SCHOOL OF ENGINEERING & DESIGN
ELECTRONIC & COMPUTER ENGINEERING

Intelligent Joint Channel Parameter Estimation Techniques for Mobile Wireless Positioning Applications

Wei Li
BRUNEL UNIVERSITY

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

Supervisor: Dr Qiang Ni
Co-supervisor: Dr Wenbing Yao

Aug 2010

Abstract

Mobile wireless positioning has recently received great attention. For mobile wireless communication networks, an inherently suitable approach is to obtain the parameters that are used for positioning estimates from the radio signal measurements between a mobile device and one or more fixed base stations. However, obtaining accurate estimates of these location-dependent channel parameters is a challenging task. The focus of this thesis is on the estimation of these channel parameters for mobile wireless positioning applications. In particular, we investigate novel estimators that *jointly* estimate more than one type of channel parameters. We first perform a comprehensive critical review on the most recent and popular joint channel parameter estimation techniques. Secondly, we improve a state-of-the-art technique, namely the Space Alternating Generalised Expectation maximisation (SAGE) algorithm by employing adaptive interference cancellation to improve the estimation accuracy of weaker paths. Thirdly, a novel intelligent channel parameter estimation technique using Evolution Strategy (ES) is proposed to overcome the drawbacks of the existing iterative maximum likelihood methods. Furthermore, given that in reality it is difficult to obtain the number of multipath in advance, we propose a two tier Hierarchically Organised ES to jointly estimate the number of multipath as well as the channel parameters. Finally, we extend the proposed ES method to further estimate the Doppler shift in mobile environments. Our proposed intelligent joint channel estimation techniques are shown to exhibit excellent performance even with low Signal to Noise Ratio (SNR) channel conditions as well as robust against uncertainties in initialisations.

Acknowledgements

Needless to say, the completion of this thesis would not have been possible without the help and support of some amazing people around me. I would like to first express my deepest gratitude to Dr Qiang Ni for his great support and guidance. His suggestions and critical advice, in the course of numerous fruitful discussions we had, are highly appreciated, and were also a significant contribution to the success of this work.

I would like to sincerely thank Dr Wenbing Yao for giving me this opportunity in the first place, I have learned so much under her guidance and encouragement. It was a privilege and truly inspirational to have had worked with her. Many thanks go to Dr Peter Duffett-Smith for trusting in me and for his valuable feedback on many of the published work in this thesis.

I would like to acknowledge EPSRC and Cambridge Silicon Radio for their generous financial support for this research.

I also wish to express my appreciation for the examiners for taking the time to review this thesis.

Finally, heartfelt thanks go to my parents for their constant support and encouragement, who have done everything they could to help me complete this challenging chapter of my life.

Contents

List of Figures

List of Tables

Abbreviations

AWGN	**A**dditive **W**hite **G**aussian **N**oise
AIC	**A**kaike's **I**nformation **C**riterion
BS	**B**ase **S**tation
CPU	**C**entral **P**rocessing **U**nit
CRLB	**C**ramer-**R**ao **L**ower **B**ound
DoA	**D**irection **o**f **A**rrival
EA	**E**volutionary **A**lgorithm
EM	**E**xpectation **M**aximisation
ES	**E**volution **S**trategy
ESPRIT	**E**stimation of **S**ignal **P**arameters via **R**otational **I**nvariance **T**echniques
GPS	**G**lobal **P**ositioning **S**ystem
JADE	**J**oint **A**ngle and **D**elay **E**stimation
LBS	**L**ocation **B**ased **S**ervices
LLS	**L**ower **L**evel **S**trategy
LMDP	**L**ocation and **M**otion **D**ependent channel **P**arameters
LMS	**L**east **M**ean **S**quares
LoP	**L**ine **o**f **P**osition
LoS	**L**ine **o**f **S**ight
MDL	**M**inimum **D**escription **L**ength
ME	**M**ean **E**rror
MIMO	**M**ultiple-**I**nput **M**ultiple-**O**utput
ML	**M**aximum **L**ikelihood
MLE	**M**aximum **L**ikelihood **E**stimator
MS	**M**obile **S**tation
MSE	**M**ean **S**quare **E**rror

MUSIC	**MU**ltiple **SI**gnal **C**lassification
PIC	**P**arallel **I**nterference **C**ancellation
RMSE	**R**oot **M**ean **S**quare **E**rror
RSS	**R**eceived **S**ignal **S**trength
RToF	**R**ound **T**ime **o**f **F**light
SAGE	**S**pace-**A**lternating **G**eneralized **E**xpectation-maximization
SD	**S**tandard **D**eviation
SIC	**S**erial **I**nterference **C**ancellation
SNR	**S**ignal to **N**oise **R**atio
TDoA	**T**ime **D**ifference **o**f **A**rrival
ToA	**T**ime **o**f **A**rrival
ULA	**U**niform **L**inear **A**rray
ULS	**U**pper **L**evel **S**trategy

Notations

$\mathbf{A}^*$	Matrix complex conjugate
$\mathbf{A}^T$	Matrix transpose
$\mathbf{A}^H$	Matrix Hermitian transpose
$\mathbf{A}^\dagger$	Matrix pseudo inverse (Moore-Penrose inverse)
$\Re\{\}$	Real part
$\Im\{\}$	Imaginary part
Tr$\{\}$	Trace of a matrix
vec$\{\}$	Stacking columns of a matrix orderwise into a column vector
$E\{\}$	Statistical expectation
$\|\|\ \ \|\|_F$	Frobenius norm
$\otimes$	Kronecker product
$\diamond$	Hadamard product
$\circ$	Khatri-Rao product
$\mathbf{I}_N$	An identity matrix of dimension $N \times N$
$\mathbf{0}_N$	A vector of zeros of dimension $N \times 1$
$\mathbf{1}_N$	A vector of ones of dimension $N \times 1$
$\mathbf{e}_i$	A standard unit vector pointing to the ith axis
$\mathbf{\Pi}_N$	An exchange matrix (ones on its anti-diagonal and zeros elsewhere) of dimension $N \times N$
$x \in (a, b)$	x belongs to the set of real numbers between $a < x < b$
$x \in [a, b]$	x belongs to the set of real numbers between $a \leq x \leq b$

I dedicate this work to my loving wife, whose life have so much enriched mine.

Chapter 1

Introduction

1.1 Background and Motivation

Until recently, central issues of wireless systems have been topics like standards, bandwidth, availability, or module cost. In other words, the focus was on making wireless access commercially available. Now that wireless information is widely accessible, additional stimuli to the discussion arise from the user and application side. It becomes more and more clear that the next big step in wireless systems can not be expected by upgrading the status quo, i.e. just by increasing data rate. The growing demand for mobile Internet, and the need to create novel business and revenue models in wireless networks have motivated wireless carriers and their partners to develop and deploy new technology enablers for value added data service portfolios [1]. In this context, wireless positioning based services draw significant attention.

The initial driving force behind this interest is a regulation in the United States promulgated by the Federal Communications Commission, requiring wireless carriers to be capable of delivering the position of a wireless device (or mobile station) making an emergency call to emergency authorities by October 2001. These requirements are collectively known as the Enhanced 911 (E911) mandate. The details of the FCC requirement and current positioning technologies on trial can be found in [2]. Similarly in Europe, the 1999 Communications Review (COM 1999/539) set the date of 1 January 2003 for the carriers to make location information available for emergency authorities.

1.1.1 Location Based Services

While location services have been driven by emergency and security requirements imposed on the wireless networks in the past, it is evident that the commercial demands for location motivated products is now a major driving factor. Increasingly, application-level software will incorporate location information into its features to fully utilise such information once it becomes available. This lucrative market of Location Based Services (LBS) was worth over $1.6 billion world wide in 2009, and forecast to exceed $6 billion in 2012 [3]. Below are some key applications that illustrate the potential of LBS [4]:

- *Asset tracking:* Wireless location technology can assist in advanced public safety applications, such as locating and retrieving lost children, patients, or pets. It can also be used to track personnel/assets in a hospital or manufacturing site to provide more efficient management of assets and personnel.
- *Fleet management:* Fleet operators such as police forces, emergency vehicles, and other services like shuttle and taxi companies can make use of the wireless location technology to track and operate their vehicles in an efficient manner to minimise response time. In addition, mobile phone locations of drivers on the road can be used to transform into sources of real-time traffic information to enhance transportation efficiency.
- *Wireless access security:* Location based wireless security schemes can be developed to restricting data access to a certain physical area, thus enhancing network security.
- *Mobile advertising:* Stores can use customer locations by using location specific advertising and marketing to attract customers in certain areas.

Other applications include: navigation, electronic yellow pages, location-sensitive billing etc; some of which is already available on the latest handset today.

In terms of system enhancement, location technologies provide wireless carriers with the means of improving wireless communications systems design and performance. Location information has been used to improve routing for ad hoc networks and for network planning where radio resources are dynamically redistributed to improve coverage and

capacity in area regularly visited by users. Location information can also play an important role in assisting handoffs between cells and in designing hard and soft handoff schemes for wireless networks [5].

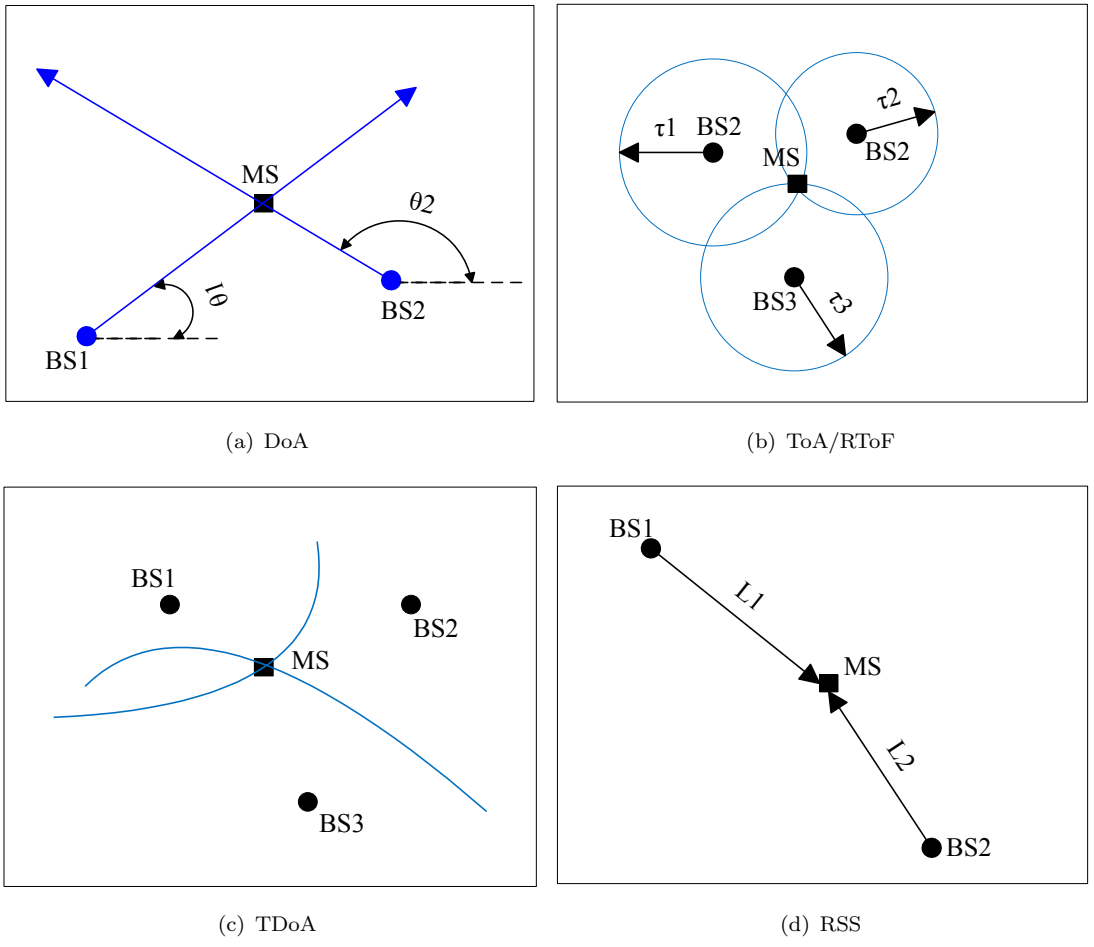
1.1.2 Wireless Positioning Technologies

Wireless positioning technologies fall into two main categories: *mobile based* and *network based*. In mobile based positioning systems, the parameters that are used to compute an MS's position are measured at the MS and is either then used in the MS to calculate its position or transferred to a central processing facility for position estimate. In network based solutions, the same parameters are measured at the BSs and transferred to the central facility for location determination. A significant advantage of network based techniques is that the MS is not involved in the location finding process; thus, the technology does not require modification to existing handsets [4]. The work undertaken in this thesis is primarily targeted at network based solutions, while it can also be adapted to mobile based systems with minimal modification. What follows is a brief overview of the foundations of wireless positioning technologies; numerous survey literature exists on the subject, and the interested readers are referred to [4–16] and the references therein for an extensive survey of different technologies and applications.

Global Positioning System (GPS) [17], a satellite based positioning system, is a popular solution for providing location in terrestrial wireless networks. It is a proven technology and provides high accuracy when a Line of Sight (LoS) path exists between the receiver and at least four satellites. However, GPS is not applicable in areas where satellites are blocked, e.g. indoors and built-up urban areas. Furthermore, the time to first fix for a conventional GPS receiver from a "cold" start can take several minutes. Additionally, adding GPS functionality to mobile devices can be costly and drain battery power at an unacceptable rate [5]. In addition, GPS based solutions can only apply to new generation of handsets and cannot be used with legacy handset devices. For the above mentioned reasons, some wireless providers may be unwilling to embrace GPS as the sole location technology.

Basically, a wireless positioning system consists of at least two separate hardware components [6]: a measuring unit that usually carries the major part of the system's "intelligence", and a signal transmitter. In network based systems, the signal transmitter is

the mobile, while several fixed BSs takes measurements from the mobile. Three different measurement principles are mainly used today: Direction of Arrival (DoA), Received Signal Strength (RSS), and propagation based systems that can be divided into three subclasses: Time of Arrival (ToA), Round Time of Flight (RToF) and Time Difference of Arrival (TDoA). Theses are commonly referred to in positioning terms as Location and Motion Dependent channel Parameters (LMDP).



(a) DoA

(b) ToA/RToF

(c) TDoA

(d) RSS

FIGURE 1.1: Positioning principles using different measuring units: DoA, ToA/RToF, TDoA and RSS. Blue lines represents LoPs. Note: the arrows do not represent direction of wave propagation.

Figure 1.1 illustrates the principles of each concept for the 2-dimensional case. DoA based systems (Figure 1.1(a)), measure the angle θ, at which a signal is incident at a BS whose position is known. Given the bearing of the incoming signal θ, a linear Line of Position (LoP) can be formed on which the MS lies. In the 2D case, a minimum of two such measurements are required at two BSs to form two linear LoPs whose intersection is the estimate of the MS position. The DoA estimates can be obtained through the

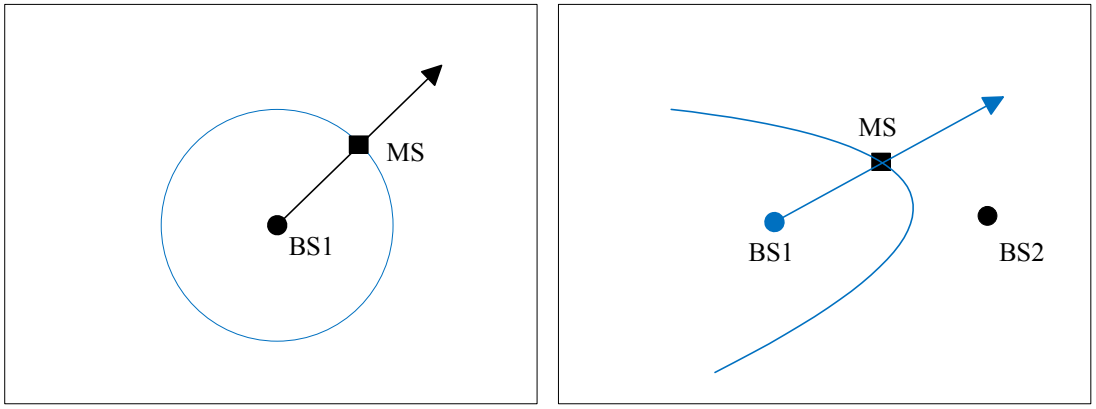
use of antenna arrays or directional antennas, hence the accuracy of position estimate is primarily affected by the angular resolution of the antenna setup, shadowing and by multipath reflections arriving with misleading directions [7].

