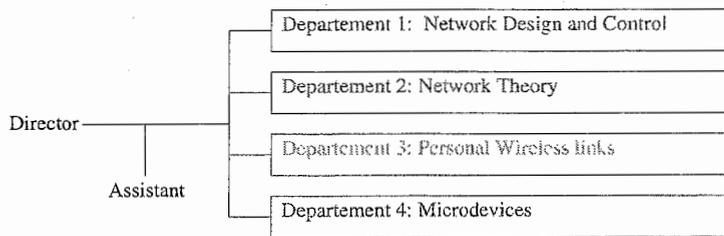
Figure 2.1: Organisation of ACR laboratories

2.2.2 Research domains

- Network and Design control department:
  They focus on technologies for the construction of ad hoc networks, such as routing, media access and adaptive quality management, and applications specific to ad hoc networks.

- Network Theory department:
  Comprehensive studies of wireless ad hoc networks are conducted to elucidate general principles and rules underlying each network layer, from physical layer to application layer, and the behavior of the whole system.

- Personal Wireless Links:
  Research is being carried out on smart antennas such as ESPAR antennas which will play a key role in wireless ad hoc networks.

- Microdevices:
  Research on fundamental technologies for semiconductor functional materials

4

and devices are being conducted in order to realize flexible, large-capacity communication links.

I joined the team of researchers of the department 3 during my six months internship, of which, the content is described in the following of this report.

Chapter 3

Objective and Background

3.1 Main Objective

Adaptive array antennas are an emerging technology that has gained much attention due to its ability to significantly increase the performance of wireless systems[1]. Recently the ESPAR antenna has been proposed for compact adaptive beamforming applied to wireless communication systems[2, 3]. Since the ESPAR can autonomously steers its beam toward the arrival direction of desired radio waves and steers nulls toward the undesired interfering waves[4], DoA estimation is a prior step to any adaptive control of the antenna. In this report, to tackle the DoA estimation topic, we present a technique for direction finding which combines the ESPRIT algorithm and the reactance-domain technique.

3.2 Previous Techniques

Wireless *ad hoc* networks[5] which consist of mobile nodes with wireless transceivers, are expected to support communication at anytime and anywhere. Since the infrastructure is free, the only cost involved depend on the user terminals. So, if an adaptive antenna, such as ESPAR antenna, can be implemented in the user terminal, a large among of battery power can be saved (in comparison with an omni-directional antenna).

Consequently, the first technique [6, 7] applied to the ESPAR antenna consists of a radio-wave detector that points out the angle for which the received signal has the maximum amplitude. Twelve narrow beams are formed and steered in 0° to 360° angles with a 30° step. 30° is the lowest bound that make possible to differentiate two adjacent beams. For each angle, the ESPAR antenna detects the received signal while the wave is arriving. Finally, the angle of maximum amplitude is searched and therefore, indicated with one of the twelve LEDs (Light Emitting Diode).

Then, the reactance-domain MUSIC algorithm was developed for ESPAR antennas [8, 9]. Recently an approach for estimating DoA based on PPCC (Power Pattern Cross Correlation) was proposed [10, 11]. Actually, some research are also being conducted on SAGE (Space Alternating Generalized Expectation) maximization algorithm for ESPAR antennas [12].

7

3.3 The New Approach

Before giving more formal explanation of the accomplished work , this is a brief introduction of the framework methodology. Indeed, this new technique, combines the ESPRIT algorithm and the reactance-domain approach to achieve the DoA estimation [13, 14].

3.3.1 Reactance-Domain Technique

In conventional adaptive arrays, the correlation matrix is created by measuring the signals outputs on each element of the array antenna but in the case of the ESPAR antenna only the single-port output can be observed and processed. So how could a correlation matrix be created ?
The reactance-domain approach [15] is a solution to cope with the ESPAR antenna specificity.

3.3.2 Conventional ESPRIT Algorithm

The ESPRIT (Estimation of Signal Parameters via Rotational Invariance Techniques) [16] algorithm, yields parameter estimates such as Direction-of-Arrival (DoA) without requiring knowledge of the array geometry and sensor element characteristics (steering vectors,...), thus it eliminates the need for sensor array calibration. The method is a subspace-based approach for source parameter estimation for planar wavefronts. To achieve DoA estimations, ESPRIT requires a translational invariance within the sensor design.

3.4 Work Approach Possibilities

After studying different documents related to the ESPRIT algorithm and the ESPAR antenna, two main axis of research were highlighted. The first is the direct application of the ESPRIT algorithm combined with the reactance-domain technique. The second is to adapted the CUBA-ESPRIT technique[17] for the ESPAR antennas. In a first consideration, *i.e.* in this present work report, direct application of the ESPRIT algorithm to ESPAR antennas has been conducted.

3.5 Methodology and Synopsis of Work

Date	Work contents	Comments
April	**Documentation work:** • The ESPAR antenna feature, comparison with conventional arrays • The Reactance-domain MUSIC algorithm for ESPAR antennas • The conventional ESPRIT algorithm • The CUBA-ESPRIT algorithm	We first tried to adapt the ESPRIT algorithm to the ESPAR antennas.
May to June	**Implementation of the ESPRIT algorithm for different antennas, to check its applicability** • a circular conventional antenna with no central element • a circular conventional antenna with central element • an ESPAR antenna	We applied the ESPRIT algorithm to antennas with a similar geometry to the ESPAR antenna in order to see if DoA estimations can be obtained with conventional antennas
June to July	**Simulation part** Simulations were conducted for performance evaluation according to different parameters such as the SNR rate or the number of samples and so on. **Writing papers for national conferences:** • IEICE Antenna and Propagation Technical Meeting, Sendai, August 25th[13]. • IEICE General Conference, Niigata, September, 23rd[14].	Oral presentations.
August to September	Preparation of the oral presentations Focus on the CUBA-ESPRIT algorithm which includes spatial smoothing technique	DoA estimation of coherent signals

Table 3.1: Intership progress & schedule

Chapter 4

WACNet and the ESPAR Antenna

In this chapter, the WACNet concept will be presented, as the ESPAR antenna application background. Next, the ESPAR antenna configuration and modelization will be detailed. The signal model used for the ESPAR antenna will also be introduced .

4.1 Wireless Ad-Hoc Community Network - Concept and Features

Recently, the concept of Wireless Ad-Hoc Community Networks (WACNet) [5] has become a hot topic. Such network can be considered as a means of communications among portable user terminals that temporally meet, where distance and time come close yet easy connection to a network infrastructure is not possible. For example, Bluetooth[1] has been developed to connect between mobile handy phones, headsets, PCs, and other devices using the ISM (Industrial, Scientific and Medical) band (2.4 GHz). Bluetooth has great possibilities to enable wireless ad-hoc networks. The concept of WACNet is proposed in order to have more scalability, more speeds, and low output power level, that makes it very attractive for mobile network use.

The WACNet consists of portable user terminals only and has no infrastructure such as hosts, base stations, switches, and hubs. It is therefore expected to be applicable anywhere; not only in fixed space such as a office company but also in a moving space such as a bus or train. It can also be used in a disaster stricken area where the existing telecommunications infrastructure has been damaged. The WACNet has scalability, i.e., the ability to accept a large number of users terminals because of autonomous segmentation and routing. The key technologies developed to achieve the WACNet are routing schemes, Space Division Multiple Access (SDMA), user terminals, microwave signal processing, and Electronically Steerable Parasitic Array Radiator (ESPAR) antennas.

---

[1]See at the official bluetooth website: *http://bluetooth.com/*.

4.1.1 Routing Schemes

WACNet routing is carried out by each user terminal autonomously. This means that each user terminal has switching and relay functions and routes are made by the user terminals through the use of these functions voluntarily. In these routing schemes, dynamic topologies such as the number of terminals and the relative locations and velocities among terminals must be considered.

4.1.2 Space Division Multiple Access (SDMA)

To achieve scalability, the adoption of dynamic network segmentation is considered. With dynamic network segmentation, a network is divided into a number of segments consisting of some terminals. As one channel is able to be used repeatedly in the different segments, the SDMA technique is very useful since it enables the limited network resources to be effectively utilized. Packets from one terminal to terminals in the same segment are delivered directly and packets to terminals in different segments are delivered via relay terminals.

4.1.3 User Terminals

Since the WACNet is infrastructure free, the only costs involved depend on the user terminals. To make the WACNet popular it is necessary to miniaturize and reduce the cost and weight of the user terminals. One of the important technologies for terminal weight reduction is to reduce the battery weight because the battery occupies a significant percentage of the weight of the user terminal. Accordingly, if an adaptive antenna can be implement in the user terminal, a large among of battery power can be saved because the adaptive antenna has a higher gain than an omnidirectional antenna, and can therefore effectively reduce the RF (Radio Frequency) power necessary to transmit packets. The adaptive antenna can also suppress co-channel interference and can therefore enlarge the capacity of a network, i.e., the number of terminals that can be accommodated in the same network.

4.1.4 Microwave Signal Processing

Adaptive antennas essentially have a great advantage in saving RF power. However, the high cost and high power consumption in their signal processing units still prevent their use from spreading to the general population in uses such as user teminals. The present adaptive antennas are Digital Beamforming (DBF) antennas (see Figure 4.1), which need the same number of RF high-power or low-band amplifiers, frequency converters, and D/A or A/D converters as the elements in conventional array antennas.

Therefore, the implementation of adaptive antennas in user terminals had been said to be impractical. This is now possible with the "Microwave signal processing" concept using RF Beamforming (RBF) (see Figure 4.2) instead of DBF.

This new beamforming architecture has been proposed to achieve the ultimate reductions in the size, weight, power dissipation, and fabrication cost of adaptive antennas. The operating principle of this architecture is based upon electromagnetic

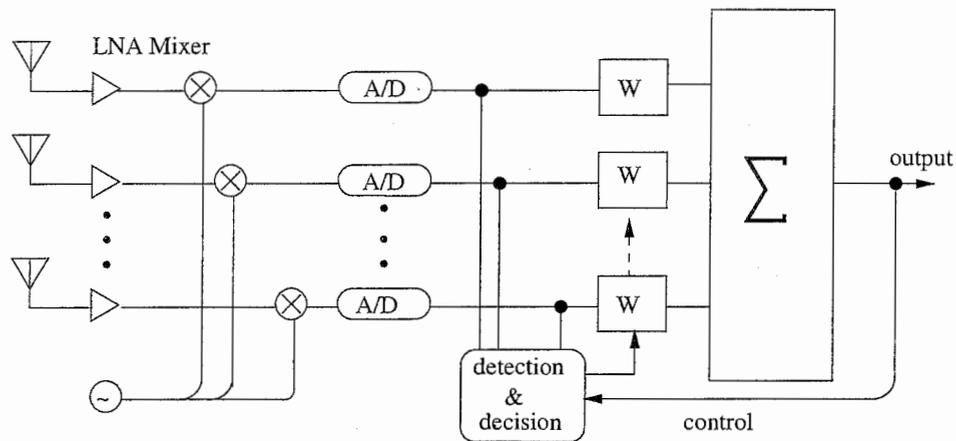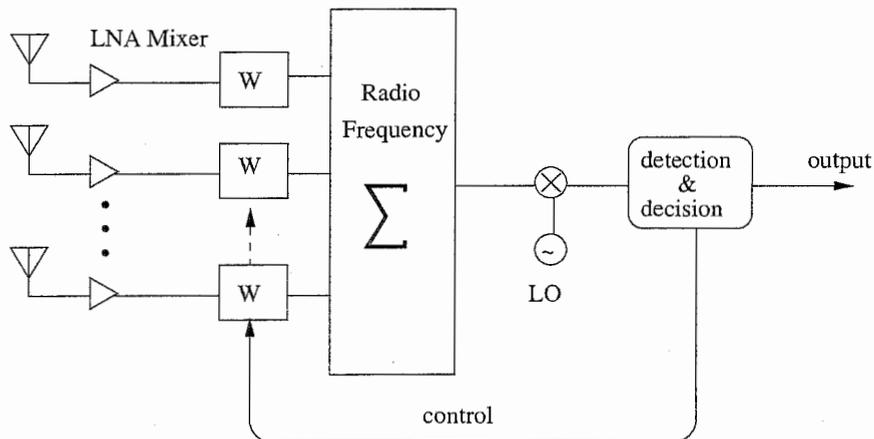Figure 4.1: DBF - Digital Beamforming

Figure 4.2: RBF - RF Beamforming

coupling among array elements. It is called "Aerial Beamforming" (ABF) (see Figure 4.3) because signal combining is carried out in space, not in circuits. Since this array requires only one RF port to feed, the RF circuit scale is drastically reduced compared with other configurations.
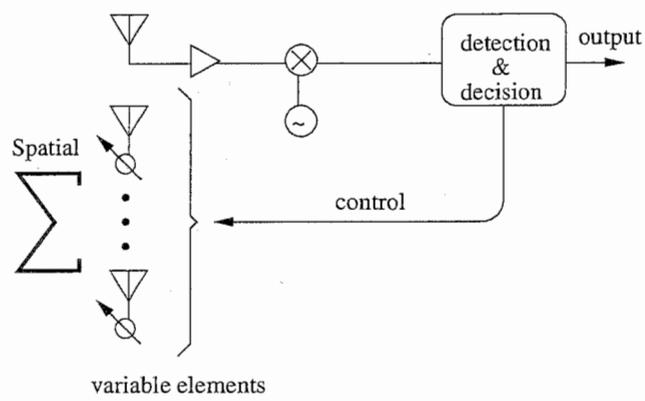
Figure 4.3: ABF - Aerial Beamforming

4.2   The ESPAR Antenna and the Signal Model

As a practical hardware realization of the ABF concept, the Electronically Steerable Parasitic Array Radiator (ESPAR) antenna [18, 3] has been proposed. In this section, the ESPAR antenna configuration is reviewed and its model detailed. Next the signal model used for the ESPAR antenna in the simulations is given [18].