In the ToA approach (Figure 1.1(b)), the time τ taken for the signal to travel from the MS to BS is measured, since radio waves travel at a constant velocity in free space, then the LoP is a circle representing constant distance from the BS. Because of this, ToA measurements from the MS and three BSs are required in order to resolve the ambiguity that only two measurements produce in the 2D case. A major issue with the ToA concept is the requirement for precise time synchronisation of all the involved BSs and MS. The same LoPs are present in the RToF case, where the time taken for the signal to travel from MS to BS then back to the MS is measured. This approach relaxes the absolute synchronisation requirement in ToA measurements, since only the delay/processing time of the transponder (the MS) needs to be known.

TDoA (Figure 1.1(c)) determines the MS position by measuring the differences in time between two BS and the MS. Since a hyperbola is a curve of constant time between a point on the hyperbola and its two foci, the LoPs for TDoA are hyperbolic. Because two BSs are required to form a single LoP, three BSs are required to generate two LoPs whose intersection provide the location of the MS. A major advantage of this approach is that it is only necessary to synchronise the measuring units (the BSs).

RSS systems (Figure 1.1(d)) are based on the propagation path loss of transmitted radio waves. Similar to propagation based systems, circular LoP of constant distance can be drawn using the free space transmission loss $L_B \sim 1/d^2$. However, this simple equation alone is inadequate for distance calculation in most real conditions [6]. Typically, advanced propagation models or actual measured field distributions in the area of interest are required also.

In many instances, several different types of measurements are available that can be used to improve an MS position estimate. The smart integration of different types of measurements obtained from different sources to improve positioning accuracy is referred to as hybrid positioning techniques [8]. Some simple forms of hybrid techniques are the integration of GPS with network based positioning in assisted-GPS [9]; or using the available serving cell information to narrow down position estimates [5].



(a) DoA/ToA (b) DoA/TDoA

FIGURE 1.2: Hybrid positioning principles based on DoA/ToA measurements and DoA/TDoA measurements. Blue lines represents LoPs. Note: the arrows do not represent direction of wave propagation.

This study is particularly interested in the hybrid directional/propagational LMDP (DoA/ToA or DoA/TDoA) fusion scenarios. The simplest form of DoA/ToA location is classic radar, where a direction and range from a known point provide a location estimate of the pinged target. This approach is particularly useful near the BS where LoS propagation is more likely and the effect of angular error is reduced. It has the additional benefit that location estimation is possible with only a single BS. This is shown in Figure 1.2(a) where the intersection of the linear and circular LoPs from the DoA and ToA measurements indicate position estimate of the MS. Hybrid DoA/TDoA methods can also be implemented in which MS position lies at the intersection of the linear and hyperbolic LoPs from the DoA and TDoA measurements. Thus, positioning of the MS is possible with only two BSs, as illustrated in Figure 1.2(b). Needless to say, this type of hybrid positioning is feasible only if the measuring BS is equipped with a mechanically steered narrow beamwidth antenna or with a fixed array of antennas. Fortunately, the utilisation of antenna arrays is already becoming widespread though the development of Multiple Input Multiple Output (MIMO) technologies in current and future cellular/wireless standards, e.g. 3G, LTE [18], 802.11n [19], 802.16e [20].

The straightforward approach for estimating the location of the MS using the above LMDP is to compute the geometric intersection of the LoPs directly as shown in Figures 1.1 and 1.2. Although this is an intuitive approach for position estimation, it does not provide an efficient data fusion mechanism, i.e. cannot utilise multiple parameter estimates in an efficient manner [10]. In practice, the LMDP measurements include noise,

and the LoPs do not necessarily intersect at a unique point. In such cases, the geometric techniques do not provide any insight as which point to choose as the position of the MS. Furthermore, in over-determined systems, where the number of measurements is more than the minimum required, the number of intersections can increase even further. Therefore, in practice, more advanced data fusion methods should be used. An overview of different types of measurement fusion techniques for wireless positioning can be found in [4, 5].

1.1.3 Challenges in LMDP Estimation

It is clear that all of the wireless positioning methods depend on combining estimates the LMDP obtained at one or more fixed BSs, while the less accurate these estimates are, the more advanced data fusion technique is required to compensate. Consequently, it is not surprising that the quality of the LMDP estimates is the main controlling factor in the performance of any wireless positioning systems. Estimating these channel parameters has been studied in many works since it is required in many wireless system design for online signal decoding purposes [4]. Yet, estimating the same parameters for wireless positioning purposes is challenging for several reasons:

- *Low SNR and Hearability:* The strength of the signal received at each BS directly impacts the quality of the LMDP estimates. In the case of very low SNR signals, the ability to obtain good LMDP estimates can be significantly impaired [15]. Mobile systems tend to suffer from high multiple access interference levels that degrade the SNR of the received signal. In highly congested wireless environments, the interference has the potential to be a significant degrading factor in LMDP estimates. Moreover, the ability to detect the MS signal at multiple BSs is limited by the use of power control algorithms, which require the MS to decrease the transmitted power when it approaches the serving BS. This in turn, decreases the received MS signal power levels at other BSs. In rural environments where the separation between cell cites is large, it is often difficult for the MS to be heard at BSs other than its serving BS, unless the MS is located in a hand-off region.

- *Multipath:* For positioning systems, the accurate estimation of the channel parameters of the first arriving ray is vital. In general, the first arriving ray is assumed

to correspond to the most direct path between the MS and BS. However, in many wireless propagation scenarios, the first ray is succeeded by many multipath component that arrives at the receiver within a short time of the first ray. These signals can combine constructively or destructively and result in the phenomenon known as multipath or fading (possibly leading to reduced SNR) and increase the variance of LMDP estimates. If the differences in delays between these multipath is smaller than the pulse shape used in the wireless systems, these rays will overlap and the system is unable to resolve the multipath.

- *Number of Multipath:* One common assumption in most existing research in channel parameter estimation is that the number of channel multipath is known in advance. In practice however, this number has to be estimated first, either via a separate mechanism or jointly with the channel parameters. Clearly, accurate knowledge of the number of paths is essential to the performance of all estimators, as the effects of under- or over- estimating the channel parameters is not isolated to any one path, and will most likely influence the estimation accuracies of other paths too.

- *Line of Sight:* In most urban and indoor scenarios, due to obstruction by buildings, walls and other objects, the LoS propagation path is not always the strongest. In some cases, it may not even be detectable with a specific receiver implementation [16]. In these scenarios, it is not possible for the standard fusion techniques in Figures 1.1 and 1.2 to obtain accurate position estimates from the estimates of the non-LoS LDMP. Although in LDMP estimation, we always assume the presence of the LoS component, the accurate estimation of an LoS in case it is weak (weak-LoS) is rather challenging. This is due to the fact that the estimation performance of most methods varies significantly between strong and weaker paths.

1.2 Aims and Objectives

The aim of this work is to study and develop novel channel parameter estimation methods for the application of mobile wireless positioning within future generation wireless networks. Our objectives can be summarised as follows:

- To study and compare the current and state-of-the-art channel parameter estimation methods for use in future generation wireless networks.
- Develop novel estimators that efficiently overcome the main drawbacks of the existing approaches without increasing the computational requirement.
- Develop a robust and efficient method to jointly estimate the channel order and the channel parameters.

1.3 Thesis Outline

The focus of this thesis is on the estimation of the LMDP for wireless positioning applications, in particular, estimators that *jointly* estimate more than one type of channel parameters (i.e. ToA/DoA). These are particular useful (but not limited to) for the hybrid positioning techniques mentioned previously, as well as for receiver design in space-time communication systems. The rest of this thesis is organised as follows:

- In Chapter 2, we present a comprehensive critical review of some of the most recent and popular joint channel parameter estimation techniques. Empirical analysis of a number of key issues concerning these methods is presented.
- In Chapter 3, the damaging effects of accumulated estimation errors from brute force interference cancellation adopted by the standard SAGE algorithm is described. An improved SAGE algorithm employing adaptive interference cancellation scheme is proposed to improve the estimation accuracy of weaker multipath.
- In Chapter 4, a novel intelligent channel parameter estimation technique based on the application of Evolution Strategy, which overcomes many of the issues inherent with iterative maximum likelihood methods such as SAGE is proposed.
- In Chapter 5, a two tier Hierarchically Organised Evolution Strategy is proposed to jointly estimate the number of multipath as well as the channel parameters.
- In Chapter 6, the proposed ES method in Chapter 4 is extended to further estimate the Doppler shift of each multipath.
- In Chapter 7, the main contributions of the thesis are summarised and possible future works are discussed.

1.4 Contributions

The main contributions of the thesis are briefly summarised here:

- We performed a comprehensive critical review of the most recent and popular joint channel parameter estimation techniques based on empirical analysis. Through simulations designed to test each of the surveyed methods in areas of particular importance in positioning applications, we concluded that the SAGE algorithm is a strong candidate for use in LMDP estimation.
- We proposed an improved SAGE algorithm employing adaptive interference cancellation to overcome a key weakness inherent with the standard SAGE algorithm in the estimation of the weaker multipath components. Our proposed modifications enable the SAGE algorithm to be less susceptible to errors accumulated from successive interference cancellation steps and improves the estimation performance of weaker multipath significantly.
- We proposed a novel intelligent channel parameter estimation technique by using an Evolution Strategy approach. Our proposed method overcomes the weaknesses of traditional iterative maximum likelihood methods like SAGE, such as low SNR performance, dependency on accurate initialisations and high computational complexity. Through an extensive empirical analysis of the strategy parameters, we show the proposed method is highly flexible, self-manageable, and less computationally demanding than SAGE. We also demonstrate that the proposed method can be utilised to effectively estimate the Doppler shifts in mobile environments.
- We proposed a two tier Hierarchically Organised Evolution Strategy to jointly estimate the number of multipath as well as the channel parameters. The proposed method is demonstrated to be robust against errors in the initialisation of both the channel order and channel parameters. In addition, it does not require delicate tuning of strategy parameters.

1.5 List of Publications

- W. Li, Q. Ni, "Joint Channel Parameter Estimation Using Evolutionary Algorithm," in *IEEE International Communications Conference*, May. 2010, pp. 1-6.

- W. Li, W. Yao, and P. Duffett-Smith, "Improving the SAGE Algorithm with Adaptive Partial Interference Cancellation," in *13th IEEE Digital Signal Processing Workshop and 5th IEEE Signal Processing Education Workshop*, Jan. 2009, pp. 404-409.

- W. Li, W. Yao, and P. Duffett-Smith, "Comparative Study of Joint TOA/DOA Estimation Techniques for Mobile Positioning Applications," in *6th IEEE Consumer Communications and Networking Conference*, Jan. 2009, pp. 1-5.

- W. Li, W. Yao, and P. Duffett-Smith, "A Comparative Study of JADE and SAGE Algorithms for Joint Multipath Parameter Estimation," in *15th IEEE International Conference on Digital Signal Processing*, Jul. 2007, pp. 59-62.

Chapter 2

Critical Review of Existing Approaches

2.1 Introduction

Traditionally, channel parameter estimation methods are based on disjoint techniques. For example, they normally estimate first the delays and, subsequently, the angle corresponding to each delay; an additional problem of the correct pairing of channel parameters often arises. Recently, various joint channel parameter estimation methods emerged as potential candidate algorithms for use in the mobile terminal positioning technology. Joint estimation of various channel parameters, particularly joint ToA/DoA estimation, has a number of advantages [21]. Firstly, the relative estimates of time delay measured at two or more synchronised base stations can be used in conjunction with DoA information measured at each of them to enhance positional accuracy. Secondly, when signals arriving from several angles are detected, the best angle to use is usually the one that is associated with the earliest arrival time because the direct path has the shortest propagation time. Thirdly, it is possible to exploit the difference in path time delays to improve angle estimation accuracy and vice-versa. Finally, joint estimation can resolve paths having identical directions or times of arrival.

The joint ToA/DoA estimation algorithms can be classified into two broad groups, based on their development and fundamental philosophy:

eigen-decomposition This type of methods rely on each parameter being estimated from a certain eigenvalue problem, where all eigenvalue problems share the same eigenvectors [22]. This allows the estimation problem to be posed as a joint diagonalisation of a collection of data matrices. Such methods include the Joint Angle and Delay Estimation (JADE) technique [23–25], and the Estimation of Signal Parameters via Rotational Invariance Techniques (ESPRIT) [26]. One of the strict conditions imposed on these approaches is that the antenna array must exhibit the Vandermonde structure in its steering matrix so that the invariance equations can be formed between data across different sub-arrays.

maximum likelihood The Maximum Likelihood (ML) method has the asymptotic properties of being unbiased [27]. This type of methods include the Expectation Maximisation (EM) [28, 29], the Space-Alternating Generalised Expectation maximisation (SAGE) [30, 31], and the Iterative Quadratic Maximum Likelihood (IQML) algorithm, which is known to offer ML performance for sufficiently high SNR [32]. These methods normally require high computational and long processing time, since they are iterative in nature.

While most of these algorithms have been applied with some success either to measured data or realistic channel models, no single comparative study of their relative performance has been reported. The most recent comprehensive surveys are reported in [33–35], which cover the classical array processing techniques for parameter estimation, such as ML, Beamforming, Weighted Subspace Fitting (WSF), and MUltiple SIgnal Classification (MUSIC). Although there exist multi-parameter approaches to most of these algorithms, these literatures only cover single parameter estimation, and not the joint case. A comparison between the SAGE and ESPRIT algorithms for 3D channel sounding has been undertaken by Tschudin *et al.* [36] in a study which concentrated on resolvability and multipath identification issues. However, this is not particularly relevant for mobile positioning where knowledge of the accuracy of ToA and DoA estimation is the priority.

This chapter therefore presents a critical review of the most recent joint parameter estimation techniques, namely ESPRIT, JADE, and SAGE. We implemented the 2D Unitary ESPRIT [37, 38], Shift Invariance-JADE (SI-JADE) [39] and SAGE algorithms in MATLAB, and carried out Monte Carlo simulations in multipath environments. We

focus on a number of core performance issues such as the estimation accuracy, spatial/temporal resolution, multipath identification; as well as what happens when the order of the channel model is not accurately estimated. In addition, we also analyse the computational complexity of the algorithms with respect to the number of paths in the channel and the number of antenna elements.

2.2 System Model

A typical wireless multipath propagation channel is illustrated in Figure 2.1, where data symbols are modulated by a known pulse shape at the mobile device and transmitted through the spatial multipath channel. The transmitted signal then undergoes a series of scattering processes (namely diffraction, reflection and refraction) before arriving at the M elements of an Uniform Linear Array (ULA) at the base station. It is assumed in the underlying channel model that a finite number L of specular plane waves (paths) are impinging at the receiver location.