4.2.1   The ESPAR Antenna Configuration

The ESPAR antenna consists of $(M + 1)$ elements with $M = 6$ in our configuration, which is depicted in Figure 4.4.
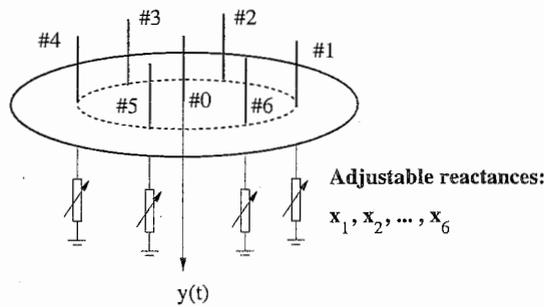
Figure 4.4: Configuration of an ESPAR antenna

The ESPAR antenna is a single-port monopole array antenna with one central active radiator surrounded by M parasitic elements.

Figure 4.4 shows the geometry of a 7-element ESPAR antenna (i.e. $M = 6$). The six parasitic elements that surround the active monopole symmetrically are loaded with reactances x_m $(m = 1, 2, \ldots, M)$. The radiation pattern of the antenna can be changed by adjusting these reactance values. The vector denoted

$$\mathbf{x} = [x_1, x_2, \ldots, x_M]^T, \tag{4.1}$$

is the reactance vector.

We now briefly formulate the output of the ESPAR antenna, in the far-field assumption. The RF signal impinging on the antenna is denoted as $\mathbf{s}(t) = [s_0(t), s_1(t), \ldots, s_M(t)]$ where $s_m(t)$ is the signal arriving on the mth element. Then y(t) can be expressed as [18]:

$$y(t) = \mathbf{i}^T \mathbf{s}(t), \tag{4.2}$$

where $\mathbf{i} = [i_0, i_1, \cdots, i_m, \cdots, i_M]^T$ is the RF current vector with the component i_m appearing on the mth element.

4.2.2 the ESPAR Antenna Model

Next, a representation of the current vector $\mathbf{i}$ is given, as a function of the reactance vector. For convenience, we derive the model of the ESPAR antenna in a transmit mode. The theorem of reciprocity tells us that the received-mode radiation pattern of an antenna is equal to that of the transmit-mode. Therefore all the following results are also available in the receive-mode.

The RF voltage of the central element, can be imposed as function of the RF current appearing on this element :

$$v_0 = V_s - Z_0 i_0, \tag{4.3}$$

where Z_0 is the output impedance, and V_s is the internal source RF voltage. In the same way we impose the RF voltage on the reactance x_m to be

$$v_m = -j x_m i_m, \quad m = 1, 2, \ldots, M. \tag{4.4}$$

So we can form the RF voltage vector $\mathbf{v} = [v_0, \cdots, v_m, \cdots, v_M]^T$. Then, the RF voltage vector can be explained as a function of $\mathbf{i}$:

$$\mathbf{v} = \begin{bmatrix} v_0 \\ v_1 \\ \vdots \\ v_m \\ \vdots \\ v_M \end{bmatrix} = \begin{bmatrix} V_s - Z_0 i_0 \\ -j x_1 i_1 \\ \vdots \\ -j x_m i_m \\ \vdots \\ -j x_M i_M \end{bmatrix} = V_s \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \end{bmatrix} - \mathbf{X} \mathbf{i}, \tag{4.5}$$

where $\mathbf{X} = diag[Z_0, j x_1, \ldots, j x_M]$ is called the reactance matrix. In addition, the RF current vector $\mathbf{i}$ and RF current voltage $\mathbf{v}$ have the relationship :

$$\mathbf{i} = \mathbf{Y} \mathbf{v}, \tag{4.6}$$

where $\mathbf{Y}$ is the admittance matrix, with

$$\mathbf{Y} = \begin{bmatrix} y_{00} & \cdots & & & y_{0M} \\ & \ddots & & & \\ \vdots & & y_{kk} & & \vdots \\ & & & \ddots & \\ y_{M0} & \cdots & & & y_{MM} \end{bmatrix} \tag{4.7}$$

and y_{kl} represents the mutual admittance between the elements k and l for $(k, l) \in [0, M]^2$.

By replacing the voltage vector in (4.6) by its expression in (4.5), the current vector becomes :

$$\mathbf{i} = V_s (\mathbf{I}_{M+1} + \mathbf{Y} \mathbf{X})^{-1} \mathbf{y}_0 \tag{4.8}$$

where $\mathbf{I}_{M+1}$ is the identity matrix of order $M+1$. Moreover, the $(M+1)$-dimensional vector $\mathbf{y}_0$ and $(M+1)$ by $(M+1)$ matrix $\mathbf{Y}$ are constant and determined by the

structure of the ESPAR antenna. Notice that this current vector $\mathbf{i}$, and thus $y(t)$ is a function of the reactance vector $\mathbf{x}$ of Eq. (4.1).

By the reciprocity theorem, it holds, similary to conventional array antennas, that

$$\forall (k, l) \in [0, M]^2, \quad y_{kl} = y_{lk}. \tag{4.9}$$

In addition, the cyclic symmetry of the elements of the ESPAR antenna implies

$$
\begin{aligned}
y_{11} &= y_{22} = y_{33} = y_{44} = y_{55} = y_{66} \\
y_{01} &= y_{02} = y_{03} = y_{04} = y_{05} = y_{06} \\
y_{12} &= y_{23} = y_{34} = y_{45} = y_{56} = y_{61} \\
y_{13} &= y_{24} = y_{35} = y_{46} = y_{51} = y_{62} \\
y_{14} &= y_{25} = y_{36}
\end{aligned}
\tag{4.10}
$$

Finaly, equations (4.9) and (4.10) imply that the admittance matrix $\mathbf{Y}$ is determined by only 6 components of the mutual admittances [19, 20] y_{00}, y_{10}, y_{11}, y_{21}, y_{31} and y_{41}. The values of these 6 components depend on the physical structure of the antenna and are constant.

Summarizing the above explanation, we write the admittance matrix $\mathbf{Y}$ as

$$
\mathbf{Y} =
\begin{bmatrix}
y_{00} & y_{10} & & \ddots & & & y_{10} \\
y_{10} & y_{11} & y_{21} & y_{31} & y_{41} & y_{31} & y_{21} \\
& y_{21} & \ddots & \ddots & \ddots & \ddots & y_{31} \\
\vdots & y_{31} & \ddots & \ddots & \ddots & \ddots & y_{41} \\
& y_{41} & \ddots & \ddots & \ddots & \ddots & y_{31} \\
& y_{31} & \ddots & \ddots & \ddots & \ddots & y_{21} \\
y_{10} & y_{21} & y_{31} & y_{41} & y_{31} & y_{21} & y_{11}
\end{bmatrix}
. \tag{4.11}
$$

Similarly, the $\mathbf{y}_0$ vector can be rewritten as $\mathbf{y}_0 = [y_{00}, y_{10}, y_{10}, \ldots, y_{10}]^T$.

The $\mathbf{i}$ current vector can also be formulated as [19]

$$\mathbf{i} = V_s(\mathbf{Z} + \mathbf{X})^{-1}\mathbf{u}_0, \tag{4.12}$$

where the $(M+1)$-dimensional vector $\mathbf{u}_0$ is taken as $[1, 0, \ldots, 0]^T$ and the $(M+1)$ by $(M+1)$ impedance matrix $\mathbf{Z} = \mathbf{Y}^{-1}$.

It should be emphasized that the signal vector $\mathbf{s}(t)$ impinging on the elements of the ESPAR antenna is not measurable. This differs from the conventional adaptive array antennas where the received signal vector on the elements is observed. For the ESPAR antenna, only the single-port output $y(t)$ can be measured.

4.2.3   The Signal model

First of all, we give a steering vector of the ESPAR antenna. Consider the $M + 1$-element ESPAR antenna as shown in Figure 4.5.

The m-th element is placed at an angle $\phi_m = \frac{2\pi}{M}(m - 1)$, $(m = 1, 2, \cdots, M)$ relative to an arbitrary axis. When an incoming wavefront is impinging on the
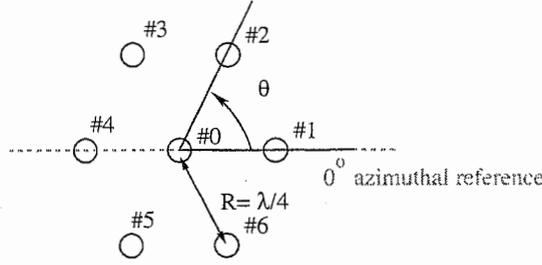
17

Figure 4.5: Top view of a hexagonal shaped ESPAR antenna.

antenna from a Direction-of-Arrival (DoA) of θ relative to the same reference axis, there is a spatial delay of $R\cos(\theta-\phi_m)$ between the signals received at the pair of the m-th element and the 0-th element. Considering the incident signal has a wavelength of λ, this spatial delay can be expressed as an electrical angular difference defined by $\frac{2\pi}{\lambda}R\cos(\theta-\phi_m)$. For the ESPAR antenna the radius is $R=\frac{\lambda}{4}$ then the steering vector for a DoA of θ can be expressed as [18]

$$
\mathbf{a}(\theta) = \begin{bmatrix} 1 \\ e^{j\frac{\pi}{2}\cos(\theta-\phi_1)} \\ e^{j\frac{\pi}{2}\cos(\theta-\phi_2)} \\ \vdots \\ e^{j\frac{\pi}{2}\cos(\theta-\phi_M)} \end{bmatrix}. \tag{4.13}
$$

Suppose there are a total number of D transmitted signals $u_d(t)$ with direction-of-arrivals (DoAs) θ_d $(d=1,\ldots,D)$. Let $s_m(t)$ $(m=0,1,\ldots,M)$ denote the RF signal impinging on the m-th element of the antenna, and let $\mathbf{s}(t)$ be the column vector with m-th component $s_m(t)$. Then, the column vector $\mathbf{s}(t)$ may be expressed as

$$
\mathbf{s}(t) = \sum_{d=1}^{D} \mathbf{a}(\theta_d)u_d(t), \tag{4.14}
$$

where $\mathbf{a}(\theta_d)$ is the steering vector depending on the structure of the ESPAR antenna for the DoA of θ_d.

According to this signal model, we can give the output of the ESPAR antenna as [19]

$$
y(t) = \mathbf{i}^T\mathbf{s}(t) + n(t) = \sum_{d=1}^{D} \mathbf{i}^T\mathbf{a}(\theta_d)u_d(t) + n(t), \tag{4.15}
$$

where $n(t)$ is an additive white Gaussian noise and $\mathbf{i}$ is the RF current vector.

Compared with conventional array antennas, ESPAR antenna has low hardware complexity, low cost and low power consumption. The ESPAR antenna, does not need RF-amplifiers, bandpass filters, A/D converters for each element like in conventional adaptive array. This specific antenna seems to be suitable for mobile applications and wireless computer network.

18

Chapter 5

The Reactance-Domain Technique

5.1 Principle

Because of the configuration of the ESPAR antenna, only the single-port output can be observed and processed. In conventional adaptive arrays, the correlation matrix is created by measuring the signals in each element of the antenna, so the problem is the same for all kinds of single-port array antennas: how to create a correlation matrix with only one output port ?

By using the reactance-domain technique, the goal is to recreate, with a single-port output array antenna, the spatial diversity of a conventional array antenna[15, 8]. First we consider that the M elements of a conventional array antenna can
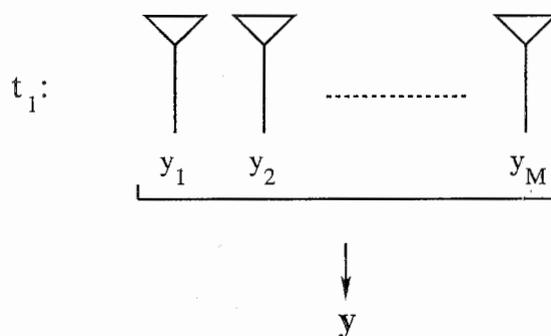
Figure 5.1: Space-domain correlation matrix.

be monitored in the same time and that the correlation matrix can be computed. This principle is depicted in fig. 5.1 with the output vector $\mathbf{y} = [y_1, y_2, \ldots, y_M]$ and the correlation matrix $\mathbf{R}_{yy} = \boldsymbol{E}\left[\mathbf{y}\mathbf{y}^H\right]$. Now we consider only a single element antenna. If the spatial diversity available in a conventional array with M elements, is still claimed, we must change the position of our single element antenna and measure the output M times by sending the same signals each time. This technique is summarized by fig. 5.2. Thus M output values can be obtained as if an M-element antenna was used. In fact a decomposition in time is performed to recreate the spatial diversity of the M-element antenna. Therefore a correlation matrix can be created. The same reasoning can be conducted with the single-port output ESPAR antenna but the reactance-domain approach will be used.