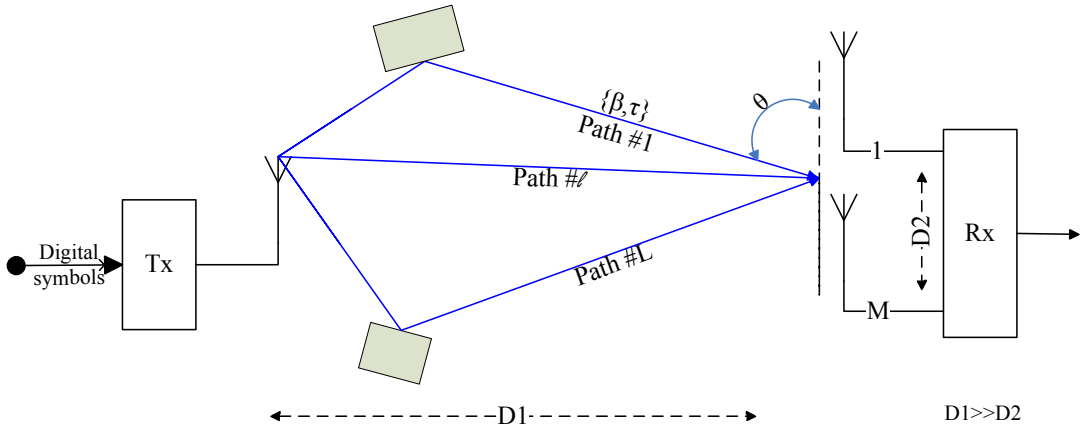


FIGURE 2.1: Illustration of a wireless multipath environment. Each path is parametrised by its ToA τ_ℓ, DoA θ_ℓ, and complex path attenuation β_ℓ. A total of L path is assumed

Assuming the digital sequence $\{b_k\}$ is transmitted over the the wireless channel and the response was measured using a M-element ULA, then the received signal vector

$\mathbf{y}(t) \triangleq [y_1(t), \ldots, y_M(t)]^T$ at the output of the antenna array in general has the form:

$$\mathbf{y}(t) = \sum_k b_k \mathbf{h}(t - kT) + \sqrt{\frac{N_o}{2}}\boldsymbol{\eta}(t) \tag{2.1}$$

where T is the symbol period and the vector $\boldsymbol{\eta}(t) \triangleq [\eta_1(t), \ldots, \eta_M(t)]^T$ denotes a standard M-dimensional complex Additive White Gaussian Noise (AWGN) with power spectral density $N_o/2$; N_o is a positive constant.

Since there are assumed to be L number of paths in this specular multipath environment, where each path is parametrised by its DoA θ_ℓ, ToA τ_ℓ, and a complex path attenuation β_ℓ, which is assumed to be a constant within a symbol period. The channel can thus be modelled as the M-element impulse response vector:

$$\mathbf{h}(t) = \sum_{\ell=1}^{L} \mathbf{a}(\theta_\ell)\beta_\ell g(t - \tau_\ell) \tag{2.2}$$

where $g(t)$ is a known pulse shape function by which $\{b_k\}$ is modulated.

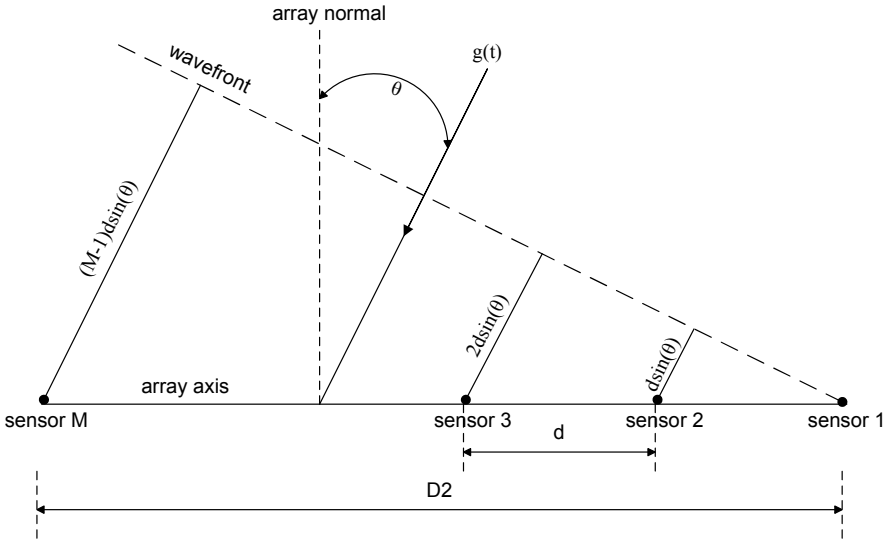


FIGURE 2.2: An ULA of M elements and inter-element spacing $d = D2/(M-1)$ along with an impinging planar wavefront.

The $M \times 1$ vector $\mathbf{a}(\theta_\ell)$ in (2.2) is known as the steering vector, which describes the nominal array response to the source impinging on the array from the direction θ_ℓ. Assuming that the MS is in the far-field of the receiver antenna array, i.e. the distances D_1 and D_2 in Figure 2.1 obey $D_1 \gg D_2$; then the impinging signal on the sensor array

is approximately a uniform plane wave, as shown in Figure 2.2. Then for the M-element ULA with spacing d between adjacent elements, each ith sensor experiences a time delay of

$$\triangle\tau = \frac{d\sin(\theta)}{c} \tag{2.3}$$

with respect to the $(i-1)$th sensor, where $c \approx 3 \times 10^8$ is the speed of light. If $g(t)$ is a narrowband signal[1] with carrier frequency f_c, the time delay $\triangle\tau$ corresponds to a phase shift of

$$\triangle\phi = 2\pi\frac{d\sin(\theta)}{\lambda} \tag{2.4}$$

between individual antenna elements, where λ is the wavelength corresponding to the carrier frequency f_c, i.e. $\lambda = c/f_c$. Hence, each element in the ULA receives a phase-shifted version of the signal, and the steering vector $\mathbf{a}(\theta_\ell)$ represents the relative phase difference between adjacent elements:

$$\mathbf{a}(\theta_\ell) = \begin{bmatrix} 1 \\ e^{-j\frac{2\pi}{\lambda}d\sin(\theta_\ell)} \\ e^{-j\frac{2\pi}{\lambda}2d\sin(\theta_\ell)} \\ \vdots \\ e^{-j\frac{2\pi}{\lambda}(M-1)d\sin(\theta_\ell)} \end{bmatrix} \tag{2.5}$$

where the first element (reference element) usually is set to have zero phase. For a range of angles $\theta \in [-\pi/2, \pi/2]$, $\mathbf{a}(\theta_\ell)$ maintains its unambiguity provided $d < \lambda/2$. For more widely spaced sensors, it is possible that there may exist pairs of angles θ_ℓ and $\theta_{\ell'}$, with $\theta_\ell \neq \theta_{\ell'}$, such that $\mathbf{a}(\theta_\ell) = \mathbf{a}(\theta_{\ell'})$. This equality holds when $d/\lambda\sin(\theta_\ell) = n + d/\lambda\sin(\theta_{\ell'})$, where $n \in \mathbb{Z}^+$. In such cases, the array response for a signal arriving from angle θ_ℓ is indistinguishable from that arriving from angle $\theta_{\ell'}$ [40].

After taking N regularly spaced samples at P times the symbol rate, the received data during each symbol period can be written as:

$$\begin{aligned} \mathbf{Y} &= [\mathbf{a}(\theta_1), \ldots, \mathbf{a}(\theta_L)] \begin{bmatrix} \beta_1 & & 0 \\ & \ddots & \\ 0 & & \beta_L \end{bmatrix} \begin{bmatrix} \mathbf{g}^T(\tau_1) \\ \vdots \\ \mathbf{g}^T(\tau_L) \end{bmatrix} + \mathbf{N} \\ &= \mathbf{A}(\boldsymbol{\theta})\mathbf{B}\mathbf{G}^T(\boldsymbol{\tau}) + \mathbf{N} \end{aligned} \tag{2.6}$$

[1] the standard narrowband assumption in array signal processing assumes signal bandwidth is much smaller than the time needed to travel across the array length

where $\mathbf{g}(\tau_\ell) = [\mathbf{g}(t_k - \tau_\ell)], k = 1 \dots N$, is an $N \times 1$ column vector containing samples of $g(t - \tau_\ell)$. The $M \times L$ Vandermonde matrix $\mathbf{A}(\boldsymbol{\theta}) = [\mathbf{a}(\theta_1), \dots, \mathbf{a}(\theta_L)]$ is known as the steering matrix (also sometimes as the array manifold).

In order to estimate the channel parameters $[\beta_\ell, \tau_\ell, \theta_\ell]$ for $\ell = 1 \dots L$, the following assumptions are generally made:

- The number of sources is small. For convenience, we consider only one source in a multipath environment, but this is no limitation.
- The number of paths is often assumed to be known. In reality, this number normally needs to be estimated. Minimum Description Length (MDL) and Akaike's Information Criterion (AIC) are two algorithms traditionally used for this purpose [41].

2.3 Simulation Environment

This section describes the common simulation environment used throughout the thesis, while non-generic simulation details can be found in individual chapters where appropriate. All of the simulation works are performed using the mathematical software MATLAB [42].

The details of the channel and signal parameters used in the simulations were as follow. The known transmitted signal $g(t)$ employed throughout the simulations is the Raised Cosine pulse shape of excess bandwidth 0.35. The length of the pulse shape is $8T$, where T is the symbol period of the Raised Cosine filter. The received data in (2.6) is observed over the interval T_o at the receiver using a 4-element ULA with inter-element spacing of $d = \lambda/2$; and sampled 4 times per symbol period T. We assume horizontal propagation only, so the term DoA refers to the azimuth direction of arrival, while all delays estimates are normalised to the symbol period T. Unless otherwise stated, 500 Monte-Carlo simulations were performed for each test scenario and averaged to obtain the final result. The Doppler effects were neglected[2].

[2]Doppler estimation is addressed in Chapter 6

2.4 The Eigen-decomposition Approach

In many practical cases, optimal techniques that require searching the solution space are often computationally prohibitive, whereas the techniques that exploit prior knowledge of the algebraic structures of the data matrices are often less computationally demanding. These methods often rely on the fact that each parameter is estimated from a certain eigenvalue problem, where all eigenvalue problems share the same eigenvectors. This allows the posing of the problem as a joint diagnalisation problem of a collection of data matrices [22]. This is the underlying principle of the Multidimensional-ESPRIT algorithm, as well as numerous ESPRIT based parameter estimation methods.

Consider initially the one-dimensional problem of DoA estimation. Assuming that L signals from far-field uncorrelated sources impinges on the array, then the $M \times 1$ array snapshot vector ($M > L$) can be written as

$$\mathbf{x}(t) = \mathbf{A}(\boldsymbol{\theta})\mathbf{s}(t) + \boldsymbol{\eta}(t) \tag{2.7}$$

where $\mathbf{s}(t)$ is the $L \times 1$ vector of the source waveforms and $\boldsymbol{\eta}(t)$ is the $M \times 1$ vector of zero mean sensor noise that is assumed to be spatially white and to have equal variance σ^2 in each sensor.

Methods which exploit the algebraic structure of the data matrix generally involve the eigen-decomposition of the autocorrelation matrix of the preceding model in (2.7)

$$\mathbf{R} = E\{\mathbf{x}\mathbf{x}^H\} = \mathbf{A}\mathbf{S}\mathbf{A}^H + \sigma^2\mathbf{I} \tag{2.8}$$

where $\mathbf{S} = E\{\mathbf{s}\mathbf{s}^H\}$ is the source covariance matrix. Notice that $\mathbf{S}$ is diagonal when the signals are uncorrelated, non-diagonal and non-singular when the signals are partially correlated, and non-diagonal but singular when some signals are fully correlated (coherent).

Since the ensemble average in (2.8) is a purely mathematical concept, the correlation matrix is typically estimated in practice using limited data samples as:

$$\hat{\mathbf{R}} = \frac{1}{N}\sum_{t=1}^{N}\mathbf{x}(t)\mathbf{x}^H(t) \tag{2.9}$$

where N is the number of snapshot available, this is better known as the sample correlation matrix.

Since the columns of the matrix $\mathbf{A}$ is different if the DoA's are unique, and hence, because of their Vandermonde structure, linearly independent. If $\mathbf{S}$ is also non-singular then the rank of $\mathbf{ASA}^H$ is L. Assuming

$$\{\lambda_1 \geq \lambda_2 \geq \ldots \lambda_M\} \text{ and } \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_M\} \tag{2.10}$$

are the eigenvalues and the corresponding eigenvectors of $\mathbf{R}$, then the above rank properties imply that

1. the $M - L$ smallest eigenvalues of $\mathbf{R}$ are all equal to the noise variance σ^2. The eigenvectors corresponding to these $M - L$ smallest eigenvalues of $\mathbf{R}$ are called noise eigenvectors, while the eigenvectors corresponding the L largest eigenvalues are called signal eigenvectors.

2. the subspace spanned by the noise eigenvectors, referred to as the *noise subspace* $\mathbf{E}_n$ (a $M \times (M - L)$ matrix), are orthogonal to the subspace spanned by the the columns of the matrix $\mathbf{A}$, known as the *signal subspace* $\mathbf{E}_s$ (a $M \times L$ matrix).

$$\{\mathbf{v}_{L+1}, \ldots, \mathbf{v}_M\} \perp \{\mathbf{a}(\theta_1), \ldots, \mathbf{a}(\theta_L)\} \tag{2.11}$$

When the source waveforms are coherent (e.g. multipath), the source correlation matrix $\mathbf{S}$ loses rank and becomes singular. In these cases, a useful pre-processing technique known as spatial sub-array smoothing [43, 44] or the improved forward/backward averaging [45–47] is often used to obtain a smoothed estimate of the $\mathbf{R}$. Where each snapshot across the entire array measurement is divided into equal sized overlapping segments (known as a sub-array) and averaged. Hence the number of independent measurements can be increased but at the cost of reduced array aperture to size of the sub-array.

2.4.1 ESPRIT

The ESPRIT algorithm [26] was first introduced as a search-free parameter estimation procedure for undamped cisoids in noise. Due to its low computational complexity, it has become one of the most widely-used high-resolution parameter estimation techniques to

date. For classical array signal processing, the ESPRIT algorithm lessens the *a priori* information requirement of complete knowledge of the array manifold $\{\mathbf{a}(\theta) : \theta \in \Theta\}$, where Θ is the field of view; by imposing a constraint on the structure of the antenna array, known as displacement invariance, i.e. the antenna array is composed of two identical sub-arrays displaced relative to each other by a known distance and direction [48]. Although the array geometry can be arbitrary, the ULA is usually employed to reduce the total number of elements by overlapping the two sub-arrays [41]. The locations of the second sub-array elements are at a constant displacement from the corresponding elements of the first sub-array. The parameter estimates are obtained by exploiting the underlying rotational invariance of the signal subspaces spanned by two temporally displaced data vectors, induced by the structure associated with the overlapped antenna array.

2.4.1.1 1D-ESPRIT

The basis of the standard ESPRIT algorithm for DoA estimation is the so-called invariance equation. Since sub-array 2 is a constant shift of the identical sub-array 1, their steering matrices are related by a rotational operator $\mathbf{\Phi}$, i.e.

$$\mathbf{J}_1\mathbf{A}\mathbf{\Phi} = \mathbf{J}_2\mathbf{A} \tag{2.12}$$

where $\mathbf{J}_1$ and $\mathbf{J}_2$ are the selection matrices used to choose the elements of the two sub-arrays from the entire array; and $\mathbf{\Phi}$ is a diagonal matrix

$$\mathbf{\Phi} = \begin{bmatrix} e^{j\phi_1} & & \\ & \ddots & \\ & & e^{j\phi_L} \end{bmatrix} \tag{2.13}$$

where the exponential components ϕ_ℓ contains the parameter of interest. The invariance equation relates the subset $\mathbf{J}_1\mathbf{A}$ via a phase rotation to $\mathbf{J}_2\mathbf{A}$.