19
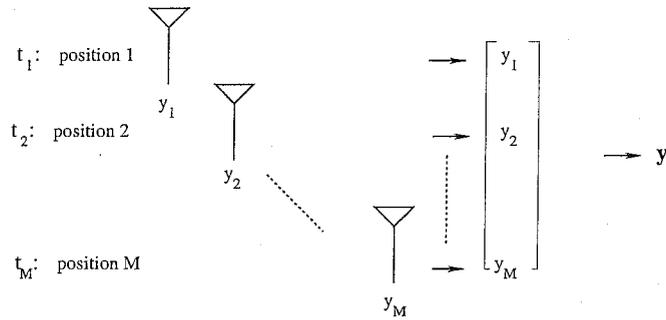
Figure 5.2: Time-domain correlation matrix.
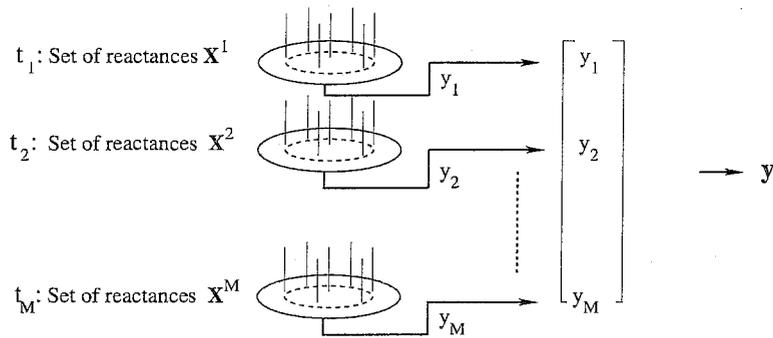
Figure 5.3: Reactance-domain correlation matrix.

The ESPAR antenna has M variable reactances. Firstly, a reactance vector is set and the output is measured. To draw a parallel with the decomposition in time, for each measure of the output the reactance vector will be changed. Consequently the spatial diversity will be conserved and by repeating this process M times, a correlation matrix can be created (see fig. 5.3).

The basic requirements to understand the special feature of the ESPAR antenna were presented. So, we can now tackle the ESPRIT algorithm topic.

Chapter 6

The ESPRIT Algorithm Principle for ESPAR Antennas

The ESPRIT algorithm achieves a significant reduction in computational complexity by imposing a constraint on the structure of the sensor array; namely that the array should possess a displacement invariance [16]. The ESPAR antenna design fits this criterion because of its 7-element regular-hexagonal shape. Therefore, we can divide the array into two isometric subarrays separated by a known displacement vector.

In practice, some parameters are required before applying the ESPRIT algorithm. Especially in the case of the ESPAR antenna, the matrix of the current vector $\mathbf{I}$ is required. $\mathbf{I}$ embodies characteristics of the structure of the antenna, and must be obtained using some calibration techniques [21]. It is defined by:

$$\mathbf{I}^T = [\mathbf{i}_1^T, \mathbf{i}_2^T, \ldots, \mathbf{i}_{M+1}^T]^T, \tag{6.1}$$

where $\mathbf{i}_m^T$ corresponds to the current vector associated to the mth set of reactances. The patterns of the set of reactances used to obtain the vectors $\mathbf{i}_m^T$ (m = 1...M+1) are respectively depicted in fig. 13, 14, 15, 16, 17, 18 and 19.

Secondly, an estimate from the measurements of the correlation matrix, $\mathbf{R}_{yy}$, of the signal is needed, with $\mathbf{y}$ defined as:

$$\mathbf{y} = [y_1, y_2, \ldots, y_{M+1}]^T, \tag{6.2}$$

where $\mathbf{y}$ is obtained using the previously explained reactance-domain technique, y_i stands for the output of the antenna measured when using the ith set of reactances. The correlation matrix $\mathbf{R}_{yy}$ is defined according to:

$$\mathbf{R}_{yy} = E[\mathbf{y}\mathbf{y}^H] = \mathbf{I}_{current}^T \mathbf{A} \mathbf{R}_{uu} \mathbf{A}^H \mathbf{I}_{current}^* + \mathbf{R}_{nn}, \tag{6.3}$$

where $E[.]$ denotes the statistical expectation operator.

Secondly, an estimate for the noise covariance matrix, denoted by $\hat{\mathbf{R}}_{nn}$, is given by the Akaike information criterion (AIC) [22, 23]:

$$\hat{R}_{nn} = \hat{\sigma}^2 \mathbf{I}_{M+1}, \tag{6.4}$$

where $\mathbf{I}_{M+1}$, which denotes the $(M + 1) \times (M + 1)$ identity matrix and $\hat{\sigma}^2$, is given by:

$$\hat{\sigma}^2 = \frac{1}{(M+1) - D} \sum_{i=D+1}^{M+1} \lambda_i, \tag{6.5}$$

where D is the number of signals and λ_i (i=1..M) the eigenvalues of $\mathbf{R}_{yy}$.

ESPRIT is based on a subspace decomposition of the signal covariance matrix. Therefore, it is necessary to get a covariance matrix that will contain the most effective data concerning the measured signal. Thus, the following modified covariance matrix of the signal $\mathbf{R}_{zz}$ is used to get an estimate of the signal subspace $\mathbf{E}_s$:

$$\mathbf{R}_{zz} = (\mathbf{I}_{current}^T)^{-1} (\mathbf{R}_{yy} - \hat{\mathbf{R}}_{nn}) \mathbf{I}_{current}^T, \tag{6.6}$$

The signal subspace $\mathbf{E}_s$ is spanned by the eigenvectors corresponding to the D highest eigenvalues of the eigendecomposition of $\mathbf{R}_{zz}$.

$\mathbf{E}_s$ is split according to:

$$\begin{bmatrix} \mathbf{E}_x \\ \mathbf{E}_y \end{bmatrix} = \mathbf{E}_s \begin{bmatrix} \mathbf{J}_1 \\ \mathbf{J}_2 \end{bmatrix}, \tag{6.7}$$

where $\mathbf{J}_1$ and $\mathbf{J}_2$ are selection matrices that depend on the subarray partition. One important parameter for applying the ESPRIT algorithm to the ESPAR antenna is the choice of the subarray partition. The subarray partition is embodied by the selection matrices $\mathbf{J}_1$ and $\mathbf{J}_2$. These matrices represent the displacement invariance that is possessed by an ESPAR antenna. Three configurations are possible, respectively denoted as sets 1, 2 and 3:
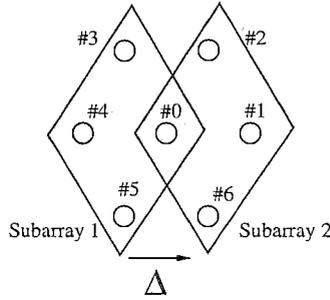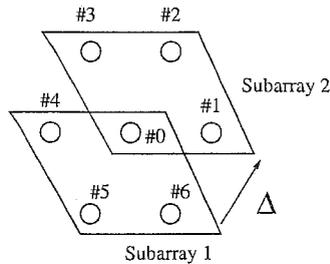

Figure 6.1: Subarray partition, set 1.

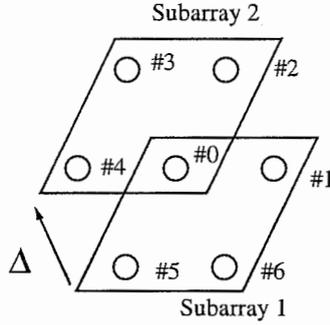Figure 6.2: Subarray partition, set 2.

Figure 6.3: Subarray partition, set 3.

The subarray configuration according to set 1 yields $\mathbf{J}_1$ and $\mathbf{J}_2$ as follows:

$$\mathbf{J}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad \mathbf{J}_2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

where $\mathbf{J}_1$ picks up elements {#0, #3, #4, #5} whereas $\mathbf{J}_2$ gathers elements {#0, #1, #2, #6} in a subarray. The vector $\vec{\Delta}$ which expresses the displacement invariance between the two subarrays is an important parameter on which the DoA estimation depends, especially the $\theta_{reference}$ value in Eq. 6.10.

Consider a conventional array antenna, such as a uniform linear array, on which the ESPRIT algorithm can be applied. In fig. 6.4, a 7-element linear array is depicted with an azimuthal reference axis that bisects the centre of the array and is perpendicular to the line formed by the antenna elements.
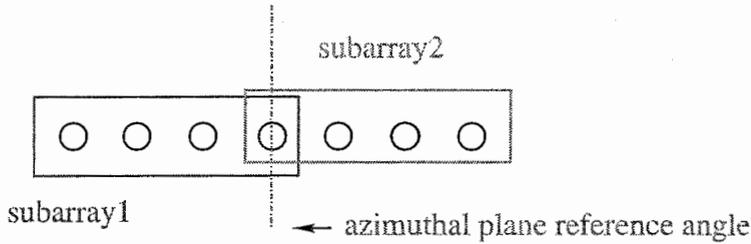
Figure 6.4: Uniform linear array.

However, for an ESPAR antenna, the azimuthal reference axis is not the same. As shown in fig. 4.5, the element #1 is used for this purpose. Furthermore, the $\theta_{reference}$ value depends on the orientation of the displacement vector $\vec{\Delta}$. For example, if the configuration depicted by fig. 6.1 is used, $\vec{\Delta}$ is perpendicular to the azimuthal reference axis of a linear antenna. Hence $\theta_{reference}$ is equal to 90° to balance this effect. Fig. 6.5 summarizes the value of $\theta_{reference}$ according to the orientation of the displacement vector $\vec{\Delta}$. The bold black arrow corresponds to the displacement vector $\vec{\Delta}$ and the configurations associated with the name set i (a) or set i (b) for i=1,2,3 are depicted in table 1.

Table 1 summarizes the values of $\theta_{reference}$ according to the subarray partition. The next step consists of computing the eigendecomposition of $\mathbf{E}_{xy}$ given by :

23

Set 3(a) $\theta_{reference} = -30^{\circ}$

Set 2(a) $\theta_{reference} = +30^{\circ}$

Set 1(b) $\theta_{reference} = -90^{\circ}$

Set 1(a) $\theta_{reference} = +90^{\circ}$

0° azimuthal reference

Set 2(b) $\theta_{reference} = -150^{\circ}$

Set 3(b) $\theta_{reference} = +150^{\circ}$

azimuthal plane reference angle
for a conventional array

Figure 6.5: $\theta_{reference}$ values according to the orientation of the displacement vector $\vec{\Delta}$.

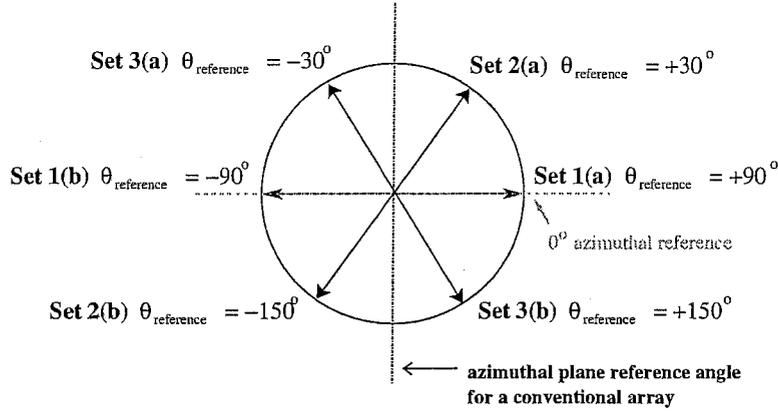| set #	set 1(a)	set 1(b)	set 2(a)	set 2(b)	set 3(a)	set3(b)
Subarray partition						
$\theta_{reference}$	$+90^{\circ}$	-90°	$+30^{\circ}$	-150°	-30°	$+150^{\circ}$

Table 6.1: $\theta_{reference}$ values according to subarray partition.

$$\mathbf{E}_{xy} \stackrel{def}{=} \begin{bmatrix} \mathbf{E}_x^* \\ \mathbf{E}_y^* \end{bmatrix} [\mathbf{E}_x \mathbf{E}_y] = \mathbf{E}\Lambda\mathbf{E}^*, \qquad (6.8)$$

where Λ is the diagonal matrix containing the eigenvalues of $\mathbf{E}_{xy}$.
Here,

$$\mathbf{E} \stackrel{def}{=} \begin{bmatrix} \mathbf{E}_{11} & \mathbf{E}_{12} \\ \mathbf{E}_{21} & \mathbf{E}_{22} \end{bmatrix}, \qquad (6.9)$$

where $\mathbf{E}_{ij}$ is a $D \times D$ submatrix.

Then the eigenvalues of $\Psi = -\mathbf{E}_{12}\mathbf{E}_{22}$ are calculated and denoted by ϕ_k. Finally the DoA estimations θ_k are:

$$\theta_k = \sin^{-1}\left(\frac{c}{\omega_0 \Delta} angle(\phi_k)\right) + \theta_{reference}, \qquad (6.10)$$

where $angle(.)$ provides the argument of ϕ_k, ω_0 is the frequency of the received signal, the constant c is the speed of propagation, and $\Delta = ||\vec{\Delta}|| = R = \frac{\lambda}{4}$ for a 7-element regular-hexagonal shaped ESPAR antenna. In this section the ESPRIT principle was detailed, in the following, simulations will be conducted for performance evaluation.

6.1 Simulation Results

Simulations were performed with these conditions:

- The SNR rate is set at 10 dB

- 5 000 snapshots are used

The bound of the number of sources that can be estimated was discussed in [24], and when the number of subarrays is equal to 2, the upper bound d_{max} is the number of elements per subarray (i.e., m=4). The simulations show that a maximum of 3 signals impinging at the same time on the antenna can be distinguished (see fig. 6.6, 6.7, 6.8). The algorithm should be able to detect up to 4 signals, nevertheless, in the ESPAR antenna case, the overlapping subarray partition makes it such that only 3 elements are considered per subarray.
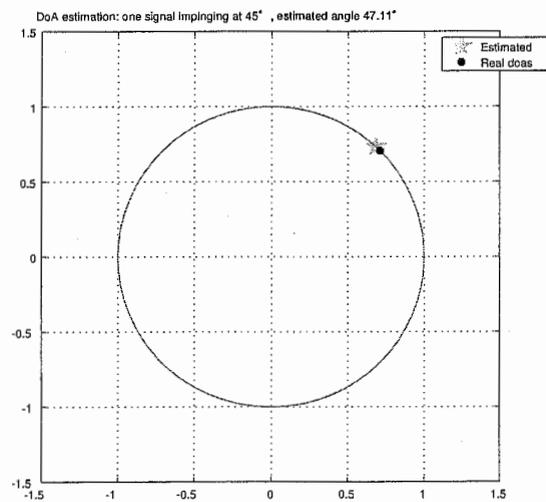

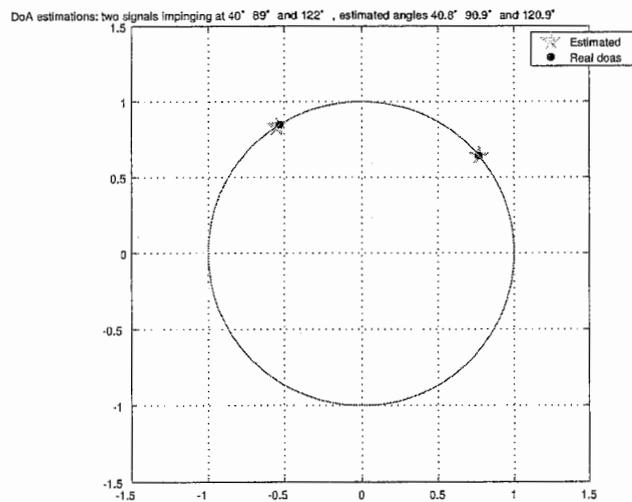Figure 6.6: DoA estimation for one impinging signal.

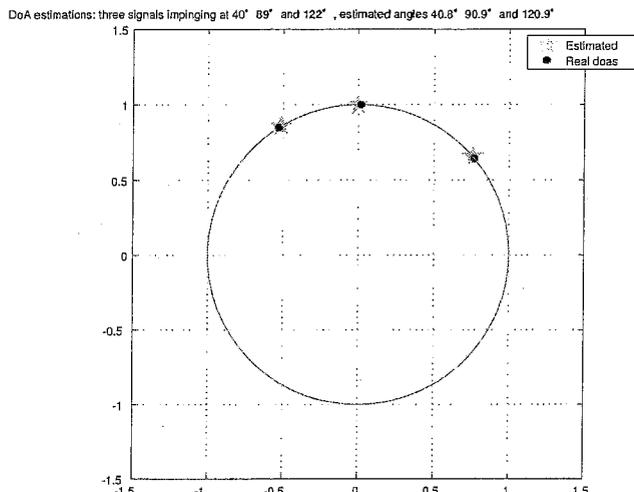Figure 6.7: DoA estimation for two impinging signals.

DoA estimations: three signals impinging at 40° 89° and 122° , estimated angles 40.8° 90.9° and 120.9°

Figure 6.8: DoA estimation for three impinging signals.

6.1.1 Checking $\theta_{reference}$ values

The reference value $\theta_{reference}$ is linked to the set of subarrays chosen. Thus the first set of simulations was conducted in order to verify this fact. The values found by simulation matched the values expected in table 6.1.

6.1.2 Cover sector

This second simulation is conducted in order to determine the cover zone of the ESPAR antenna, namely, on which range of angles DoA estimations are available and reliable.

set #	1(a)	1(b)	2(a)	2(b)	3(a)	3(b)
cover sector	[0° 180°]	[−180° 0°]	[−120° 60°]	[60° 240°]	[−60° 120°]	[−240° −60°]

Table 6.2: Cover sector according to subarray partition.

As expected, the cover sector depends on the subarray partition employed in the algorithm. Table 6.2 summarizes the DoA coverage sector according to subarray partition. Graphs from figures 6.9 and 6.10 provide the range where the ESPRIT algorithm is the most efficient using, respectively, subarray partition set 1(a) and set 1 (b).

The first observations we can make from these graphs is that the quality of the DoA estimation relies on the value of the angle. In fig. 6.9, we notice that for signals arriving almost horizontally to the array (angles around 0° to 30° and around 140° to 180°), the error magnitude is nearly ten times higher. Moreover, we see that the antenna does not possess a 360° DoA coverage area. This is due to the implementation of the ESPRIT algorithm, which makes the ESPAR antenna similar to a linear antenna that can only detect signals in the range of 0° to 180°. In a word, we can say that the accuracy of DoA estimations provided by ESPRIT
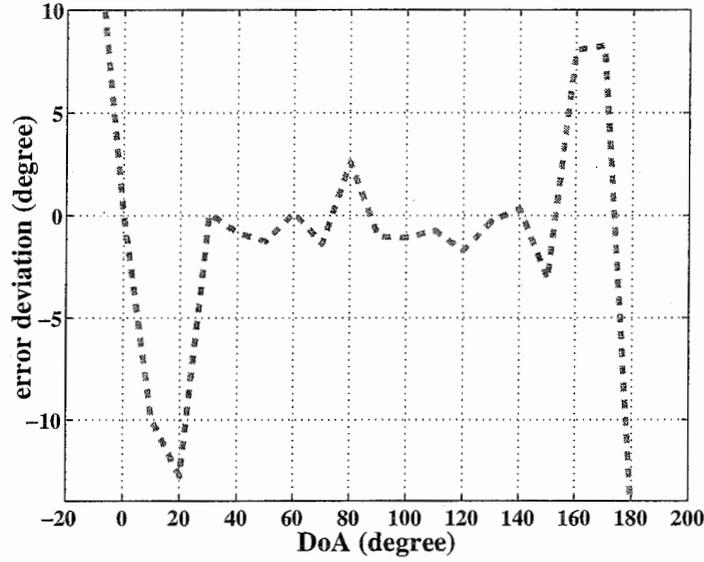
Figure 6.9: Cover sector of the ESPAR antenna using the set 1(a).
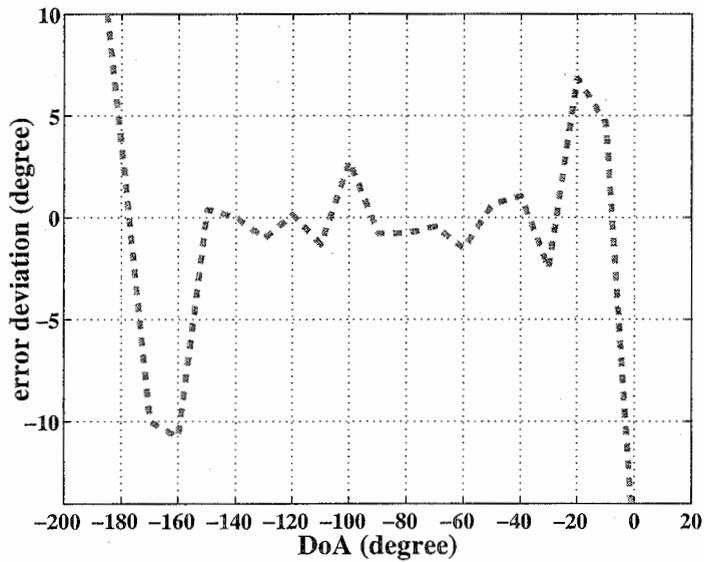
Figure 6.10: Cover sector of the ESPAR antenna using the set 1(b).

algorithm when applied to ESPAR antennas depends on the angle of the impinging signal. So for the next simulation we will reduce our sector zone as described below in table 6.1. Table 6.3 summarizes the efficient work zones for the ESPAR antenna according to the subarray partition.

set #	1(a)	1(b)	2(a)	2(b)	3(a)	3(b)
cover sector	[40° 140°]	[−140° −40°]	[−80° 20°]	[100° 200°]	[−20° 80°]	[−200° −10(

Table 6.3: Efficient cover sector according to subarray division.

6.1.3 Performances

In the third simulation, we investigate the performance of the ESPRIT algorithm. We consider a single signal impinging on the array with a DoA in the range of 20° to 160°. The subarray partition used for the simulation is the set 1(a). The output error magnitude statistics are studied under the assumptions that after 10 000 iterations, the DoA θ of the signal varies from 20° to 160° with a 5° step and that the signal and the additive noise are generated randomly. Our statistical analysis adopts a 3-dimensional cumulative distribution function (CDF) of the error magnitude, taken as $error =| \theta - \hat{\theta} |$, according to the fluctuations of the input signal. Fig. 6.12 depicts the contour-lines of the CDF curve, namely $p\left[| \theta - \hat{\theta} |\leq value_{error}\right]$ for a given DoA θ. The results are expressed in percentage.

Dispatching of errors according to DoA
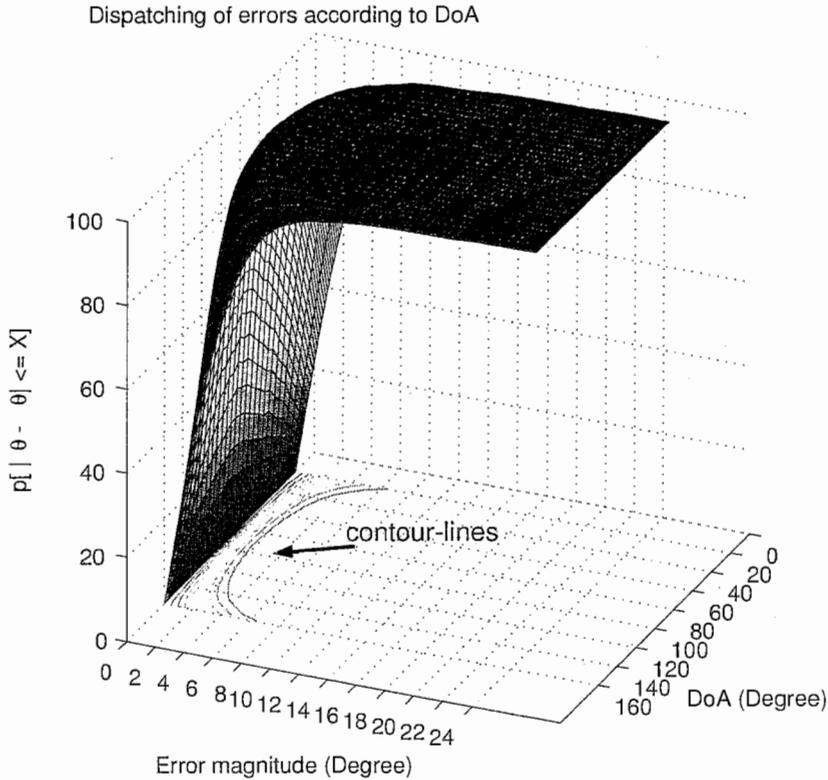
Figure 6.11: CDF curve of the error magnitude $p\left[| \theta - \hat{\theta} |\leq value\right]$, SNR=10 dB.

For example, for a signal arriving on the array with an angle in the range of 50° to 130°, the graph says that there is an 80% of probability of getting an estimation with ±2° accuracy. Furthermore, we can still observe the edge effects mentioned in the previous section. Indeed, for the angles on the edge (smaller than 40° and higher than 140°), the probability of getting a 2 degree precision estimation decreases to 40%.
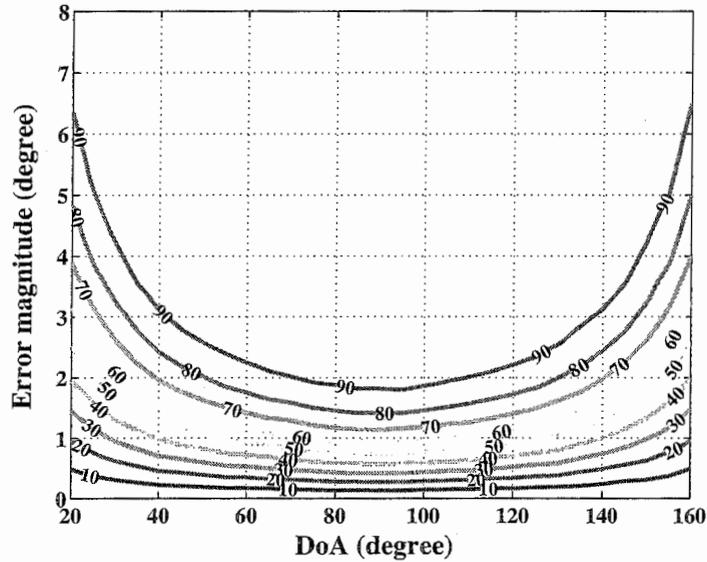
Figure 6.12: Contour-lines of the 3D CDF curve of the error magnitude, SNR=10 dB.