Since the signal subspace $\mathbf{E}_s$ spans the same subspace as the columns of $\mathbf{A}$, i.e.

$$\mathbf{A}\mathbf{T} = \mathbf{E}_s \tag{2.14}$$

where $\mathbf{T}$ is an arbitrary full rank matrix $\mathbf{T} \in \mathbb{C}^{L\times L}$. Replacing the exact signal subspace $\mathbf{E}_s$ with the estimates $\hat{\mathbf{E}}_s$ and inserting equation (2.14) in (2.12) yields

$$\mathbf{J}_1\hat{\mathbf{E}}_s\mathbf{T}^{-1}\mathbf{\Phi} \approx \mathbf{J}_2\hat{\mathbf{E}}_s\mathbf{T}^{-1} \tag{2.15}$$

Consequently, the least squares solution of

$$\hat{\mathbf{\Psi}}_{LS} = \arg\min \parallel \mathbf{J}_1\hat{\mathbf{E}}_s\mathbf{\Psi}_{LS} - \mathbf{J}_2\hat{\mathbf{E}}_s \parallel_F^2 \in \mathbb{C}^{L\times L} \tag{2.16}$$

has approximately the structure

$$\hat{\mathbf{\Psi}}_{LS} = \mathbf{T}^{-1}\mathbf{\Phi}\mathbf{T} \tag{2.17}$$

This says that $\hat{\mathbf{\Psi}}_{LS}$ and $\mathbf{\Phi}$ are *similar* and, hence, they have the same eigenvalues, which are the diagonal elements of Φ. Hence, the eigenvalues of the solution of $\hat{\mathbf{\Psi}}_{LS}$ are the estimates of the L phase factors $e^{j\phi_\ell}$ [49].

2.4.1.2 2D-ESPRIT

When multiple parameter estimates are required, it is necessary for the the steering matrix $\mathbf{A}$ to become a function of multiple parameters, and the multiple estimates must be obtained simultaneously. The 2D-ESPRIT algorithm was used to jointly estimate the azimuth and elevation of impinging waves using a uniform rectangular array in [37]. The resulting double Vandermonde structure of the steering matrix allows a set of invariance equations to be formed. In the case of joint ToA/DoA estimation, the double Vandermonde structure does not arise naturally and must be induced before applying 2D-ESPRIT.

To facilitate ToA estimation, the received data is first transformed using a discrete Fourier transform, which maps the delays into phase shifts, followed by de-convolution

with the known pulse shape:

$$\begin{aligned}\mathbf{Y} &= \Big[\mathbf{a}(\psi_1)\ldots\mathbf{a}(\psi_L)\Big]\begin{bmatrix}\beta_1 & & 0\\ & \ddots & \\ 0 & & \beta_L\end{bmatrix}\begin{bmatrix}\mathbf{g}^T(\phi_1)\\ \vdots \\ \mathbf{g}^T(\phi_L)\end{bmatrix}+\mathbf{N}\\ &= \mathbf{A}(\boldsymbol{\psi})\mathbf{B}\mathbf{G}^T(\boldsymbol{\phi})+\mathbf{N}\end{aligned} \tag{2.18}$$

where $\mathbf{g}(\phi_\ell)=[1,\phi_\ell,\ldots,\phi_\ell^{N-1}]^T$, $\phi_\ell=e^{-j(2\pi P/N)\tau_\ell}$ and similarly, for a ULA with sensor array spacing d wavelengths $\mathbf{a}(\psi_\ell)=[1,\psi_\ell,\ldots,\psi_\ell^{M-1}]^T$, $\psi_\ell=e^{j2\pi d\theta_\ell}$.

The subsequent extension of 1D-ESPRIT to 2D-ESPRIT is straightforward. After applying multiple dimensional smoothing the data matrix, it exhibits a multiple rotational invariance structure, and the columns of the steering matrix $\mathbf{A}$ can be written as the Kronecker products of the steering vectors from each data dimension according to [50]:

$$\mathbf{a}(\mu_1,\ldots,\mu_R)=\mathbf{a}(\mu_R)\otimes\ldots\otimes\mathbf{a}(\mu_1) \tag{2.19}$$

where R denotes the number of data dimensions. In the 2D ToA/DoA case, one such approach is to apply the vec{} operator to $\mathbf{Y}$, specifically, the general relation $\text{vec}\{\mathbf{A}\text{diag}(\mathbf{b})\mathbf{C}\}=(\mathbf{C}^T\circ\mathbf{A})\mathbf{b}$, hence:

$$\text{vec}\{\mathbf{Y}\}=[\mathbf{G}(\boldsymbol{\phi})\circ\mathbf{A}(\boldsymbol{\psi})]\boldsymbol{\beta}+\mathbf{n}=\mathbf{U}(\boldsymbol{\phi},\boldsymbol{\psi})\boldsymbol{\beta}+\mathbf{n} \tag{2.20}$$

where $\mathbf{U}(\boldsymbol{\phi},\boldsymbol{\psi})=[\mathbf{u}(\phi_1,\psi_1),\ldots,\mathbf{u}(\phi_L,\psi_L)]$ is the combined *space-time response* matrix, with entries $\mathbf{u}(\phi_\ell,\psi_\ell)=\mathbf{g}(\phi_\ell)\otimes\mathbf{a}(\psi_\ell)$ and $\boldsymbol{\beta}=[\beta_1,\ldots,\beta_L]^T$. Consequently, it is not difficult to see that $\mathbf{U}(\boldsymbol{\psi},\boldsymbol{\phi})$ now has the following double Vandermonde structure:

$$\mathbf{U}(\boldsymbol{\psi},\boldsymbol{\phi})=\begin{bmatrix}1\cdot\mathbf{a}(\psi_1) & 1\cdot\mathbf{a}(\psi_2) & \ldots & 1\cdot\mathbf{a}(\psi_L)\\ \phi_1\cdot\mathbf{a}(\psi_1) & \phi_2\cdot\mathbf{a}(\psi_2) & \ldots & \phi_L\cdot\mathbf{a}(\psi_L)\\ \vdots & \vdots & \ldots & \vdots\\ \phi_1^{N-1}\cdot\mathbf{a}(\psi_1) & \phi_2^{N-1}\cdot\mathbf{a}(\psi_2) & \ldots & \phi_L^{N-1}\cdot\mathbf{a}(\psi_L)\end{bmatrix}=\begin{bmatrix}\mathbf{A}(\boldsymbol{\psi})\\ \mathbf{A}(\boldsymbol{\psi})\cdot\boldsymbol{\Phi}\\ \vdots\\ \mathbf{A}(\boldsymbol{\psi})\cdot\boldsymbol{\Phi}^{N-1}\end{bmatrix} \tag{2.21}$$

where the shift-invariance between rows of the array response in (2.12) remains unchanged in the sub-matrix $\mathbf{A}(\boldsymbol{\psi})$ part of $\mathbf{U}(\boldsymbol{\psi}, \boldsymbol{\phi})$ according to:

$$\boldsymbol{\Psi} = \begin{bmatrix} e^{j2\pi d\theta_1} & & \\ & \ddots & \\ & & e^{j2\pi d\theta_L} \end{bmatrix} \tag{2.22}$$

In addition, each M rows of $\mathbf{U}(\boldsymbol{\psi}, \boldsymbol{\phi})$ are related by the rotational operator $\boldsymbol{\Phi}$:

$$\boldsymbol{\Phi} = \begin{bmatrix} e^{-j(2\pi P/N)\tau_1} & & \\ & \ddots & \\ & & e^{-j(2\pi P/N)\tau_L} \end{bmatrix} \tag{2.23}$$

Therefore, the following set of invariance equations can be constructed,

$$\begin{aligned} \mathbf{J}_\psi \mathbf{U}\boldsymbol{\Psi} &= \mathbf{J}'_\psi \mathbf{U} \\ \mathbf{J}_\phi \mathbf{U}\boldsymbol{\Phi} &= \mathbf{J}'_\phi \mathbf{U} \end{aligned} \tag{2.24}$$

where $\mathbf{J}_\psi$ and $\mathbf{J}'_\psi$ selects every $M-1$ rows of $\mathbf{Y}$ starting from the first and second row respectively; while $\mathbf{J}_\phi$ and $\mathbf{J}'_\phi$ selects the first and last $(N-1)\times M$ rows of $\mathbf{Y}$ respectively.

Similar to (2.14), a matrix $\mathbf{E}$ (the signal subspace) containing a basis of the column span of $\mathbf{U}$ can be estimated by taking the left singular vectors corresponding to the largest L singular values of $\mathbf{Y}$. Without noise, $\mathbf{E}$ and $\mathbf{U}$ can be related through a transformation $\mathbf{T}$ by:

$$\mathbf{UT} = \mathbf{E} \tag{2.25}$$

where $\mathbf{T}$ is an arbitrary full rank matrix $\mathbf{T} \in \mathbb{C}^{L\times L}$. In addition, the following invariance equations between the signal eigenvectors of the sub-arrays holds:

$$\begin{cases} \mathbf{E}_\psi \triangleq \mathbf{J}_\psi \mathbf{E} = \mathbf{J}_\psi \mathbf{UT} \\ \mathbf{E}'_\psi \triangleq \mathbf{J}'_\psi \mathbf{E} = \mathbf{J}'_\psi \mathbf{UT} = \mathbf{J}_\psi \mathbf{U}\boldsymbol{\Psi}\mathbf{T} \end{cases} \qquad \begin{cases} \mathbf{E}_\phi \triangleq \mathbf{J}_\phi \mathbf{E} = \mathbf{J}_\phi \mathbf{UT} \\ \mathbf{E}'_\phi \triangleq \mathbf{J}'_\phi \mathbf{E} = \mathbf{J}'_\phi \mathbf{UT} = \mathbf{J}_\phi \mathbf{U}\boldsymbol{\Phi}\mathbf{T} \end{cases} \tag{2.26}$$

Then the estimation of the ψ_ℓ's and ϕ_ℓ's can be shown to be equivalent to the joint diagonalisation problem of [51]

$$\begin{aligned} \mathbf{E}_\psi^\dagger \mathbf{E}'_\psi &= \mathbf{T}^{-1}\mathbf{\Psi}\mathbf{T} \\ \mathbf{E}_\phi^\dagger \mathbf{E}'_\phi &= \mathbf{T}^{-1}\mathbf{\Phi}\mathbf{T} \end{aligned} \tag{2.27}$$

hence, the eigenvalues $\mathbf{E}_\psi^\dagger \mathbf{E}'_\psi$ of must be equal to the diagonal elements of $\mathbf{\Psi}$, and similarly for $\mathbf{\Phi}$. The connection of the θ_ℓ's and τ_ℓ's is provided by the fact that they have the same eigenvectors: the columns of $\mathbf{T}^{-1}$. Interested readers are referred to [50] and [39] for details of joint diagonlisation methods.

2.4.2 JADE

The JADE [23–25] algorithm is a method that exploits the stationarity of the angles and delays, as well as the independence of fading over many time-slots in a time slotted mobile system, by combining multiple estimates of the channel impulse response over many time slots.

Assuming samples of the estimate of the channel matrix $\mathbf{H}^{(n)}$ in (2.2) to have been already recovered from the received data, this can be performed using either blind channel estimation or employing training sequences; then applying the vec operator to $\hat{\mathbf{H}}^{(n)}$ yields:

$$\hat{\mathbf{h}}^{(n)} = \mathbf{U}(\boldsymbol{\theta}, \boldsymbol{\tau})\boldsymbol{\beta}^{(n)} + \mathbf{v}^{(n)}, \quad n = 1, \ldots, S \tag{2.28}$$

where $\hat{\mathbf{h}}^{(n)} = \text{vec}\{\hat{\mathbf{H}}^{(n)}\}$, and $\mathbf{v}^{(n)} = \text{vec}\{\mathbf{V}^{(n)}\}$; $\mathbf{V}^{(n)}$ is the channel estimation error during the nth symbol period.

Since the angle/delay parameters are quasi-stationary, then the space-time response matrix $\mathbf{U}(\boldsymbol{\theta}, \boldsymbol{\tau})$ can be assumed to be time invariant over the observation interval. The noisy channel estimates $\hat{\mathbf{h}}^{(n)}$ can be then combined as

$$\hat{\mathbf{H}} = \mathbf{U}(\boldsymbol{\theta}, \boldsymbol{\tau})\mathbf{B} + \mathbf{V} \tag{2.29}$$

where $\mathbf{B} = [\boldsymbol{\beta}^{(1)} \ldots \boldsymbol{\beta}^{(S)}]$ and similarly for $\mathbf{V}$.

The final step is to estimate the parameters of interest from the estimated channel $\hat{\mathbf{H}}$. Many of the well-known methods such as ML, subspace fitting, ESPRIT and MUSIC that have been developed for DoA estimation are applicable to the JADE problem [51].

2.4.2.1 JADE-ESPRIT

JADE-ESPRIT [51–53] is one of the two closed-forms solutions of the JADE method. As the name suggests, it is based on the 2D-ESPRIT algorithm but applied to the combined channel estimates in (2.29) instead. Since the main principles are almost identical to 2D-ESPRIT, its details are omitted here and the interested readers are referred to the mentioned references.

2.4.2.2 SI-JADE

Another way to introduce the shift-invariance structure required is to construct a block Hankel matrix [54] $\boldsymbol{\mathcal{H}}$ by stacking horizontal shifts of $\hat{\mathbf{H}}^{(n)}$. This is the basis of SI-JADE method [39]. Define $\mathbf{H}_{(i)}$ as the matrix formed by left shifting $\mathbf{H}$ by i columns. Then a block Hankel matrix $\boldsymbol{\mathcal{H}}$ can be formed by stacking m shifted versions of $\hat{\mathbf{H}}^{(n)}$ as $\boldsymbol{\mathcal{H}} = [\hat{\mathbf{H}}^{(n)}_{(1)}, \hat{\mathbf{H}}^{(n)}_{(2)}, \ldots, \hat{\mathbf{H}}^{(n)}_{(m)}]^T$:

$$\boldsymbol{\mathcal{H}} = \begin{bmatrix} \mathbf{a}(\psi_1)\beta_1[1\ \phi_1\ \phi_1^2 \ldots] & + \ \ldots \ + & \mathbf{a}(\psi_L)\beta_L[1\ \phi_1\ \phi_1^2 \ldots] \\ \mathbf{a}(\psi_1)\beta_1[\phi_1\ \phi_1^2\ \phi_1^3 \ldots] & + \ \ldots \ + & \mathbf{a}(\psi_L)\beta_L[\phi_L\ \phi_L^2\ \phi_L^3 \ldots] \\ \vdots & \vdots & \vdots \\ \mathbf{a}(\psi_1)\beta_1[\phi_1^{m-1}\ \phi_1^m\ \phi_1^{m+1} \ldots] & + \ \ldots \ + & \mathbf{a}(\psi_L)\beta_L[\phi_L^{m-1}\ \phi_L^m\ \phi_L^{m+1} \ldots] \end{bmatrix} \tag{2.30}$$

Consequently, the Hankel matrix then has the shift-invariance structure similar to that of (2.21):

$$\boldsymbol{\mathcal{H}} = \begin{bmatrix} \mathbf{ABG} \\ \mathbf{A\Phi BG} \\ \vdots \\ \mathbf{A\Phi}^{m-1}\mathbf{BG} \end{bmatrix} \tag{2.31}$$

The problem is then again reduced to one that can be solved by using a two-dimensional ESPRIT-like shift-invariance technique to separate and estimate the phase shifts. This

in turn is the joint diagonalisation problem of

$$\begin{aligned}(\mathbf{J}_\phi \boldsymbol{\mathcal{H}})^\dagger (\mathbf{J}'_\phi \boldsymbol{\mathcal{H}}) &= \mathbf{T}^{-1}\boldsymbol{\Phi}\mathbf{T} \\ (\mathbf{J}_\psi \boldsymbol{\mathcal{H}})^\dagger (\mathbf{J}'_\psi \boldsymbol{\mathcal{H}}) &= \mathbf{T}^{-1}\boldsymbol{\Psi}\mathbf{T}\end{aligned} \tag{2.32}$$

where the selection matrices $\mathbf{J}_\phi$ and $\mathbf{J}_\psi$ are similar to those employed in (2.24).