6.1.4 SNR Influence

In the fourth simulation, we want to evaluate how sensitive the ESPRIT algorithm is to noise. Curves 6.13 and 6.14 provide the error repartition, respectively, with an SNR rate of -5 dB and 0 dB. We also have to take into consideration graph 6.12.
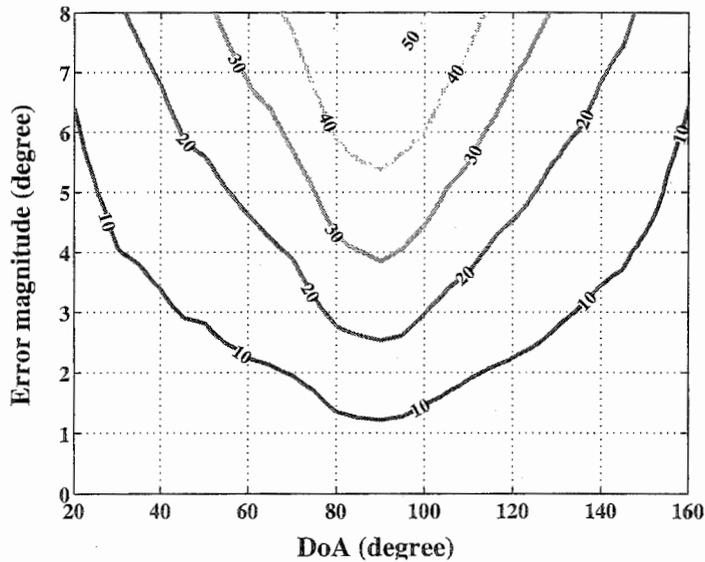
Figure 6.13: Contour-lines of the 3D CDF curve of the error magnitude $p\left[|\theta - \hat{\theta}| \leq value\right]$, SNR=-5 dB.

Consider we have a signal impinging at $60°$ on the antenna and we expect a $2°$ precision. The results can be summarized in table 6.4 .
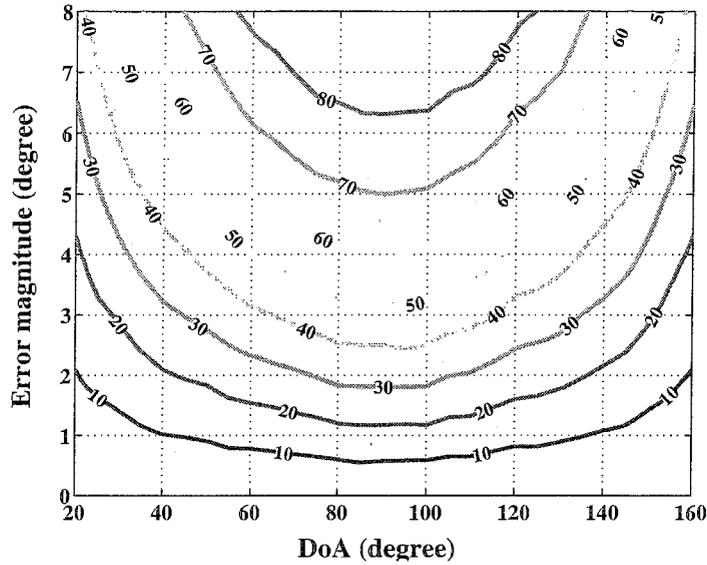
Figure 6.14: Contour-lines of the 3D CDF curve of the error magnitude $p\left[\mid \theta - \hat{\theta} \mid \leq value\right]$, SNR=0 dB.

SNR (dB)	-5	0	10
Probability	$\approx 10\%$	$\approx 25\%$	$>80\%$

Table 6.4: Probabilities of getting a DoA with a 2° precision according to SNR.

Just by quoting this example, we can say than the estimations are quite sensitive to noise. The lower the noise is, the better the estimates are.

6.1.5 Influence of the Number of Samples

In the fifth simulation, the goal is to determine how the number of samples for the signal can change the accuracy of DoA estimations. We consider in fig. 6.15, one signal impinging on the antenna, with a DoA equal to 95°.

The minimum number of samples used for the signal depends on how accurate the estimations must be. For instance, if we want a ±0.5° accuracy, at least 1,000 samples are required.

Influence of the number of samples on a DoA estimation
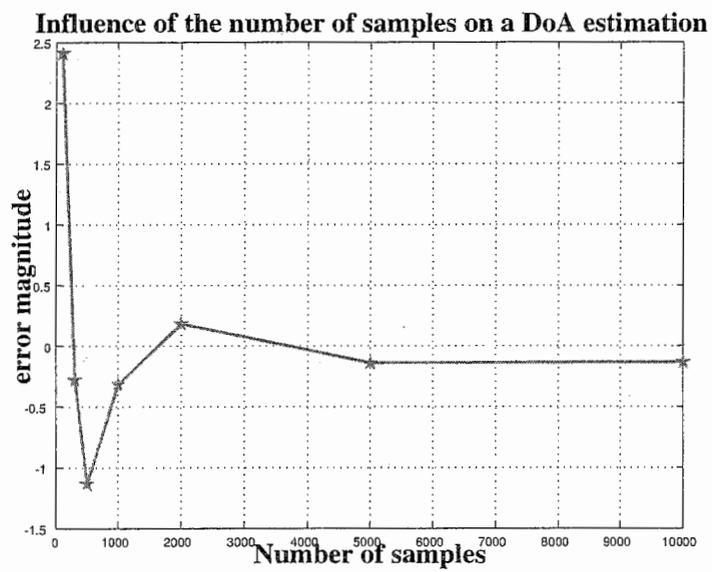
Figure 6.15: Influence of the number of samples on DoA estimation for a signal impinging at 95°.

Chapter 7

Improvement

The application of the ESPRIT algorithm to the ESPAR antenna only yields DoA estimations for angles in half the full azimuthal plane sector. But the ESPAR antenna, with its circular structure, should make possible detection of signals on a 360° area. By using in the same time the three possible combinaison of subarrays within a 7-element ESPAR antenna, a technique was then found to achieve detection of signal on a 360° area.

7.1 The Method

The element arrangement of a 7-element hexagonal-regular shaped ESPAR antenna offers different subarray partitions for the computation of the algorithm. As said previously (see table 6.1.2), each subarray partition provides a different cover sector and fig. 7.1 summarizes all the possibilities.
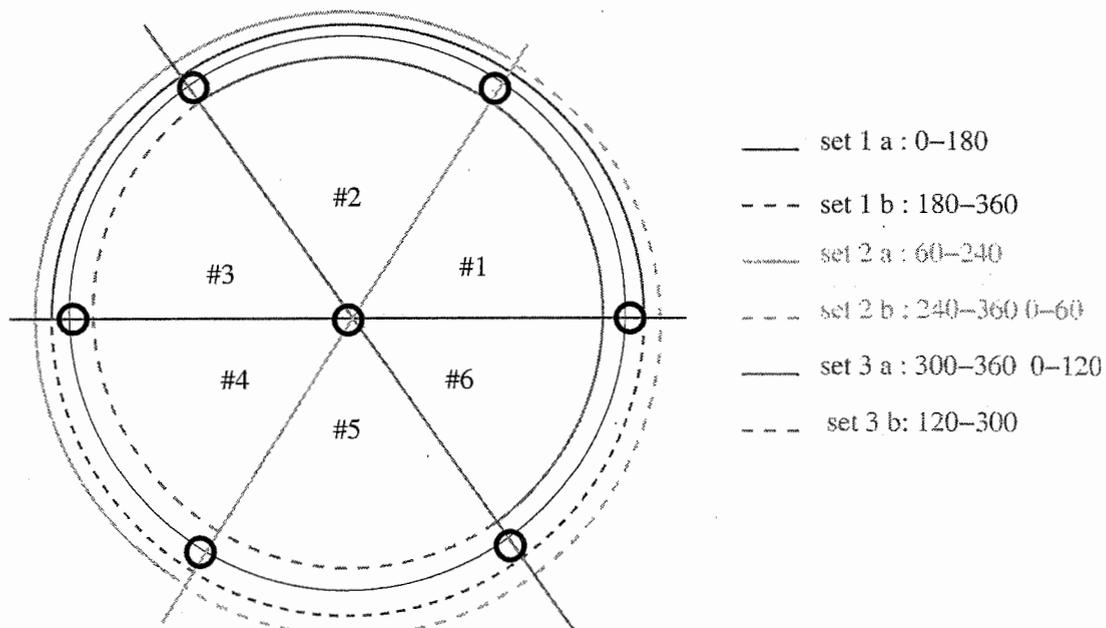
Figure 7.1: Cover sectors according to subarray partition.

If we look carefully to this schematic, we see that each 60°-angular sector is covered at least three times for the DoA estimation. From this observation the following reasoning was drawn. DoA estimation can be performed three times using a different subarray partition set #i a for $i = 1..3$. This will yield three estimates denoted: doa1a,doa2a, and doa3a. But in fact, we can obtain six DoA estimations according to these formulas:

- doa1b = - doa1a

- doa2b = 120 - doa2a

- doa3b = - 120 - doa3a

These relations stems from the symmetries within the 7-element ESPAR antenna. But among these six results, some of them will be very closed to the real DoA angle whereas others will be wrong either because the angle is a broad-side angle or not in the cover sector.

So, the next step consists of identifying the 60°-angular sector that contains the more estimates. The decision is simply based on a numerical criterion. Therefore, the matching table 7.1 provides the DoA estimates to use according to the 60°-angular sector.

Sector	Set to use	DoA estimates
#1	set 3 b	doa3b
#2	set 1 a	doa1a
#3	set 2 a	doa2a
#4	set 3 a	doa3a
#5	set 1 b	doa1b
#6	set 2 b	doa2b

Table 7.1: Decisions

Indeed, as each cover sector is composed of three angular sectors, the middle angular sector will be taken as its representative. For example, set 3 a gathers angular sectors # 6, #1 and # 2 so its representative is the sector # 1. Therefore, for a signal impinging in the sector #1, the best subarray partition to use is set 3 a. The fig. 7.2 represents the six DoA estimations for a single signal impinging at 48°.

Estimates	doa1a	doa1b	doa2a	doa2b	doa3a	doa3b
	44.09	-44.09	-	-	-170.88	50.88

Table 7.2: DoA estimates for a signal impinging at 48°

doa1b and doa2a are not available as they are not estimations. Indeed, in fig. 7.2, they appear out of the unity circle but they are closed to the real DoA angle. The algorithm said that the estimation provided by the subarray partition set 3 b should be taken as the good estimation.
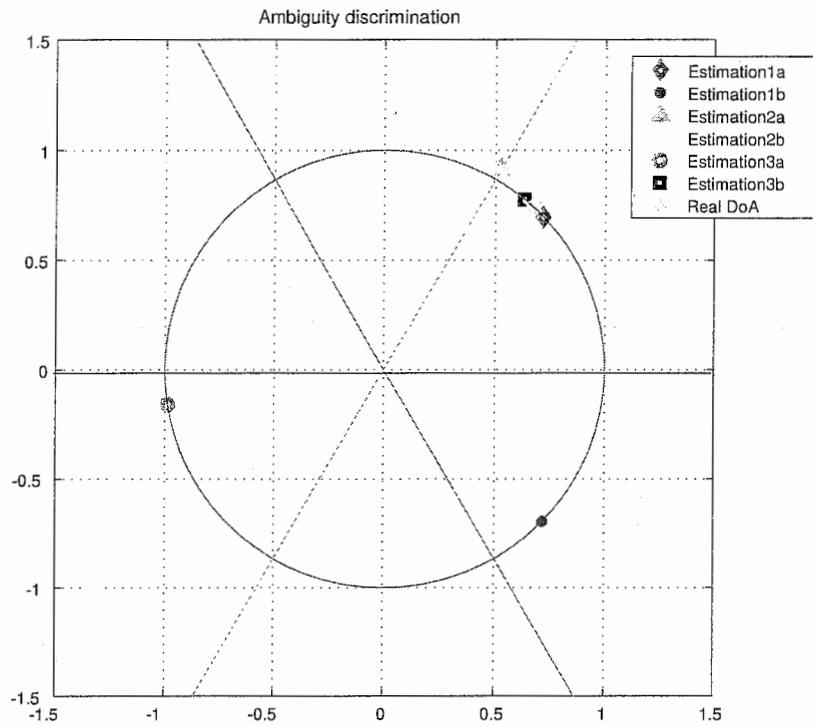
Figure 7.2: DoA estimations for a signal impinging at 48°.
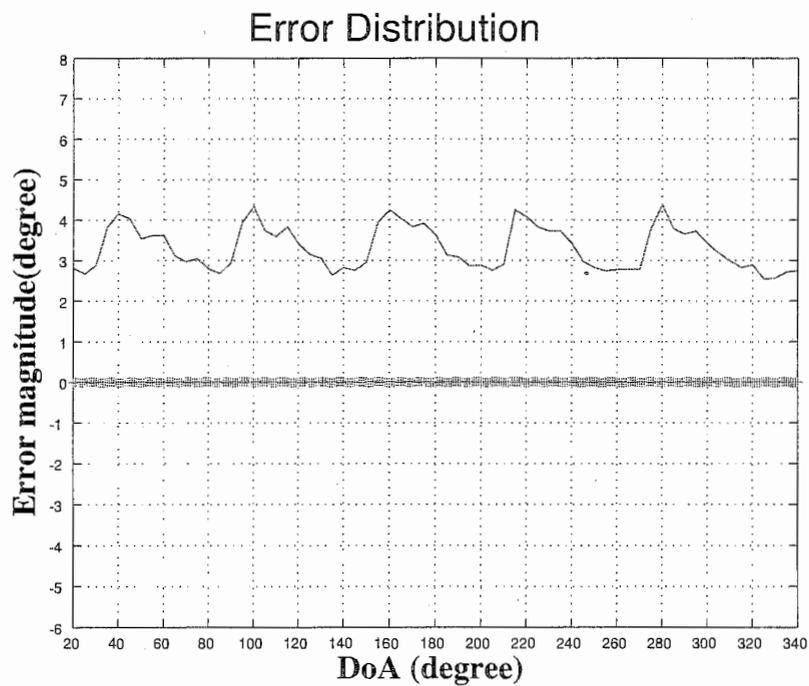
Figure 7.3: Error distribution.