2.4.2.3 JADE-MUSIC

Another suboptimal approach is based on the MUSIC algorithm, which involves only a two dimensional search. Since the true space-time channel vector $\mathbf{u}(\theta, \tau)$ is orthogonal to the noise subspace $\mathbf{E}_n$, hence the temporal-spatial parameter estimates can be determined by the locations of the L largest peaks in the two dimensional MUSIC spectrum [23]:

$$\frac{\mathbf{u}^*\mathbf{u}}{\mathbf{u}^*\mathbf{E}_n\mathbf{E}_n^*\mathbf{u}} \tag{2.33}$$

where the explicit dependencies on θ_ℓ and τ_ℓ is dropped for convenience. JADE-MUSIC was applied to estimate jointly the direction of arrival/departure and delay of a MIMO communication system in [55].

2.5 The Maximum Likelihood Approach

Perhaps the most well-known and frequently-used model-based approach for estimation in signal processing is the maximum likelihood technique. ML based methods tend to be more robust and accurate. However, they have had limited application, due to the high computational load of the multivariate non-linear maximisation problem involved.

Given that the received signal $\mathbf{y}(t)$ is sampled regularly N times at $\{t_1, \ldots, t_N\}$, and assuming the following conditions hold:

- The array manifold $\{\mathbf{a}(\theta) : \theta \in \Theta\}$, where Θ is the field of view, is known.
- The noise samples are $\{\boldsymbol{\eta}(t_k)\}$ are i.i.d Gaussian random vectors with zero mean and covariance matrix $\sigma^2\mathbf{I}_L$, where σ^2 is an unknown scalar.

- The received data vector $\mathbf{y}(t_k)$ is also an i.i.d Gaussian random vector with mean $\mathbf{A}(\boldsymbol{\theta})\mathbf{B}\mathbf{G}^T(\boldsymbol{\tau})$ and covariance matrix $\sigma^2\mathbf{I}_L$.

it follows that the p.d.f of the received data vector $\mathbf{y}(t_k)$ is given by:

$$p\big(\mathbf{y}(t_1),\ldots,\mathbf{y}(t_N)\big) = \prod_{k=1}^{N} \frac{1}{|\pi\sigma^2\mathbf{I}|} e^{\|\mathbf{e}(t_k)\|^2/\sigma^2} \tag{2.34}$$

where

$$\mathbf{e}(t_k) = \mathbf{y}(t_k) - \sum_{\ell=1}^{L} \mathbf{a}(\theta_\ell)\beta_\ell g(t_k - \tau_\ell) \tag{2.35}$$

Hence, the MLE for the complete set of unknown $[\hat{\sigma}^2, \hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\beta}}]$ is given by:

$$[\hat{\sigma}^2, \hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\beta}}] = \underset{\sigma^2,\boldsymbol{\theta},\boldsymbol{\tau},\boldsymbol{\beta}}{\arg\max} \left\{ -NM\log\sigma^2 - \frac{1}{\sigma^2}\sum_{k=1}^{N} \|\mathbf{e}(t_k)\|^2 \right\} \tag{2.36}$$

Separate MLE of the nuisance parameters σ^2 can be formed and substituted back into (2.36) to reduce the total number parameters involved in the optimisation, therefore, maximising (2.36) w.r.t σ^2 yields

$$\hat{\sigma}^2 = \frac{1}{MN}\sum_{k=1}^{N} \|\mathbf{e}(t_k)\|^2 \tag{2.37}$$

and substituting back into (2.36) and re-writing in matrix notation gives:

$$[\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\beta}}] = \underset{\boldsymbol{\theta},\boldsymbol{\tau},\boldsymbol{\beta}}{\arg\min} \left\{ \|\mathbf{Y} - \mathbf{A}\mathbf{B}\mathbf{G}\|^2 \right\} \tag{2.38}$$

Furthermore, applying the general relation $\text{vec}\{\mathbf{A}\text{diag}(\mathbf{b})\mathbf{C}\} = (\mathbf{C}^T \circ \mathbf{A})\mathbf{b}$ and maximising w.r.t $\boldsymbol{\beta}$ yields

$$\hat{\boldsymbol{\beta}} = (\mathbf{D}^H\mathbf{D})^{-1}\mathbf{D}^H\text{vec}\{\mathbf{Y}\} \tag{2.39}$$

where $\mathbf{D} \triangleq \mathbf{G}^T \circ \mathbf{A}$; which when substituted back into (2.38) yields the MLE for $[\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\tau}}]$ as:

$$[\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\tau}}] = \underset{\boldsymbol{\theta},\boldsymbol{\tau}}{\arg\max} \|\mathbf{P}_{\mathbf{D}}^{\perp}\text{vec}\{\mathbf{Y}\}\|^2 \tag{2.40}$$

where $\mathbf{P}_{\mathbf{D}}^{\perp}$ is the orthogonal projection on the columns of $\mathbf{D}$:

$$\mathbf{P}_{\mathbf{D}}^{\perp} = \mathbf{D}(\mathbf{D}^H\mathbf{D})^{-1}\mathbf{D}^H \tag{2.41}$$

A useful geometric interpretation of the MLE in (2.40) is that it finds the L steering vectors that form a signal subspace that is as close as possible to vec$\{\mathbf{Y}\}$. Closeness is measured by the magnitude of the projection of vec$\{\mathbf{Y}\}$ onto the estimated signal subspace [54]. However, this involves a $2L$-dimensional optimisation over both $\boldsymbol{\theta}$ and $\boldsymbol{\tau}$ and, hence, is computationally prohibitive even for moderate values of L. Although the MLE in (2.40) eliminates the nuisance parameter β, it is more computationally demanding than (2.38) when the size of $\mathbf{D}$ is large; hence, it is sometimes preferable to use the MLE in (2.38) instead.

A similar MLE for joint channel parameter estimation in a multipath channel was derived within [56] using the Fourier transformed data (where the delays are transformed into phase shifts). The MLE of $[\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\beta}}]$ can be shown to be

$$[\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\beta}}] = \underset{\boldsymbol{\theta},\boldsymbol{\tau},\boldsymbol{\beta}}{\arg\min} \sum_{k=1}^{N} \left\| \mathbf{y}(\omega_k) - \sum_{\ell=1}^{L} \mathbf{a}(\theta_\ell)\beta_\ell g(\omega_k) e^{-j\omega_k \tau_\ell} \right\|^2 \tag{2.42}$$

where $\mathbf{y}(\omega_k)$ and $g(\omega_k)$ are the DFTs of $\mathbf{y}(t_k)$ and $g(t_k)$, the samples of the received signal $\mathbf{y}(t)$ and delayed copies of the known narrowband signal $g(t)$ respectively.

2.5.1 EM

The Expectation Maximisation (EM) algorithm [28, 29], formulated by Dempster *et al*, is a iterative method for solving ML estimation problems in cases where there is a many-to-one mapping from an underlying distribution to the distribution governing the observation [57], i.e. direct access to the data necessary to estimate the parameters is impossible.

By decomposing the observed (incomplete) data into its unobservable (complete) signal components and then estimate the parameters of each signal components separately, then the complex maximisation in (2.38) is de-coupled to L separate 3D ML maximisations. Hence, the complexity of the algorithm is essentially unaffected by the number of signal components. Larger number of components can be accommodated by increasing the number of ML processors in parallel. At each iteration, the current parameter estimates are used to decompose the observed data better and thus improve the next parameter estimates.

Define the contribution of the ℓth wave to the M baseband signals at output of the antenna array as

$$\begin{aligned} \mathbf{s}_\ell(t;\boldsymbol{\omega}_\ell) &\triangleq [s_{1,\ell}(t;\boldsymbol{\omega}_\ell),\ldots,s_{M,\ell}(t;\boldsymbol{\omega}_\ell)]^T \\ &= \beta_\ell \mathbf{a}(\theta_\ell) g(t-\tau_\ell) \end{aligned} \tag{2.43}$$

where the vector $\boldsymbol{\omega}_\ell \triangleq [\beta_\ell, \theta_\ell, \tau_\ell]$ contains the parameters of the ℓth wave. Then the received signal vector $\mathbf{y}(t) \triangleq [y_1(t),\ldots,y_M(t)]^T$ at the output of the antenna array reads:

$$\mathbf{y}(t) = \sum_{\ell=1}^{L} \mathbf{s}_\ell(t;\boldsymbol{\omega}_\ell) + \sqrt{\frac{N_o}{2}}\boldsymbol{\eta}(t) \tag{2.44}$$

In this expression, $\boldsymbol{\eta}(t) \triangleq [\eta_1(t),\ldots,\eta_M(t)]^T$ denotes a standard M-dimensional vector valued complex AWGN with power spectral density $N_o/2$; N_o is a positive constant.

For the problem of estimation of signal parameters at an antenna array, the set of L vectors $\mathbf{s}_\ell(t;\boldsymbol{\omega}_\ell)$ in (2.43), corrupted by a part of the additive noise, constitute a natural set of complete-unobservable data, i.e.

$$\mathbf{x}_\ell(t) \triangleq \mathbf{s}_\ell(t;\boldsymbol{\omega}_\ell) + \sqrt{\frac{N_o\gamma_\ell}{2}}\boldsymbol{\eta}_\ell(t) \tag{2.45}$$

where the non-negative parameters γ_ℓ satisfy $\sum_\ell \gamma_\ell = 1$ so that the the noise vector $\boldsymbol{\eta}(t)$ can be decomposed into the set $\{\sqrt{\gamma_1}\boldsymbol{\eta}_1(t),\ldots,\sqrt{\gamma_L}\boldsymbol{\eta}_L(t)\}$. While the observed signal vector $\mathbf{y}(t)$ at the output of the antenna arrays in (2.44) forms the incomplete data space. It is related to the complete data according to $\mathbf{y}(t) = \sum_\ell^L \mathbf{x}_\ell(t)$.

Since $\mathbf{x}_1(t),\ldots,\mathbf{x}_L(t)$ are independent, the components $\mathbf{x}_{\ell'}(t)$ are irrelevant for the estimation of $\boldsymbol{\omega}_\ell$, where $\ell' \neq \ell$. However $\mathbf{x}_\ell(t)$ is not actually observable, it has to be estimated first. This is done based on the observation $\mathbf{y}(t)$ of the incomplete data and a previous estimate $\hat{\boldsymbol{\omega}}'$ of $\boldsymbol{\omega}$:

$$\hat{\mathbf{x}}_\ell(t) = \mathbf{s}_\ell(t;\hat{\boldsymbol{\omega}}'_\ell) + \gamma_\ell\left[\mathbf{y}(t) - \sum_{\ell'=1}^{L}\mathbf{s}_\ell(t;\hat{\boldsymbol{\omega}}'_{\ell'})\right] \tag{2.46}$$

where the first term is the contribution of the ℓth wave assuming $\boldsymbol{\omega}_\ell = \boldsymbol{\omega}'_\ell$, and the expression within the brackets is an estimate of the noise $\sqrt{\frac{N_o}{2}}\boldsymbol{\eta}(t)$ based on the hypothesis that $\forall\ell : \boldsymbol{\omega}_\ell = \boldsymbol{\omega}'_\ell$. The wave parameters $\boldsymbol{\omega}_\ell$ can then be further estimated by

computing the MLE based on the observation $\hat{\mathbf{x}}_\ell(t)$:

$$\begin{aligned}\widehat{(\tau_\ell,\theta_\ell)}_{\mathrm{ML}}\big(\mathbf{x}_\ell(t)\big) &= \underset{[\tau,\theta]}{\arg\max}\left\{\left|z\Big(\tau,\theta;\mathbf{x}_\ell(t)\Big)\right|\right\}\\ \big(\hat{\beta}_\ell\big)_{\mathrm{ML}}\big(\mathbf{x}_\ell(t)\big) &= \frac{1}{MT_gP_g}z\Big(\widehat{(\tau_\ell,\theta_\ell)}_{\mathrm{ML}}\big(\mathbf{x}_\ell(t)\big);\mathbf{x}_\ell(t)\Big)\end{aligned} \tag{2.47}$$

where

$$z\Big(\tau,\theta;\mathbf{x}_\ell(t)\Big) \triangleq \int_{T_o} g^*(t'-\tau)\mathbf{a}^H(\theta)\mathbf{x}_\ell(t')\,dt' \tag{2.48}$$

where T_o is the observation interval of the received data $\mathbf{y}(t)$.

Steps (2.46) and (2.47) are known as the Expectation (E-step) and Maximisation (M-step) steps of the EM algorithm. The algorithm iterates between estimating the likelihood of the complete data using the incomplete data and the current parameter estimates (E-step) and maximising the estimated log-likelihood function to obtain the updated parameter estimates (M-step). Under mild regularity conditions, the iterations of the EM algorithm converge to a stationary point of the observed log-likelihood function, where at each iteration, the likelihood of the estimated parameters is always increased [58]. This is known the monoticity property of the EM algorithm.

2.5.2 SAGE

The EM algorithm, which updates all of parameters simultaneously, has two main drawbacks: 1) slow convergence 2) difficult maximisation step due to coupling when smoothness penalties are used [30]. The SAGE algorithm, which is a two-fold extension of the EM algorithm, overcomes these problems by updating the parameters sequentially while alternating between several small hidden data spaces. The algorithm replaces the high-dimensional optimisation process of (2.47) in the EM algorithm by several separate low-dimensional maximisation steps, while still maintaining the basic monotonicity of the EM algorithm, i.e. the likelihood of the estimated parameters is always increased at each iteration [58].

Each step of the SAGE algorithm consists of estimating a subset of $\hat{\boldsymbol{\omega}}_\ell = [\hat{\tau}_\ell, \hat{\theta}_\ell, \hat{\beta}_\ell]$, while keeping the estimates of the other components fixed. The coordinate-wise update procedure used to obtain the new estimate $\hat{\boldsymbol{\omega}}''_\ell$ of the wave ℓ given the current estimate

$\hat{\boldsymbol{\omega}}'_\ell$ is

$$\begin{aligned}\hat{\tau}''_\ell &= \arg\max_{\tau}\left\{\left|z\left(\tau,\hat{\theta}'_\ell;\hat{\mathbf{x}}_\ell(t;\hat{\boldsymbol{\omega}}')\right)\right|\right\}\\ \hat{\theta}''_\ell &= \arg\max_{\theta}\left\{\left|z\left(\hat{\tau}''_\ell,\theta;\hat{\mathbf{x}}_\ell(t;\hat{\boldsymbol{\omega}}')\right)\right|\right\}\\ \hat{\beta}''_\ell &= \frac{1}{MTP}z\left(\hat{\tau}''_\ell,\hat{\theta}''_\ell;\hat{\mathbf{x}}_\ell(t;\hat{\boldsymbol{\omega}}')\right)\end{aligned} \tag{2.49}$$

where

$$\hat{\mathbf{x}}_\ell(t;\hat{\boldsymbol{\omega}}') = \mathbf{y}(t) - \sum_{\ell'=1,\ell'\neq\ell}^{L}\mathbf{s}_\ell(t;\hat{\boldsymbol{\omega}}'_{\ell'}) \tag{2.50}$$

Carrying out this updating procedure for all L components defines one iteration cycle of the SAGE algorithm. Equations (2.50) and (2.49) are the E-step and M-step respectively of the SAGE algorithm. The E-step calculates an estimate $\hat{\mathbf{x}}_\ell(t;\hat{\boldsymbol{\omega}}')$ of the noise corrupted version of $\mathbf{s}_\ell(t,\boldsymbol{\omega}_\ell)$, the contribution of the ℓth path to the received data, by subtracting the estimated contribution of all waves, except the ℓth one, from the received signal. This process is known as Parallel Interference Cancellation (PIC).