The next simulation was conducted for performance evaluation of this technique

on a 360° area.

The first observation we can draw is that the error magnitude is quite low and rather constant for all the angles. Moreover, the fact that the error mean is not closed to zero can be justified. By the fact that the ESPRIT algorithm provides biased estimations the error mean can not be null.

Chapter 8

Conclusion

WACNet concept is supposed to support network communications at anytime and anywhere. One of the key components developed to achieve the WACNet is the ESPAR antenna implemented in user terminals. But the ESPAR antenna presents a special feature: it is a single-port output antenna. So, algorithms designed for conventional antennas cannot be directly applied. In this report, we proposed an application of the ESPRIT algorithm to the ESPAR antenna that can improve the performance of the WACNet.

This report shows that it was possible to apply the conventional array antenna oriented ESPRIT algorithm, with a non-conventional seven-element ESPAR antenna. The ESPRIT algorithm requires displacement invariance within the array structure, which is a strong constraint on the geometry of the antenna. DoA estimations can be achieved thanks to the 7-element hexagonal-regular shaped ESPAR antenna. Indeed, by exploiting the underlying translational invariance designed into the array and by combining the ESPRIT algorithm and the reactance domain approach, DoA estimations can be obtained. Although the ESPAR antenna can detect any signal within a 360° cover area, the application of this modified ESPRIT algorithm reduces the cover area to a sector of half of the full azimuthal plane sector. Furthermore, simulations show that the performance of the algorithm is linked to the value of the angle impinging on the array as we could observe some edge effects for angles around the limit values of the cover sector. We could expect these results as we have adapted a technique formerly designed for a linear antenna to an ESPAR antenna. But by using at the same time, the three possible combinaison of subarray partition, DoA estimation could be achieved on full the azimuthal plane. Simulations were performed for non-coherent signals. But to cope with the coherent source estimation problem, the proposed technique is not sufficiently efficient. We are thinking to implement a technique based on the CUBA-ESPRIT algorithm [17], which is used for circular uniform multibeam antenna designs. The ESPAR antenna could be considered such an antenna.

Acknowledgements

Bibliography

[1] Hamid Krim and Mats Viberg. Two decades of array signal processing research: The parametric approach. *IEEE Signal Processing Magazine*, 13(4):67–94, July 1996.

[2] Takashi Ohira. Microwave signal processing and devices for adaptive beamforming. In *IEEE Antennas and Propagation Society International Symposium*, volume 2, pages 583–586, Salt Lake City, 16-21 July 2000.

[3] Takashi Ohira and Koichi Gyoda. Electronically steerable passive array radiator antennas for low-cost analog adaptive beamforming. In *IEEE International Conference on Phased Array Systems and Technology*, pages 101–104, Dana Point, CA, USA, May 21st-25th 2000.

[4] Masaya Hashiguchi, Jun Cheng, Kyouichi Iigusa, Eddy Taillefer, Akifumi Hirata, and Takashi Ohira. Basic experiments for adaptive beamforming of the ESPAR antenna. *IEICE Proceedings of the 2001 communication society conference*, B-1-65:71–76, Sept. 18th 2001. in Japanese.

[5] Koichi Gyoda, Yuichiro Ohno, Youito Kado, and Takashi Ohira. Wacnet: Wireless ad-hoc community network. In *IEEE International Conference on Industrial Electronics, Control and Instrumentation*, pages 1153–1158, Nagoya Japan, October 22nd-28th 2000. in Japanese.

[6] Takashi Ohira and Koichi Gyoda. Hand-held microwave direction-of-arrival finder based on varactor-tuned analog aerial beamforming. In *IEEE Asia-Pacific Microwave Conference*, volume 2, pages 585–588, Taïpeï, Taiwan, Dec. 2001.

[7] Tomoshige Furuhi, Masaya Hashiguchi, Takashi Ohira, Mineo Asada, and Toshimi Okada. Snowfield test of wristwatch-type microwave beacons and hand-held direction-of-arrival finders. *IEICE Transaction Journal*, J86-B(2):219–225, Feb. 2002. in Japanese.

[8] Cyril Plapous, Jun Cheng, Eddy Taillefer, Akifumi Hirata, and Takashi Ohira. Reactance-domain MUSIC algorithm for ESPAR antennas. *IEICE Technical Report, RCS2002-147*, 102(282):1–8, Aug. 2002.

[9] Eddy Taillefer, Cyril Plapous, Jun Cheng, Kyouichi Iigusa, and Takashi Ohira. Reactance-domain MUSIC for ESPAR antennas (experiment). In *IEEE Wire-*

less Communications and Networking Conference, volume 1, pages 98–102, New Orleans, Louisiana, March 16nd-20th 2003.

[10] Eddy Taillefer, Akifumi Hirata, and Takashi Ohira. Doa estimation based on power pattern cross correlation with an ESPAR antenna. In *IEICE General Conference*, volume B, page 67, Sendai, Mars 2003. in Japanese.

[11] Eddy Taillefer, Akifumi Hirata, and Takashi Ohira. High-accuracy direction-of-arrival estimation using an ESPAR antenna and cross correlation between received signal and radiation pattern. In *European Microwave Week 2003, on Wireless Technology*, Munich, October 2003.

[12] Mathieu Lenoble, Thomas Hunziker, and Takashi Ohira. Direction-of-arrival and time-delay estimation with an ESPAR antenna employing SAGE algorithm. In *IEICE Antenna and Propagation Technical Meeting*, pages –, Sendai, August 2003.

[13] Emmanuelle Chu, Eddy Taillefer, and Takashi Ohira. Direction-of-arrival estimation with a seven-element regular-hexagonal shaped ESPAR antenna employing ESPRIT algorithm. In *IEICE Antenna and Propagation Technical Meeting*, pages –, Sendai, August 2003.

[14] Emmanuelle Chu, Eddy Taillefer, and Takashi Ohira. Direction-of-arrival estimation with a seven-element regular-hexagonal shaped ESPAR antenna employing ESPRIT algorithm. In *IEICE General Conference*, pages –, Niigata, September 2003.

[15] Akifumi Hirata, Jun Cheng, Masaya Hashiguchi, and Takashi Ohira. Reactance-domain signal processing for ESPAR antennas - spatial correlation and correlation matrix -. *IEICE Technical report RCS2002-148*, 102(282):9–14, Aug. 2002. in Japanese.

[16] Richard Roy and Thomas Kailath. ESPRIT- estimation of signal parameters via rotational invariance techniques. In *IEEE Transactions on Acoustics, Speech, and Signal Processing*, volume 37, pages 984–995, July 1989.

[17] Andreas Richter and Reiner S. Thomä. Angle estimation with circular uniform beam arrays using the CUBA-ESPRIT. In *Millennium Conference on Antenna and Propagation AP2000*, pages 9–14, Davos, Switzerland, April 2000.

[18] Jun Cheng, Yukihiro Kamiya, and Takashi Ohira. Adaptive beamforming of ESPAR antenna based on steepest gradient algorithm. *IEICE Transaction Communication*, E84-B(7):1790–1800, July 2001.

[19] Takashi Ohira, Kyoichi Gyoda, A. Akiyama, and Kehu Yang. Equivalent weight vector and array factor formulation for ESPAR antennas. *IEICE Technical report AP2000-44, SAT2000-41, NW2000-41*, July 2000. in Japanese.

[20] Kyouichi Iigusa, Masaya Hashiguchi, Akifumi Hirata, and Takashi Ohira. ESPAR antenna parameters fitting based on measured data. *IEICE Technical Report, AP2001-104, RCS2001-143,*, pages 93–100, October 2001. in Japanese.

[21] C.M.S. See. Sensor array calibration in presence of mutual coupling and unknown sensor gains and phases. In *Electronics Letters*, volume 30, pages 373–374, March 3rd 1994.

[22] Mati Wax and Thomas Kailath. Detection of signals by information theoretic criteria. In *IEEE Transactions on Acoustics, Speech, and Signal Processing*, volume ASSP-33, pages 387–392, April 1985.

[23] Don H. Johnson and Dan E. Dudgeon. *Array signal Processing: Concepts and Techniques*. P.T.R Prentice-Hall, Inc., 1 edition, 1993. ISBN 0-413-048513-6.

[24] A. Lee Swindlehurst, Björn Ottersten, Richard Roy, and Thomas Kailath. Multiple invariance esprit. In *IEEE Transactions on Signal Processing*, volume 40, pages 867–881, April 1992.

Appendix A : MatLab source codes

rea2curb.m

> **program name**: rea2curb.m
> **comments**: Transformation of reactances to current with admittance matrix

```
function cur=reactances2currentbis(zs,rea)

% Given reactance vector, calulate the current vector

% set the admittances
% experimental parameters

% y00 =   0.0008616-j*0.0120795;
% y10 = -0.0006963+j*0.0036462;
% y11 =   0.0044216-j*0.0071600;
% y21 =   0.0009721+j*0.0047851;
% y31 = -0.0005376-j*0.0011297;
% y41 =   0.0001701-j*0.0002950;


% iigusa kara 1

y00 =   0.00023344-j*0.0066918;
y10 = -0.0001816+j*0.00242469;
y11 =   0.00300676-j*0.0044176;
y21 =   0.00097427+j*0.00299986;
y31 = -0.0003066-j*0.0003067;
y41 =   0.0000973-j*0.0001353;

Y = [y00 y10 y10 y10 y10 y10 y10;
     y10 y11 y21 y31 y41 y31 y21;
     y10 y21 y11 y21 y31 y41 y31;
     y10 y31 y21 y11 y21 y31 y41;
     y10 y41 y31 y21 y11 y21 y31;
     y10 y31 y41 y31 y21 y11 y21;
     y10 y21 y31 y41 y31 y21 y11];

Z = inv(Y);
```

```
% iigusa kara 2
Z(1,1) = 145*exp(j*2*pi*275/360);

% set the admittance vector
u0 = [1 0 0 0 0 0 0].';

% calulate the current vector
cur=2*zs*inv(Z+diag([zs;j*rea]))*u0;
```

digvolt2imp.m

program name: digvolt2imp.m
comments: toto

```
function z = digvolt2imp (dv)

z = 2e-7 * dv.^2 -0.0005*dv + 0.393 -j*(49.22 + 0.02067*dv);
%z = -j*(49.21 + 0.021*dv);
```

signalgeneratorv1.m

program name: signalgeneratorv1.m
comments: toto

```
function [sig] = signalgeneratorv1(nval,type,param1,param2)

%   signalgeneratorv1(N,TYPE,K,L)
%
%   Genere des constellations de points aleatoires uniformement repartis
%   N     : taille du vecteur
%   K     : parametre K
%   L     : parametre L
%   TYPE :
%   'QAM'    -> Quadrature amplitude modulation   +-(2*K+1)+-j(2*L+1)
%   'PSK'    -> Phase Shift Keying                exp(2*pi*j*[1:K]/K)
%   'BPSK'   -> Binary Phase Shift Keying         +-1
%   'QPSK'   -> Quadrature Phase Shift Keying     +-1+-j
%   'AMFM'   -> Amplitude-Phase modulation        [1:K]*exp(2*pi*j*[1:L]/L)

tmp = [];
j = sqrt(-1);
switch (type)
case 'QAM'
    for k=0:param1-1
        for l=0:param2-1
            tmp = [tmp; k*2+1 + (l*2+1)*j; -(k*2+1) - (2*l+1)*j; k*2+1 - (l*2+1)*j; -(k*2+1)
        end
    end
    sig = tmp(randint(nval,1,length(tmp))+1);
case 'QPSK'
    sig = 1-2*randint(nval,1,2) + (1-2*randint(nval,1,2))*j;
case 'BPSK'
    sig = 1-2*randint(nval,1,2);
case 'PSK'
    sig = exp(2*pi*j*(randint(nval,1,param1)+1)/param1);
case 'AMPM'
    for k=1:param1
        tmp = [tmp; k*exp(2*pi*j*(randint(nval,1,param2)+1)/param2)];
    end
    sig = tmp(randint(nval,1,length(tmp))+1);
end
```