While the EM method can start with any arbitrary initial guess and still converge relatively quickly, the convergence rate of SAGE is highly dependent on the choice of initialisation values. Typically, the initialisation steps are based on Successive Interference Cancellation (SIC), where the parameter vector of wave ℓ is obtained by removing estimates of the interference caused by previously estimated waves from the received signal $\mathbf{y}(t)$,

$$\hat{\mathbf{x}}_\ell(t;\hat{\boldsymbol{\omega}}) = \mathbf{y}(t) - \sum_{\ell'=1}^{\ell-1}\mathbf{s}_\ell(t;\hat{\boldsymbol{\omega}}'_{\ell'}) \tag{2.51}$$

Various enhancements have also been suggested to ensure best possible initialisation, such as incorporating super-resolution techniques such as MUSIC to obtain accurate initial estimates of the delays when closely-spaced multipath are present.

It is worth noting that although the standard SAGE initialisation steps require the use of SIC, no constraint is placed on the SAGE E-step in (2.50), and both SIC and PIC can be used interchangeably. Extensive simulations in [59, 60] have shown that PIC based estimation is biased and causes instability in the algorithm if the estimated L is less than the true number of dominant paths in the channel. Later on, we will demonstrate

that neither of these approaches are well suited to certain scenarios. To summaries, the key differences between SIC and PIC procedures are as follows:

- **SIC:** Estimated complete data are subject to interference from the un-removed $(\ell+1)$th as well as residue components from previously estimated paths that are not totally removed.
- **PIC:** Estimated complete data are subject to only the interference from the residue component of the other $L-1$ paths, as long as the model-order estimation is accurate.

The SAGE algorithm has been extensively tested in synthetic and real environments, with results reported in [61, 62] and [63] for the MIMO case. Results show rapid convergence of RMSEs close to the corresponding CRLB values for synthetic values. The larger the separation between two waves in terms of $\Delta\tau$, $\Delta\theta$ is, the faster the convergence. The high resolution ability of the scheme is also apparent, with it capable of separating waves as soon as one parameter differ by more than roughly half the intrinsic resolution of the measurement equipment. It is also noticeable that the separation ability in delays is better than that in angle.

2.5.3 JADE-ML

Assuming that the estimation noise $\mathbf{V}$ in (2.29) is zero mean Gaussian and spatially white, and that the entries of the complex fading matrix $\mathbf{B}$ can be modelled as unknown deterministic quantities, the deterministic ML approach of the JADE method yields the following minimisation problem [23]:

$$\min_{\boldsymbol{\theta},\boldsymbol{\tau},\mathbf{B}} ||\hat{\mathbf{H}} - \mathbf{U}(\boldsymbol{\theta},\boldsymbol{\tau})\mathbf{B}||_F^2 \tag{2.52}$$

where the optimisation on $\mathbf{B}$ can be separated, and the problem is reduced to:

$$[\hat{\boldsymbol{\theta}},\hat{\boldsymbol{\tau}}] = \arg\max_{\boldsymbol{\theta},\boldsymbol{\tau}} \mathrm{tr}\left(\mathbf{P}_{\mathbf{U}}\hat{\mathbf{R}}_{\hat{\mathbf{H}}}\right) \tag{2.53}$$

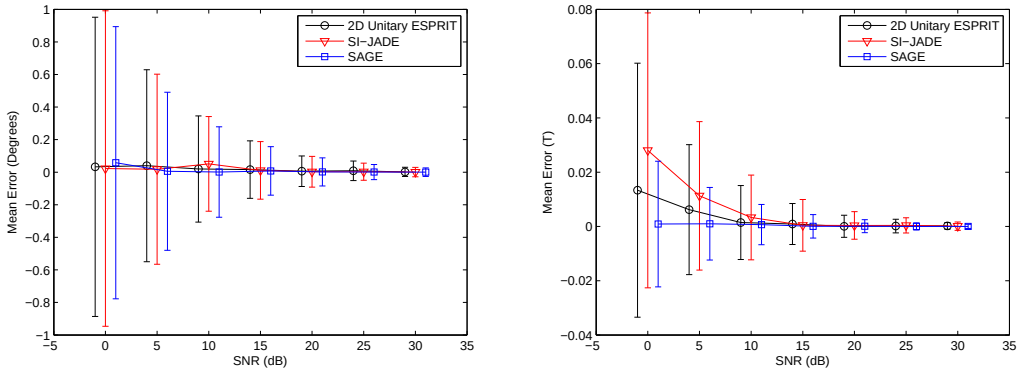
where $\mathbf{P}_{\mathbf{U}} = \mathbf{U}(\mathbf{U}^H\mathbf{U})^{-1}\mathbf{U}^H$ and $\hat{\mathbf{R}}_{\hat{\mathbf{H}}} = \hat{\mathbf{H}}\hat{\mathbf{H}}^H/M$.

2.6 Key Aspects and Discussions

In this section, the performance of the 2-D Unitary ESPRIT, SI-JADE [64] and SAGE algorithms are compared in a multipath environment. The main goal of this study is to compare their estimation accuracy of the Line of Sight (LoS) component as well as the temporal/spatial resolution. In addition, key aspects of these algorithms, such as their robustness to model errors, in particular, the effects of the under-estimation of the number of paths as well as computational complexity will be addressed.

2.6.1 Special Case - Estimation of One Wave

The estimation accuracy of channel parameters can be effected by a number of factors: the number of multipath, sampling rate, multipath resolution, multipath amplitudes and the number of antenna elements at the receiver etc. We will look at the performance under multipath conditions later on, here, we consider a simple ideal channel with just one path. Our intention is to compare the algorithms' optimum performance with just the LoS component present, and to establish the performance gain one can get from the use of more antenna elements in ToA/DoA estimation.



(a) DoA (b) ToA

FIGURE 2.3: Single-wave estimation error and standard deviation as a function of SNR.

In the first experiment, we estimated the ToA/DoA of the sole (the LoS component) path in the channel at different SNR values. The results should provide a good comparison of the optimum performances of these algorithms, without any interference from other multipath. Figure 2.3 shows the mean error and the standard deviation of the

errors as a function of SNR for the LoS component. All three algorithms demonstrate comparable accuracy for DoA estimation as shown in Figure 2.3 (a). However, the superior performance of SAGE for ToA estimation is clearly seen in Figure 2.3 (b), more so at lower SNR (0dB to 5dB), where its mean error appears almost to be "unaffected" by the change in SNR as compared to the others.

Next, we repeated the same experiment while increasing the number of antenna elements at the receiver, i.e. M=4, 8, and 16. It is well known from array processing literature that the angular resolution for any DoA estimation algorithm is greatly dependent on the number of antenna elements, i.e. overall length of array. In this case, we are interested to see what levels of improvement in estimation accuracy could be obtained from this for both DoA and ToA. Comparisons are also made against the Cramer-Rao Lower Bound (CRLB) for DoA estimation.

The covariance matrix $\mathbf{C}_{\hat{\boldsymbol{\omega}}}$ of any unbiased estimator $\hat{\boldsymbol{\omega}} \triangleq [\hat{\beta}, \hat{\theta}, \hat{\tau}]$ satisfies [27]:

$$\mathbf{C}_{\hat{\boldsymbol{\omega}}} = E\Big[(\hat{\boldsymbol{\omega}} - \boldsymbol{\omega})^T(\hat{\boldsymbol{\omega}} - \boldsymbol{\omega})\Big] \geq \mathbf{F}^{-1}(\boldsymbol{\omega}) \tag{2.54}$$

where the $n \times n$ positive definite real matrix $\mathbf{F}(\boldsymbol{\omega})$ is the Fisher information matrix of $\boldsymbol{\omega}$, its elements are given as:

$$[\mathbf{F}(\boldsymbol{\omega})]_{ij} = -E\left[\frac{\partial^2 \ln p(\mathbf{y};\boldsymbol{\theta})}{\partial \boldsymbol{\omega}_i \partial \boldsymbol{\omega}_j}\right] \tag{2.55}$$

where the derivatives are evaluated at the true value of $\boldsymbol{\omega}$ and the expectation is taken with respect to $p(\mathbf{y};\boldsymbol{\theta})$.

The diagonal element $\left[\mathbf{F}^{-1}(\boldsymbol{\omega})\right]_{ii}$ of $\mathbf{F}^{-1}(\boldsymbol{\omega})$ is referred to as the CRLB of $\hat{\boldsymbol{\omega}}_i$, such that if $\hat{\boldsymbol{\omega}}_i$ is unbiased, then its variance is lower bounded as:

$$\begin{aligned} \mathrm{var}(\hat{\boldsymbol{\omega}}_i) &= \left[\mathbf{C}_{\hat{\boldsymbol{\omega}}}\right]_{ii} \\ &\geq \left[\mathbf{F}^{-1}(\boldsymbol{\omega})\right]_{ii} \triangleq \mathrm{CRLB}(\hat{\boldsymbol{\omega}}_i) \end{aligned} \tag{2.56}$$

an estimator which is unbiased and attains the CRLB for all $\boldsymbol{\omega}$ is said to be *efficient* in that it efficiently uses the data. Even if such an estimator does not exist, the CRLB is helpful in determining the Minimum Variance Unbiased (MVU) estimator, which as

the name suggest, its variance is uniformly less than the variance of all other unbiased estimators.

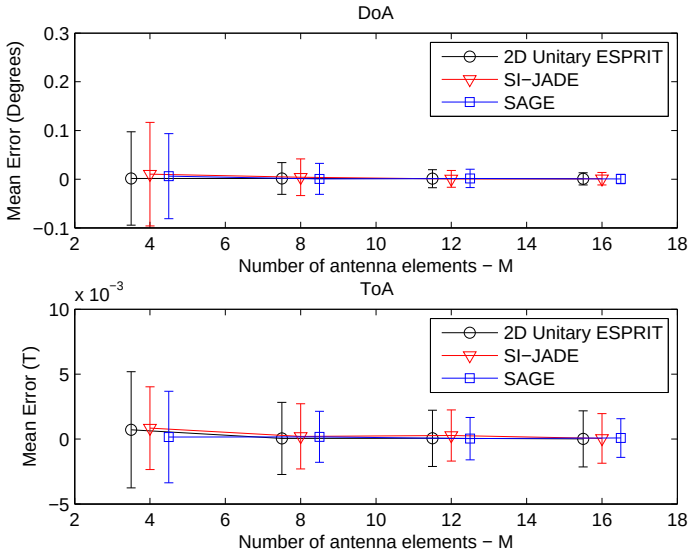
Figure 2.4 (a) shows the mean error and standard deviation of the errors versus the number of antenna elements. From the top figure in Figure 2.4 (a), we can see a significant reduction in standard deviation values, as expected, in DoA estimation as the number of antenna elements increases. From the CRLB for DoA estimation in [65] and [31], it can be shown that the variance of DoA estimation for all three algorithms is proportional to $1/M^3$. Figure 2.4 (b) shows the ratio of this improvement more clearly, where the RMSE of the DoA estimates are plotted against SNR for every doubling of the number of antenna elements. From which we can identify a constant reduction in RMSE each time the number of antenna elements is doubled. This follows the same pattern for the CRLB of SI-JADE also plotted in the same figure.

On the other hand, we also observe a slight improvement for ToA estimates as shown in the bottom figure of Figure 2.4 (a), in fact the RMSE improves roughly as $1/M$. Hence for a ToA-based mobile positioning system, the cost of increasing the number of antenna elements will probably outweigh the benefit it brings.

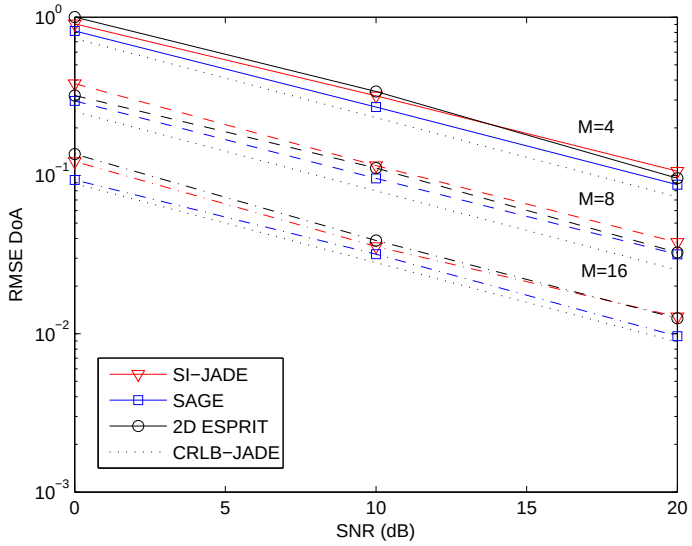
2.6.2 Multipath Resolution and Identification

To study the resolution performance (ability to distinguish closely spaced paths in both temporal/spatial domain) of joint channel parameter estimation algorithms, it is necessary to consider the joint effect of closely spaced paths in both dimensions; as suppose to the delay or angular resolution of two paths separately.

Assume a simple 2 path channel with equal amplitude and initial equal DoA/ToA values, i.e. $\{\theta_1 = 10^\circ, \tau_1 = 1T\}$ and $\{\theta_2 = 10^\circ, \tau_2 = 1T\}$, we estimate the DoA and ToA values of both paths as the path separation (*both* temporal and spatial) is increased: $\{\theta_2 = 10^\circ \rightarrow 25^\circ, \tau_2 = 1T \rightarrow 1.6T\}$. Fig 2.5, 2.6 and 2.7 shows the distributions of the two paths for JADE, ESPRIT and SAGE respectively. From the results for JADE and ESPRIT, we can see that an minimum separation of $0.6T$ in ToA is required to achieve reasonably accurate detection of both paths when there is zero separation in DoA; and 15° separation in DoA is required when both paths have the same ToA. In fact, all cases on the main anti-diagonal exhibits reasonable detection of both paths.



(a) DoA and ToA estimation error and standard deviation values as a function of the number of antenna elements - M; SNR=20dB



(b) RMSE of DoA estimation for M=4,8,16

FIGURE 2.4: Single-wave estimation as a function of number of antenna elements. Each cluster of lines in (b) shows the RMSEs of the DoA estimates at one value of M, hence a constant reduction of RMSE can be observed every time M is doubled.

In the SAGE result, resolution ability in both domains is comparably weaker. The resolution of conventional techniques like SAGE for delay and angle estimation is limited by the intrinsic resolution of the measurement equipment. For cross-correlation based ToA estimation, this is approximately limited by the inverse of the bandwidth of the transmitted signal. While the resolution for beamforming based DoA estimation is limited by the beamwidth of the measuring array. For a transmitted signal with symbol

period T and M-element ULA at the receiver, these figures are $\Delta\tau = T$ and $\Delta\theta = 360°/(\pi M)$ [31]. Hence the 15° separation is clearly not enough when the two paths have the same delay.

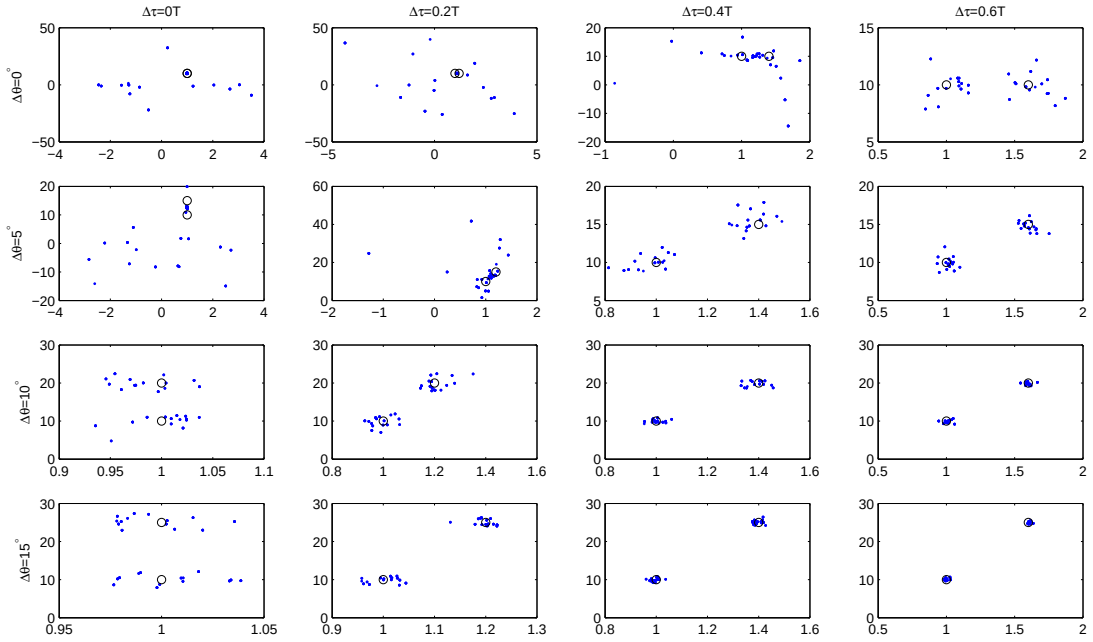


FIGURE 2.5: Distribution of JADE estimates as paths separation is increased. Starting at top left corner, where the parameters of the two paths are identical: $\{\theta_1 = \theta_2 = 10°, \tau_1 = \tau_2 = 1T\}$; τ_2 is increased by $0.2T$ each time from left to right and θ_2 is increased by 5° each time from top to bottom. The parameters of path one remains unchanged. SNR=15dB.

Next, we compare the algorithms' performance in the multipath environment where the channel contain 10 paths within a delay spread of $10T$ to represent a relatively strong multipath scenario. We assess the algorithms' ability to identify the multipath. Figure 2.8 shows the distributions of the path estimates from 50 simulations at 15dB. We can observe similar performances in terms of identifying each of the multipath, such that the 4 closest paths were easily visible to all three algorithms, while the remaining paths were estimated with varying degrees of success. It is clear from Figure 2.8(a) and (b) that JADE and ESPRIT's estimates for these furtherest paths appear to be some what random, in addition, a number of paths has been missed completely. On the other hand, SAGE has produced some partially correct estimates for these paths (e.g. correct delay, wrong angle), which is clearly evident for the 2 furtherest paths in Figure 2.8(c). In fact, the large variance which implies the randomness in JADE and ESPRIT's estimates can

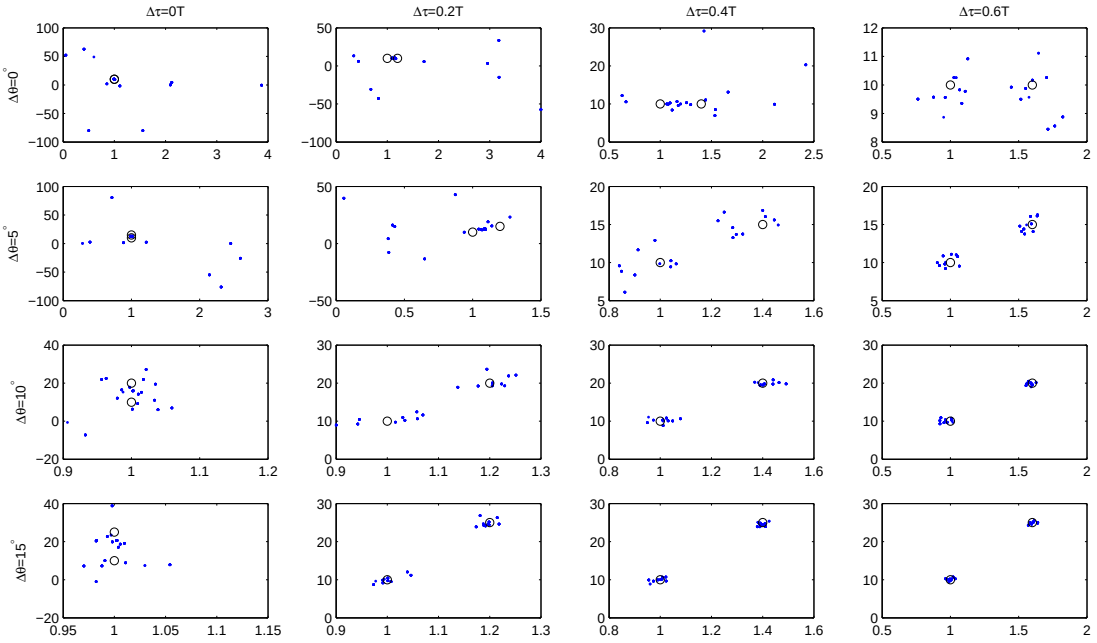


FIGURE 2.6: Distribution of ESPRIT estimates as paths separation is increased. Starting at top left corner, where the parameters of the two paths are identical: $\{\theta_1 = \theta_2 = 10°, \tau_1 = \tau_2 = 1T\}$; τ_2 is increased by $0.2T$ each time from left to right and θ_2 is increased by 5° each time from top to bottom. The parameters of path one remains unchanged. SNR=15dB.

have a significant impact on the positioning accuracies if these data were used in the positioning of a mobile device.

A principle reason for some of SAGE's partially correctly path estimates is due to the sequential nature in which it processes the multipath; such that the interference cancellation stage of the algorithm can have a negative impact on the estimation results. In the case of a channel containing a strong LoS component, if the channel parameters for the LoS component are not estimated very accurately, then it won't be removed "cleanly" from the overall data. Subsequently, the remaining contribution from the stronger components effects the estimation accuracy of the weaker (normally further) paths.

2.6.3 Multipath Estimation Accuracy

In this section, we are concerned with the estimation of the LoS component under multipath conditions. Figure 2.9 shows the estimation accuracy of the LoS component against SNR in a multipath channel, which consists of 5 well-separated multipath with relatively

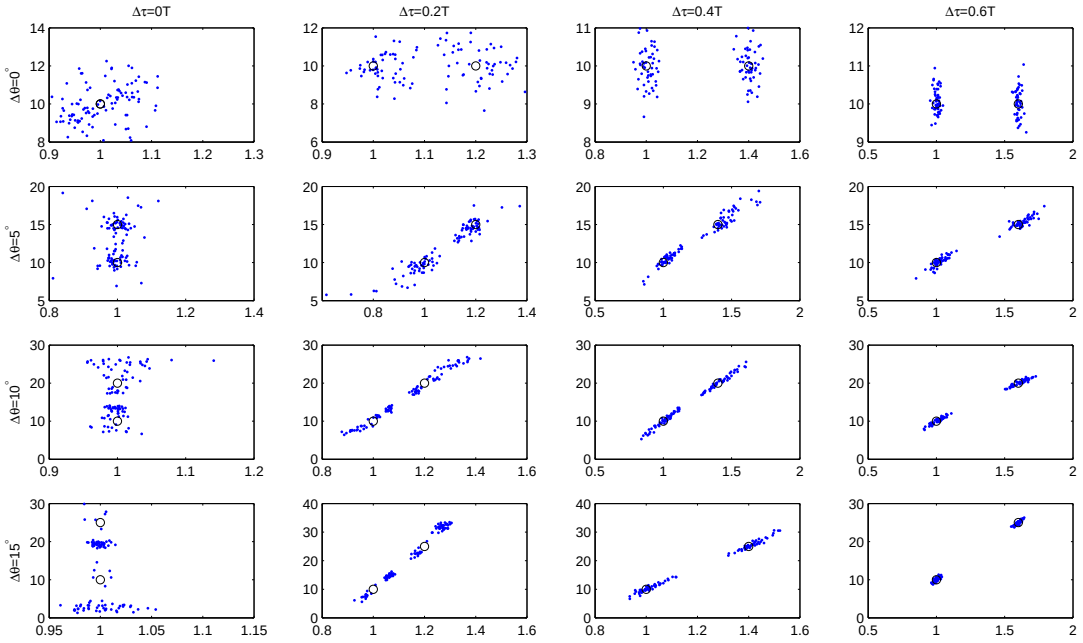
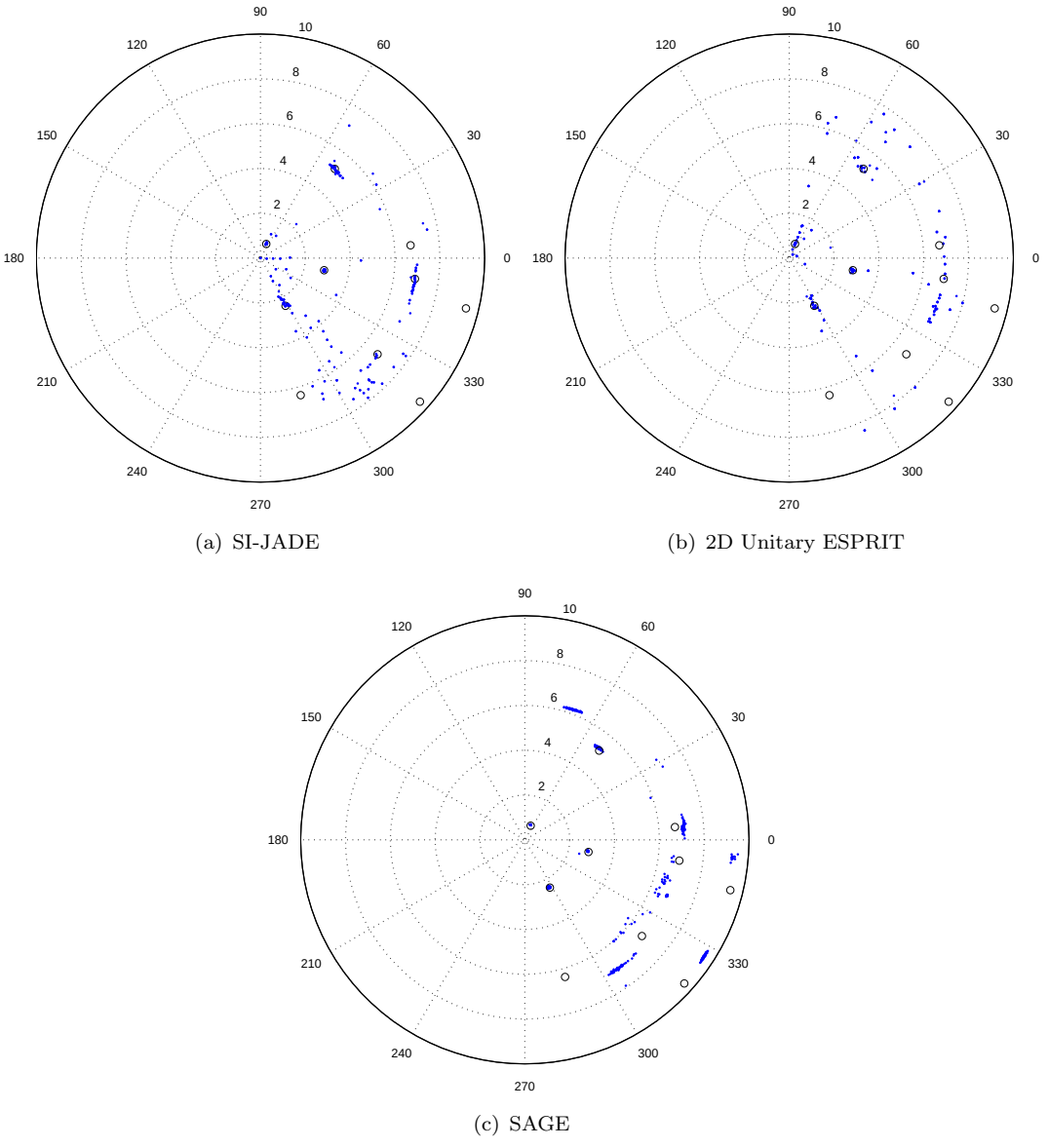


FIGURE 2.7: Distribution of SAGE estimates as paths separation is increased. Starting at top left corner, where the parameters of the two paths are identical: $\{\theta_1 = \theta_2 = 10°, \tau_1 = \tau_2 = 1T\}$; τ_2 is increased by $0.2T$ each time from left to right and θ_2 is increased by 5° each time from top to bottom. The parameters of path one remains unchanged. SNR=15dB.

high amplitudes. It should be obvious that the relative performances of the algorithms is different from that shown in Figure 2.3 under the one path scenario, i.e. performance of ESPRIT/JADE has somewhat deteriorated in comparison with SAGE. It was evident in Figure 2.8 that the estimation accuracy will degrade under multipath conditions, while in the case for ESPRIT and JADE, widely distributed erroneous estimates are produced, this is particularly obvious at lower SNR conditions. Hence the result in Figure 2.9 are a direct result of these incorrect estimates. In keeping with realistic conditions, we cannot assume any prior knowledge of the MS postilion, hence, the LoS component estimates are calculated by always using the path with the shortest delay value at each simulation, this is done regardless of whether another path is much closer to the actual value. In this case, the ToA mean error (ME) and standard deviation (SD) values for SAGE are approximately 6 and 4 times smaller respectively at 5dB. Even at 10dB, its ME and SD values are 4 and 2 times smaller. While only in the high SNR region (>15dB), the performance of all three methods are comparable.



(a) SI-JADE

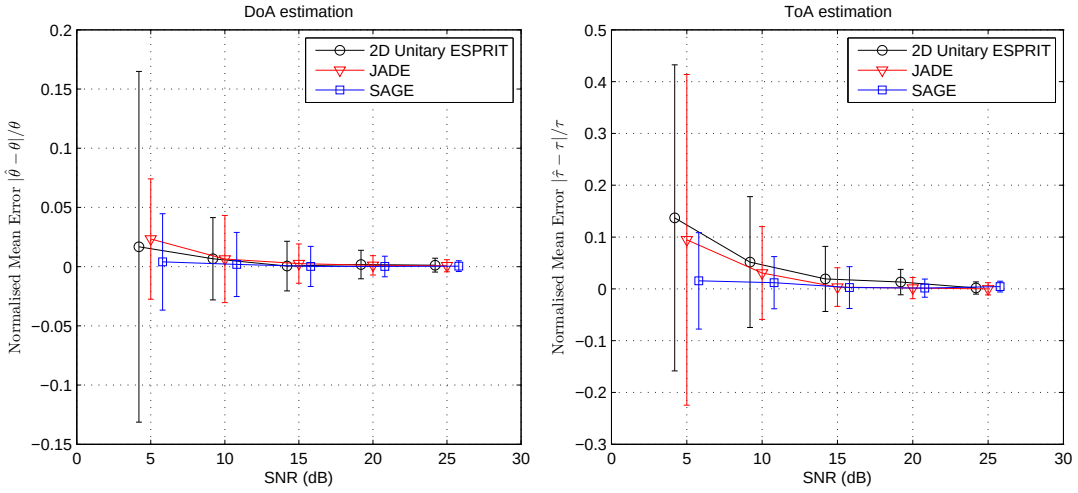
(b) 2D Unitary ESPRIT

(c) SAGE

FIGURE 2.8: Distribution of path estimates after 50 simulations at 15dB. The circles indicate actual multipath locations.