Appendix B: ESPRIT using subarray partitions 1.a and 1.b

eespritSet11.m

> **program name**: eespritSet11.m
> **comments**: Esprit algorithm used with subarray partition 1.a

```
%% Zone de couverture set 1 [0 180]

clf;
clear all;
P = 5000;
SNR = 10;
x0 = 50;
angled = -20:10:200;

Niterations = 10000;
erreur1 = zeros(1,length(angled));
estimation1 = zeros(1,Niterations);;




for idAngle=1:1:length(angled)
doas = [angled(idAngle)];




%psi = diag(exp(sqrt(-1)*(pi/2)*sin(doas)));

el1 = [1 0 0 0].';
el2 = [0 1 0 0].';
el3 = [0 0 1 0].';
el4 = [0 0 0 1].';
zer = [0 0 0 0].';
```

```
% set 1.1
J1 = [el3, el1, el2, zer, zer, zer, el4];
J2 = [el1, zer, zer, el2, el3, el4, zer];


react1 = [ 0 0 0 0 0 0 ;
    2047 2047 -2048 -2048 -2048 2047;
    2047 2047 2047 -2048 -2048 -2048;
    -2048 2047 2047 2047 -2048 -2048;
    -2048 -2048 2047 2047 2047 -2048;
    -2048 -2048 -2048 2047 2047 2047;
    2047 -2048 -2048 -2048 2047 2047
];




[M tmp] = size(react1);
currents1 = zeros(M-1,7);
for index= 1:M
    imp = [x0; digvolt2imp(react1(index,:)).'];
     current1 = rea2curb(imp(1),imp(2:7));
    currents1(index,:) = current1.';
end

icur1 = inv(currents1);


rad2deg = 180/pi;
deg2rad = pi/180;

nsig = length(doas);

symbols = zeros(nsig,P);
for index = 1:nsig
    symbols(index,:) = 1-2*randint(1,P,2); % BPSK
end


A = [                   ones(1,nsig);
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*doas));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-60)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-120)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-180)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-240)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-300)))
];
```

```
psym = diag((symbols*symbols')/P);
noise = randn(M,P) + sqrt(-1)*randn(M,P);

%#####################################################################

% premiere boucle


%#####################################################################


% Tous les signaux ont le meme SNR
sig = diag(sqrt((2*(10.^(SNR/10.0)))./psym)) * symbols;



signal1 = ((currents1*A)*sig);
x1 = signal1 + noise;

Ryy1 = x1*x1';
[Uy1,Sy1,Vy1]=svd(Ryy1);
sigma1 =0;
for indice = nsig+1:M
sigma1 = 1/(M-nsig)*(Sy1(indice,indice))+sigma1;
end

Rxx1 = icur1*(Ryy1-sigma1*eye(M))*currents1;

pa=nsig;


for iteration =1:1:Niterations
[Uxx1, Sxx1, Vxx1] = svd(Rxx1);



Es1 = Uxx1(:,1:pa);



Ux1 =J1*Es1;
Uy1 =J2*Es1;
Uz1 = [Ux1'*Ux1 Ux1'*Uy1; Uy1'*Ux1 Uy1'*Uy1];
[U1,S1,V1]=svd(Uz1);
Vn1= V1(:,pa+1:2*pa);
```

```
Phi1 = -Vn1(1:pa,:)*inv(Vn1(pa+1:2*pa,:));
[Up1,Sp1,Vp1] = svd(Phi1);
phi1 = eig(Phi1);


estimation1(iteration) = asin((2/pi)*angle(phi1))*rad2deg +90 ;


end


erreur1(idAngle) = mean(estimation1)-doas;


end
save   couverture1 angled erreur1
```

traceCover11.m

```
load couverture1;
plot(angled,erreur1,'--r','linewidth',4)
hold on;
xlabel('DoA (degree)','FontWeight','Bold','FontSize',20,'FontName','Times');
ylabel('Error deviation (degree)','FontWeight','Bold','FontSize',20,'FontName','Times');
x1=0;
x2=180;
y = -10:10;
plot(x1,y)
plot(x2,y)
axis([-20 200 -14 10]);
grid
```

eespritSet12.m

```
%% Zone de couverture set 1 [-180 0]

clf;
clear all;
P = 5000;
SNR = 10;
x0 = 50;
angled = -200:10:20;

Niterations = 10000;
erreur1 = zeros(1,length(angled));
estimation1 = zeros(1,Niterations);;



for idAngle=1:1:length(angled)
doas = [angled(idAngle)];




%psi = diag(exp(sqrt(-1)*(pi/2)*sin(doas)));

el1 = [1 0 0 0].';
el2 = [0 1 0 0].';
el3 = [0 0 1 0].';
el4 = [0 0 0 1].';
zer = [0 0 0 0].';

%set 1.2
J2 = [el3, el1, el2, zer, zer, zer, el4];
J1 = [el1, zer, zer, el2, el3, el4, zer];


react1 = [ 0 0 0 0 0 0 ;
    2047 2047 -2048 -2048 -2048 2047;
    2047 2047 2047 -2048 -2048 -2048;
    -2048 2047 2047 2047 -2048 -2048;
    -2048 -2048 2047 2047 2047 -2048;
    -2048 -2048 -2048 2047 2047 2047;
    2047 -2048 -2048 -2048 2047 2047
];
```

```
[M tmp] = size(react1);
currents1 = zeros(M-1,7);
for index= 1:M
    imp = [x0; digvolt2imp(react1(index,:)).'];
     current1 = rea2curb(imp(1),imp(2:7));
    currents1(index,:) = current1.';
end

icur1 = inv(currents1);


rad2deg = 180/pi;
deg2rad = pi/180;

nsig = length(doas);

symbols = zeros(nsig,P);
for index = 1:nsig
    symbols(index,:) = 1-2*randint(1,P,2); % BPSK
end



A = [                   ones(1,nsig);
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*doas));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-60)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-120)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-180)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-240)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-300)))
];

psym = diag((symbols*symbols')/P);
noise = randn(M,P) + sqrt(-1)*randn(M,P);

%###############################################################################

% premiere boucle


%###############################################################################
```

```
% Tous les signaux ont le meme SNR
sig = diag(sqrt((2*(10.^(SNR/10.0)))./psym)) * symbols;




signal1 = ((currents1*A)*sig);
x1 = signal1 + noise;

Ryy1 = x1*x1';
[Uy1,Sy1,Vy1]=svd(Ryy1);
sigma1 =0;
for indice = nsig+1:M
sigma1 = 1/(M-nsig)*(Sy1(indice,indice))+sigma1;
end

Rxx1 = icur1*(Ryy1-sigma1*eye(M))*currents1;

pa=nsig;


for iteration =1:1:Niterations
[Uxx1, Sxx1, Vxx1] = svd(Rxx1);



Es1 = Uxx1(:,1:pa);




Ux1 =J1*Es1;
Uy1 =J2*Es1;
Uz1 = [Ux1'*Ux1 Ux1'*Uy1; Uy1'*Ux1 Uy1'*Uy1];
[U1,S1,V1]=svd(Uz1);
Vn1= V1(:,pa+1:2*pa);
Phi1 = -Vn1(1:pa,:)*inv(Vn1(pa+1:2*pa,:));
[Up1,Sp1,Vp1] = svd(Phi1);
phi1 = eig(Phi1);



estimation1(iteration) = asin((2/pi)*angle(phi1))*rad2deg -90 ;


end
```

```
erreur1(idAngle) = mean(estimation1)-doas;


end

save  couverture12 angled erreur1
```

traceCover12.m

program name: traceCover12.m
comments: Draw the cover sector for subarray partition 1.b

```
load couverture12;
plot(angled,erreur1,'--r','linewidth',4)
hold on;
xlabel('DoA (degree)','FontWeight','Bold','FontSize',20,'FontName','Times');
ylabel('Error deviation (degree)','FontWeight','Bold','FontSize',20,'FontName','Times');
x1=0;
x2=-180;
y = -10:10;
plot(x1,y)
plot(x2,y)
axis([-200 20 -14 10]);

grid
```

eespritSample.m

program name: eespritSample.m
comments: Evaluation of the influence of the number of samples

```
%%%% pour tester l influence du nombre d echantillons

%%%% SNR fixe


clf;
clear all;
% on construit le signal avec 10000 echantillons et on utilisera une partie
% pour garder le meme signal et bruit)
P = 10000;
echantillon = [ 100 300 500 1000 2000 5000 10000];
SNR = 20;
x0 = 50;
angled = [40 65 95 120];
Niterations = 10000;
matrice = zeros(length(echantillon),length(angled));

%% debut de boucle pour les angles

for idAngle=1:1:length(angled)
doas = [angled(idAngle)];
%doas = [10];
estimation = zeros(1,Niterations);

erreur = zeros(1,length(echantillon));


%psi = diag(exp(sqrt(-1)*(pi/2)*sin(doas)));

el1 = [1 0 0 0].';
el2 = [0 1 0 0].';
el3 = [0 0 1 0].';
el4 = [0 0 0 1].';
zer = [0 0 0 0].';

J1 = [el3, el1, el2, zer, zer, zer, el4];
J2 = [el1, zer, zer, el2, el3, el4, zer];

% J2 = [el1, zer, zer, zer, el2, el3, el4];
% J1 = [el3, el4, el1, el2, zer, zer, zer];

react = [ 0 0 0 0 0 0 ;
    2047 2047 -2048 -2048 -2048 2047;
```

```
    2047 2047 2047 -2048 -2048 -2048;
    -2048 2047 2047 2047 -2048 -2048;
    -2048 -2048 2047 2047 2047 -2048;
    -2048 -2048 -2048 2047 2047 2047;
    2047 -2048 -2048 -2048 2047 2047
];


% react = [ 0 0 0 0 0 0 ;
%       1000 200 -400 -542 -1988 680;
%       2047 2047 2047 -2048 -2048 -2048;
%       -2048 2047 2047 2047 -2048 -2048;
%       -2048 -2048 2047 2047 2047 -2048;
%       -28 -204 -8 207 47 7;
%       2047 -2048 -2048 -2048 2047 2047
% ];

[M tmp] = size(react);
currents = zeros(M-1,7);
for index= 1:M
    imp = [x0; digvolt2imp(react(index,:)).'];
     current = rea2curb(imp(1),imp(2:7));
    currents(index,:) = current.';
end

icur = inv(currents);

rad2deg = 180/pi;
deg2rad = pi/180;

nsig = length(doas);

symbols = zeros(nsig,P);
for index = 1:nsig
    symbols(index,:) = 1-2*randint(1,P,2); % BPSK
end



A = [                 ones(1,nsig);
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*doas));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-60)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-120)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-180)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-240)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-300)))
];
```

```
noise = randn(M,P) + sqrt(-1)*randn(M,P);

  %%  debut de boucle pour le nombre d echantillons

for idEch =1:length(echantillon)
symbolsEch = symbols(1:echantillon(idEch));
psym = diag((symbolsEch*symbolsEch')/echantillon(idEch));

noiseEch =noise(1:7,1:echantillon(idEch));


% Tous les signaux ont le meme SNR
sig = diag(sqrt((2*(10.^(SNR/10.0)))./psym)) * symbolsEch;



signal = ((currents*A)*sig);
x = signal + noiseEch;

Ryy = x*x';
[Uy,Sy,Vy]=svd(Ryy);
sigma =0;
for indice = nsig+1:M
sigma = 1/(M-nsig)*(Sy(indice,indice))+sigma;
end

Rxx = icur*(Ryy-sigma*eye(M))*currents;



pa=nsig;

%%% debut de boucle pour la moyenne
for iteration =1:Niterations
[Uxx, Sxx, Vxx] = svd(Rxx);
sv = diag(Sxx);


Es = Uxx(:,1:pa);
En = Uxx(:,pa+1:M);


Ux =J1*Es;
Uy =J2*Es;
Uz = [Ux'*Ux Ux'*Uy; Uy'*Ux Uy'*Uy];
[U,S,V]=svd(Uz);
```

61

```matlab
Vn= V(:,pa+1:2*pa);
Phi = -Vn(1:pa,:)*inv(Vn(pa+1:2*pa,:));
[Up,Sp,Vp] = svd(Phi);
phi = eig(Phi);



estimation(iteration) = asin((2/pi)*angle(phi))*rad2deg +90 ;
end
%%% fin de boucle pour la moyenne
erreur(idEch) = mean(estimation) - doas;

end
% plot(echantillon,erreur);
% hold on;

%%  fin de boucle pour le nombre d echantillons
matrice(:,idAngle) = erreur.';

end
%% debut de boucle pour les angles

%%%---- 1 erreur en fonction du SNR

figure(1);
plot(echantillon,matrice(:,1),'linewidth',4)
xlabel('Number of samples','FontWeight','Bold','FontSize',12,'FontName','Times');
ylabel('error magnitude','FontWeight','Bold','FontSize',12,'FontName','Times');

 grid
 h = get(gcf,'CurrentAxes');
 set(h,'FontWeight','Bold','FontSize',16,'FontName','Times','LineWidth',1);
 set(gca,'xtick',0:2000:10000);
 set(gca,'xtick',0:1000:10000);
figure(2);
plot(echantillon,matrice(:,2),'+-g','linewidth',4)
xlabel('Number of samples','FontWeight','Bold','FontSize',12,'FontName','Times');
ylabel('error magnitude','FontWeight','Bold','FontSize',12,'FontName','Times');

 grid
 h = get(gcf,'CurrentAxes');
 set(h,'FontWeight','Bold','FontSize',16,'FontName','Times','LineWidth',1);
 set(gca,'xtick',0:2000:10000);
 set(gca,'xtick',0:1000:10000);
figure(3);
plot(echantillon,matrice(:,3),'--r','linewidth',4)
xlabel('Number of samples','FontWeight','Bold','FontSize',20,'FontName','Times');
ylabel('error magnitude','FontWeight','Bold','FontSize',20,'FontName','Times');
```

```
grid
h = get(gcf,'CurrentAxes');
set(h,'FontWeight','Bold','FontSize',16,'FontName','Times','LineWidth',1);
set(gca,'xtick',0:2000:10000);
set(gca,'xtick',0:1000:10000);
figure(4);
plot(echantillon,matrice(:,4),'.-k','linewidth',4)
%plot(echantillon,matrice(:,5),'--b')
%plot(echantillon,matrice(:,6),':r')
%plot(echantillon,matrice(:,7),'.k')
hold off
legend(num2str(angled(1)),num2str(angled(2)),num2str(angled(3)),num2str(angled(4)))
%,num2str(angled(5)),num2str(angled(6)))%,num2str(angled(7)))
%axis([-20 20 -150 150])
xlabel('Number of samples','FontWeight','Bold','FontSize',12,'FontName','Times');
ylabel('error magnitude','FontWeight','Bold','FontSize',12,'FontName','Times');

grid
h = get(gcf,'CurrentAxes');
set(h,'FontWeight','Bold','FontSize',16,'FontName','Times','LineWidth',1);
set(gca,'xtick',0:2000:10000);
set(gca,'xtick',0:1000:10000);
```

eespritSNR.m

```matlab
%function [doa] = eesprit(doas)

clf;
clear all;
P = 5000;
SNR = -20:5:20
x0 = 50;
angled1 =[40 ];
angled2 =[40 65 ];
angled3 = [40 65 95 ];
angled =[40 65 95 120];
Niterations = 10000;

matrice = zeros(length(SNR),length(angled));
for idAngle=1:1:length(angled)
doas = [angled(idAngle)];
estimation = zeros(1,Niterations);

erreur = zeros(1,length(SNR));


%psi = diag(exp(sqrt(-1)*(pi/2)*sin(doas)));

el1 = [1 0 0 0].';
el2 = [0 1 0 0].';
el3 = [0 0 1 0].';
el4 = [0 0 0 1].';
zer = [0 0 0 0].';

J1 = [el3, el1, el2, zer, zer, zer, el4];
J2 = [el1, zer, zer, el2, el3, el4, zer];

% J2 = [el1, zer, zer, zer, el2, el3, el4];
% J1 = [el3, el4, el1, el2, zer, zer, zer];

react = [ 0 0 0 0 0 0 ;
    2047 2047 -2048 -2048 -2048 2047;
    2047 2047 2047 -2048 -2048 -2048;
    -2048 2047 2047 2047 -2048 -2048;
    -2048 -2048 2047 2047 2047 -2048;
    -2048 -2048 -2048 2047 2047 2047;
    2047 -2048 -2048 -2048 2047 2047
];
```

```
% react = [ 0 0 0 0 0 0 ;
%      1000 200 -400 -542 -1988 680;
%      2047 2047 2047 -2048 -2048 -2048;
%      -2048 2047 2047 2047 -2048 -2048;
%      -2048 -2048 2047 2047 2047 -2048;
%      -28 -204 -8 207 47 7;
%      2047 -2048 -2048 -2048 2047 2047
% ];

[M tmp] = size(react);
currents = zeros(M-1,7);
for index= 1:M
    imp = [x0; digvolt2imp(react(index,:)).'];
     current = rea2curb(imp(1),imp(2:7));
    currents(index,:) = current.';
end


icur = inv(currents);

rad2deg = 180/pi;
deg2rad = pi/180;

nsig = length(doas);

symbols = zeros(nsig,P);
for index = 1:nsig
    symbols(index,:) = 1-2*randint(1,P,2); % BPSK
end




A = [                  ones(1,nsig);
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*doas));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-60)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-120)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-180)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-240)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-300)))
];

psym = diag((symbols*symbols')/P);
noise = randn(M,P) + sqrt(-1)*randn(M,P);

%###############################################################################

% premiere boucle
```

65

```
%################################################################

for i = 1:length(SNR)
% Tous les signaux ont le meme SNR
sig = diag(sqrt((2*(10.^(SNR(i)/10.0)))./psym)) * symbols;




signal = ((currents*A)*sig);
x = signal + noise;

Ryy = x*x';
[Uy,Sy,Vy]=svd(Ryy);
sigma =0;
for indice = nsig+1:M
sigma = 1/(M-nsig)*(Sy(indice,indice))+sigma;
end

Rxx = icur*(Ryy-sigma*eye(M))*currents;

pa=nsig;

for iteration =1:Niterations
[Uxx, Sxx, Vxx] = svd(Rxx);
sv = diag(Sxx);


Es = Uxx(:,1:pa);
En = Uxx(:,pa+1:M);


Ux =J1*Es;
Uy =J2*Es;
Uz = [Ux'*Ux Ux'*Uy; Uy'*Ux Uy'*Uy];
[U,S,V]=svd(Uz);
Vn= V(:,pa+1:2*pa);
Phi = -Vn(1:pa,:)*inv(Vn(pa+1:2*pa,:));
[Up,Sp,Vp] = svd(Phi);
phi = eig(Phi);



estimation(iteration) = asin((2/pi)*angle(phi))*rad2deg +90 ;
end

erreur(i) = mean(estimation) - doas;
```

```
end
matrice(:,idAngle) = erreur.';
%plot(SNR,erreur)
%hold on;

end

%%%---- 1 erreur en fonction du SNR


% save snr1 angled SNR matrice Niterations SNR
figure(1);

plot(SNR,matrice(:,1),'linewidth',4)
hold on;
 plot(SNR,matrice(:,2),'+g','linewidth',4)
 plot(SNR,matrice(:,3),'--r','linewidth',4)
 plot(SNR,matrice(:,4),'.-k','linewidth',4)
% % % plot(SNR,matrice(:,5),'ob')
% % plot(SNR,matrice(:,6),'+r')
% % plot(SNR,matrice(:,7),'.k')
%  hold off
%  legend(num2str(angled(1)),num2str(angled(2)),num2str(angled(3)),num2str(angled(4)))
%,num2str(angled(5)),num2str(angled(6)),num2str(angled(7)))
%  axis([-20 20 -150 150])
 set(gca,'xtick',-20:5:20)
% title('Doa estimation of single impinging signal according to SNR')
 xlabel('SNR','FontWeight','Bold','FontSize',20,'FontName','Times');
 ylabel('error magnitude','FontWeight','Bold','FontSize',20,'FontName','Times')
  h = get(gcf,'CurrentAxes');
    set(h,'FontWeight','Bold','FontSize',20,'FontName','Times');
```

Appendix C:

combinaison.m

program name: combinaison.m
comments: The signal and the noise are generated

```
clear all;
P = 5000;
SNR = 0;
x0 = 50;
doas = [50];

% for the matrices J1 and J2
el1 = [1 0 0 0].';
el2 = [0 1 0 0].';
el3 = [0 0 1 0].';
el4 = [0 0 0 1].';
zer = [0 0 0 0].';

% the reactances
react = [ 0 0 0 0 0 0 ;
    2047 2047 -2048 -2048 -2048 2047;
    2047 2047 2047 -2048 -2048 -2048;
    -2048 2047 2047 2047 -2048 -2048;
    -2048 -2048 2047 2047 2047 -2048;
    -2048 -2048 -2048 2047 2047 2047;
    2047 -2048 -2048 -2048 2047 2047
];

[M tmp] = size(react);
currents = zeros(M-1,7);
for index= 1:M
    imp = [x0; digvolt2imp(react(index,:)).'];
     current = rea2curb(imp(1),imp(2:7));
    currents(index,:) = current.';
end

icur = inv(currents);
```

```
rad2deg = 180/pi;
deg2rad = pi/180;

nsig = length(doas);

symbols = zeros(nsig,P);
for index = 1:nsig
    symbols(index,:) = 1-2*randint(1,P,2); % BPSK
end




A = [                   ones(1,nsig);
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*doas));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-60)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-120)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-180)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-240)));
    exp(sqrt(-1)*(pi/2)*cos(deg2rad*(doas-300)))
];

psym = diag((symbols*symbols')/P);

% Tous les signaux ont le meme SNR
sig = diag(sqrt((2*(10.^(SNR/10.0)))./psym)) * symbols;
noise = randn(M,P) + sqrt(-1)*randn(M,P);

signal = ((currents*A)*sig);
x = signal + noise;

Ryy = x*x';
Rxx = icur*Ryy*currents;

pa=nsig;

[Uxx, Sxx, Vxx] = svd(Rxx);
sv = diag(Sxx);
Es = Uxx(:,1:pa);
En = Uxx(:,pa+1:M);

save data
```

TroisSet.m

program name: TroisSet.m
comments: Implementation of the technique using the three different subarray partition set 1a, set 2a and set 3a

```
clear all;
clf;
load data
trouve=0;
indice=0;
% set 1a
J11a = [el3, el1, el2, zer, zer, zer, el4];
J21a = [el1, zer, zer, el2, el3, el4, zer];


Ux =J11a*Es;
Uy =J21a*Es;
Uz = [Ux'*Ux Ux'*Uy; Uy'*Ux Uy'*Uy];
[U,S,V]=svd(Uz);
Vn= V(:,pa+1:2*pa);
Phi = -Vn(1:pa,:)*inv(Vn(pa+1:2*pa,:));
[Up,Sp,Vp] = svd(Phi);
phi = eig(Phi);

doa1a = asin((2/pi)*angle(phi))*rad2deg +90 ;
doa1b = -doa1a;




%set2a
 J12a = [el3, el4 ,el1, el2, zer, zer, zer];
 J22a = [el1, zer, zer, zer, el2, el3, el4];



 Ux =J12a*Es;
Uy =J22a*Es;
Uz = [Ux'*Ux Ux'*Uy; Uy'*Ux Uy'*Uy];
[U,S,V]=svd(Uz);
Vn= V(:,pa+1:2*pa);
Phi = -Vn(1:pa,:)*inv(Vn(pa+1:2*pa,:));
[Up,Sp,Vp] = svd(Phi);
phi = eig(Phi);

doa2a = asin((2/pi)*angle(phi))*rad2deg +150 ;
```

71

```
  doa2b = 120 - doa2a ;



%set 3a
J13a = [el3,zer,el4,el1,el2,zer,zer];
J23a = [el1,el4,zer,zer,zer,el2,el3];


 Ux =J13a*Es;
Uy =J23a*Es;
Uz = [Ux'*Ux Ux'*Uy; Uy'*Ux Uy'*Uy];
[U,S,V]=svd(Uz);
Vn= V(:,pa+1:2*pa);
Phi = -Vn(1:pa,:)*inv(Vn(pa+1:2*pa,:));
[Up,Sp,Vp] = svd(Phi);
phi = eig(Phi);

doa3a = asin((2/pi)*angle(phi))*rad2deg -150 ;

doa3b = -120 - doa3a ;


%%%---- Tracer les phi et les doas
th=0:pi/100:pi*2;
figure(1);
ags = exp(sqrt(-1)*doas*deg2rad);
plot(cos(doa1a*deg2rad),sin(doa1a*deg2rad),'rd',cos(doa1b*deg2rad),sin(doa1b*deg2rad),'b*',
cos(doa2a*deg2rad),sin(doa2a*deg2rad),'g^',cos(doa2b*deg2rad),sin(doa2b*deg2rad),'y*',
cos(doa3a*deg2rad),sin(doa3a*deg2rad),'mo',cos(doa3b*deg2rad),sin(doa3b*deg2rad),'ks',
real(ags),imag(ags),'ch','LineWidth',3);

hold on;
plot(sin(th),cos(th),'b-');
grid
legend('Estimation1a','Estimation1b','Estimation2a','Estimation2b',
'Estimation3a','Estimation3b','Real DoA');
axis([ -1.5 1.5 -1.5 1.5]);
title('Ambiguity discrimination')
axis square;
x=-1.5:1.5;
y=0;
plot(x,0,'m');
plot(x,sqrt(3)*x,'g');
plot(x,-sqrt(3)*x,'r');
hold off;
% doa1a
% doa1b
% doa2a
```

```
% doa2b
% doa3a
% doa3b

Tableau =[doa1a,doa1b,doa2a,doa2b,doa3a,doa3b];

Point0=0;
Point60=60;
Point120=120;
Point180=180;
Point240=240;
Point300=300;

sector=zeros(1,6);

for i=1:length(Tableau)
    if Tableau(i)<0
        Tableau(i)=Tableau(i)+360;
    end
end

for  i=1:length(Tableau)
    Tableau(i)
    if Point0 < Tableau(i) & Tableau(i) <Point60
        sector(1)=sector(1)+1
    end
    if Point60 <Tableau(i) & Tableau(i)<Point120
        sector(2)=sector(2)+1
    end
    if Point120 <Tableau(i) & Tableau(i)<Point180
        sector(3)=sector(3)+1
    end
    if Point180 <Tableau(i) & Tableau(i)<Point240
        sector(4)=sector(4)+1
    end
    if Point240 <Tableau(i) & Tableau(i)<Point300
        sector(5)=sector(5)+1
    end
    if Point300 <Tableau(i)
        sector(6)=sector(6)+1
    end
end

% Set 1a covers Sectors #1 2 3
% Set 1b # 4 5 6
% Set 2a # 2 3 4
% Set 2b # 5 6 1
% Set 3a # 6 1 2
% Set 3b # 3 4 5
```

73

```
for index=1:length(sector)
    if sector(index)>= max(sector)
    trouve=index;
    end
end
display('The subarray partition to use is');
if trouve==1
    display('Set3b');
    indice=6;
    end
     if trouve==2
    display('Set1a ');
   indice=1;
    end
     if trouve==3
    display('Set2a');
    indice=3;
    end
     if trouve==4
    display('Set3a');
    indice=5;
    end
     if trouve==5
    display('Set1b');
    indice=2;
    end
     if trouve==6
    display('Set2b');
    indice=4;
    end
Tableau(indice);
```