2.6.4 Locationing Accuracy

As mentioned previously, a major advantage of the joint estimation of ToA/DoA is that the location of a mobile device can be estimated based on data acquired from a single base station. In this section, we examen the potential positioning accuracy derived from the ToA/DoA estimates of the LoS component. Using ToA/DoA estimates from 50 independent trials, we can obtain the position estimate of the MS from the intersection of the linear and circular Lops formed by the ToA and DoA measured at the reference BS respectively, see Figure 1.2. The Cartesian position estimate of the MS is defined as



(a) Direction of arrival (b) Time of arrival

FIGURE 2.9: Normalised mean errors in DoA and ToA estimation of the LoS component. Entire length of vertical bar represents ±1 standard deviation. Results are averaged over 200 independent runs.

[66]:

$$\hat{x} = x_{\text{BS}} + c\hat{\tau}_{\text{LoS}} \cos(\hat{\theta}_{\text{LoS}})$$
$$\hat{y} = y_{\text{BS}} + c\hat{\tau}_{\text{LoS}} \sin(\hat{\theta}_{\text{LoS}}) \tag{2.57}$$

where, $c \approx 3 \times 10^8$ is the speed of light. Without loss of generality, we can assume the coordinate system is centred at the reference BS, i.e. $x_{\text{BS}} = y_{\text{BS}} = 0$.

Table 2.1 shows the combined standard deviations in range estimates[3]: $\sigma_{\hat{x}} + \sigma_{\hat{y}}$, for both a weak LoS channel and strong LoS channel under different SNR conditions. We define a weak LoS channel as when the LoS component is not the strongest multipath component, this is typically observed in some urban and indoor environments where a small proportion of the LoS component propagates through the obstruction and is marginally detectable at the receiver. To simulate this environment, the amplitude of the LoS component was set to approximately 1/3 of that of the strongest path. In both cases, the MS is placed approximately 200m away from the serving BS. Employing the same method of choosing the LoS component as in 2.6.3, we can observe that the inconsistencies of the ESPRIT and JADE estimates under non-ideal conditions has a significant effect on overall positioning performance. In the weak LoS channel, the ESPRIT estimates is

[3]since these estimators are unbiased, this is the same as combined RMSE

unable to give "useful" position indication under any SNR conditions, while JADE is only marginally better. The accuracy of SAGE's estimates are good when the SNR is high (> 10dB) in the weak LoS channel. In the strong LoS scenario, the performance of all three estimators are comparable under high SNR conditions. However, the accuracy of SAGE's estimates are good even in very low SNR conditions, where at 0dB, its combined standard deviation is nearly 6 times less than the others.

		0dB	5dB	10dB	15dB	20dB
Weak LoS	ESPRIT	$305.4m$	$130.9m$	$101.5m$	$72.7m$	$67.6m$
	JADE	$262.4m$	$120.3m$	$132.5m$	$64.6m$	$10.1m$
	SAGE	$269.7m$	$122.9m$	$21.0m$	$12.2m$	$6.9m$
Strong LoS	ESPRIT	$191.4m$	$116.1m$	$49.7m$	$13.0m$	$4.8m$
	JADE	$213.8m$	$41.2m$	$19.3m$	$9.3m$	$4.3m$
	SAGE	$34.9m$	$15.0m$	$10.7m$	$6.4m$	$3.6m$

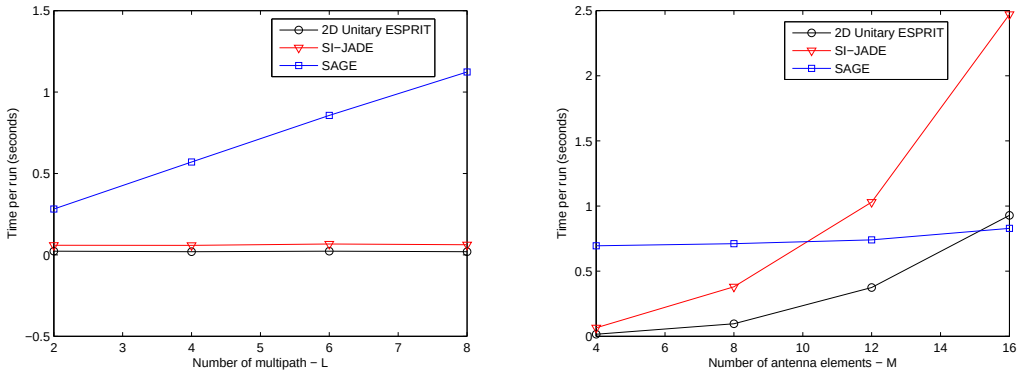
TABLE 2.1: Combined standard deviation ($\sigma_{\hat{x}} + \sigma_{\hat{y}}$) in range estimates using joint ToA/DoA estimates from ESPRIT, JADE and SAGE. $L = 6$ for both strong and weak LoS channels. Symbol period $T = 1\mu s$. Sampling interval $= T/4$.

2.6.5 Computational Complexity and Real Time Processing

In this section, our aim is to analyse each algorithm's complexity with respect to the number of multipath and the number of antenna elements (i.e. input data size). Here, we measure computational complexity in terms of CPU time per Monte Carlo run of the algorithms. It is important to stress that the magnitude, or even the order, of the CPU execution time of each algorithm is not the emphasis of this section. Our focus is on the relationship of the complexity to the number of paths and number of antenna elements, since the magnitude/order of the complexity for each method is highly dependent on code implementation, which is outside the scope of this study.

Figure 2.10 (a) shows the CPU time against the number of multipath. Unsurprisingly, a linear relationship is shown between the execution time and path number for SAGE, since each SAGE cycle only estimate the parameters of one path. The eigen-decomposition approaches estimate all the paths through a joint diagonalisation approach, hence increasing the number of paths has no effect on the CPU time. One might expect that increasing the size of the channel matrix will have a significant effect on all the algorithms. However, this is not the case as shown in Figure 2.10 (b), where the CPU time is plotted as a function of antenna elements. This is because the most complicated step

of SAGE, e.g. the cost function in (2.48), involves only matrix-vector multiplications as supposed to matrix-matrix multiplications within that of the other two methods. In addition, the actual optimisation of the cost function (which is independent of the matrix size) far outweighs the numerical calculation of the integral, hence an increase in matrix size contributes little to the total CPU time.



(a) Complexity as a function of number of multipath; Number of antenna elements = 4

(b) Complexity as a function of number of antenna elements; Number of multipath = 5

FIGURE 2.10: Computational complexity comparison of the algorithms measured in CPU time per Monte Carlo run. The focus here is on the relationship of the complexity to the number of paths and number of antenna elements, not the magnitude or even order.

Although typically much faster than ML based methods, the step of obtaining the signal/noise subspace ((2.25) and (2.33)) represents a barrier that prevents the on-line realisation of a significant proportion of eigen-decomposition based methods for large matrix sample sizes. This step is usually achieved by the eigen-decomposition of a covariance matrix or the singular value decomposition (SVD) of a data matrix directly. An estimate of covariance matrix is obtained using the sample covariance matrix $\hat{\mathbf{R}} = \mathbf{Y}^H\mathbf{Y}/N$, where $\mathbf{Y} = [\mathbf{y}(1), \ldots, \mathbf{y}(N)]^T$ is the $M \times N$ snapshot of the received data. It is not difficult to see that the singular values and right singular vectors of $\mathbf{Y}$ are the eigenvalues of $\hat{\mathbf{R}}$ respectively. Therefore, the signal subspace can be directly estimated from the $\mathbf{Y}$ using the SVD, but nevertheless, both approaches have cubic complexity, i.e. $O(n^3)$, for an $n \times n$ matrix, doubling input data results in 8 times more operations, such as matrix multiplications. An eigen-decomposition requires $O(M^3)$ multiplications for an $M \times M$ matrix plus $O((NM)^2)$ multiplications for forming the covariance matrix. An SVD needs at least $O((NM)^2 + M^3)$ multiplications for an $N \times M$ $(N > M)$ matrix [67].

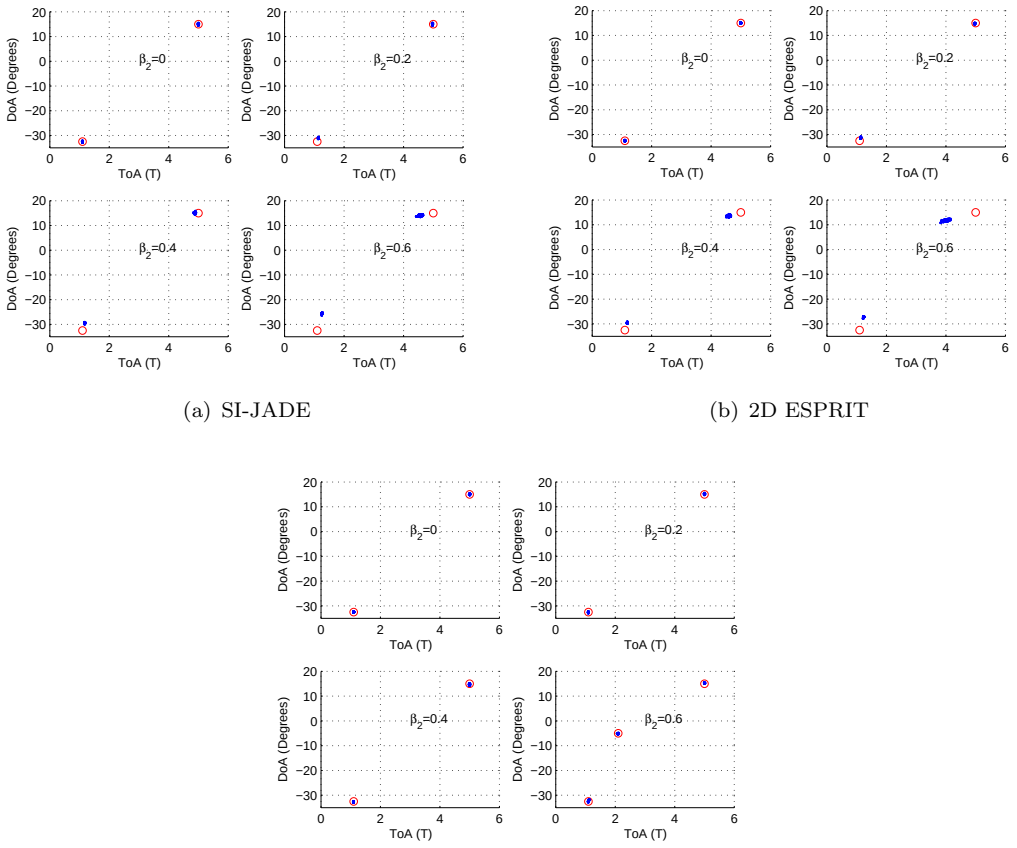
The complexity of JADE-MUSIC is a one-time $O((MN)^3)$ for the eigen-decomposition of the covariance matrix $\mathbf{R}$, plus a 2-D search with $O((MN)^2)$ per gridpoint. SI-JADE requires only $O(mMN^3)$ operations (m is the stacking parameter), resulting from the rank reduction of the channel matrix. JADE-ESPRIT requires $O((MN)^3)$ operations for the most computationally-expensive step of obtaining a basis for the column span of the space-time manifold matrix $\mathbf{U}$ in (2.25) [24].

None of the above is computationally efficient to update as new data arrive, and hence they are not well suited to real-time applications where the required subspace is to be tracked. Interested readers may refer to [68] and [69] for a thorough overview of most of the adaptive algorithms for subspace tracking.

The EM and SAGE algorithms in their original forms are more appropriate for off-line processing. In order to eliminate the delay in decision-making, reduce storage, and increase the computational efficiency in real-time applications, it is desirable, and often necessary, to process the received data in a recursive manner [70]. Many recursion-based EM algorithms are available. A good example is given in [70] employing stochastic approximation, while that in [71] is specific for DoA estimation. A recently proposed recursive SAGE-inspired approach is given in [72]. The term "SAGE-inspired" is used because the proposed algorithm updates all parameters simultaneously.

2.6.6 Robustness to Model Errors

An important aspect of these estimation algorithms is their robustness to model errors and noise. Here, the robustness of the algorithms to the (mild) under-estimation of the number of channel path is investigated. We examine the effects of utilising incorrect L knowledge on the estimation accuracy of each methods. The true setup parameters are: $L = 3$, with the angle, delay and amplitude: $\boldsymbol{\theta} = [-32.5; -5; 15]^\circ$, $\boldsymbol{\tau} = [1.1; 2.1; 5]T$, $\boldsymbol{\beta} = [1; \beta_2; 0.6]$ where $\beta_2 = 0 \rightarrow 0.6$. In the experiment, the amplitude of the first and third paths is held constant, while that of the second varies from 0 to 0.6. Hence when $\beta_2 = 0$, there are really only two paths. During the simulation, we specify the estimated number of path to be 2. Hence a model error occurs when β_2 becomes non-zero.



(a) SI-JADE

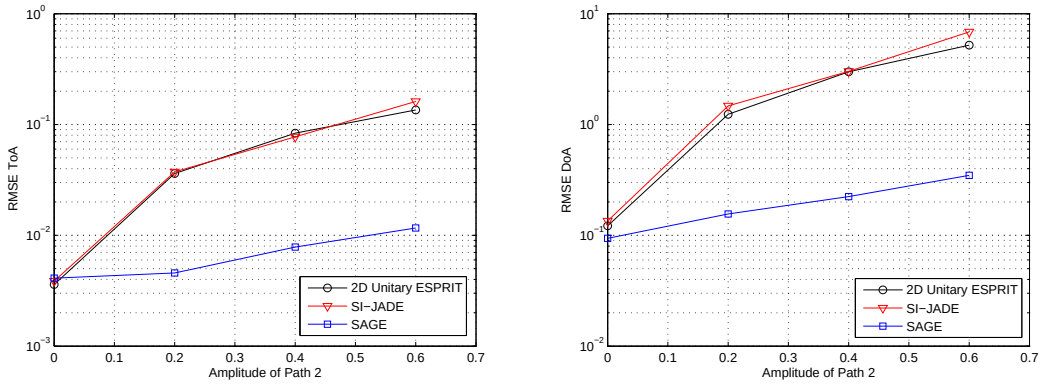
(b) 2D ESPRIT

(c) SAGE

FIGURE 2.11: Distribution of path estimates when L is underestimated. Circles indicate actual multipath locations.

Figure 2.11 shows the distributions of the path estimates for each method at increasing amplitude levels of the second path. We can clearly observe that in the case of SI-JADE and 2D-ESPRIT, there is a strong degradation in estimation of both paths as the amplitude of the second becomes significant, in which the deterioration in the third path more obvious since it is the weaker path. The observed behaviour of SAGE is somewhat different, it detects the specified number of the most significant paths, and simply ignores the rest. Hence the estimation of both paths remains relatively unaffected when β_2 is small; as soon as β_2 becomes comparable to that of the third path, then both of them are equally likely to be picked. The order in which the paths are estimated is determined during initialisation. The first initialisation cycle estimates the delay of the path that gives best cost-function evaluation, and it is then followed by the initialisation of the angle and amplitude. This path is then removed by using interference cancellation. The next initialisation cycle picks out the delay of the path that gives the new optimum, and

so on. This is a desirable behaviour for a real system such that a slight underestimate of the actual number of paths should not prevent the most significant paths from being identified correctly.



(a) ToA (b) DoA

FIGURE 2.12: RMSE in DoA and ToA of the first path when L is underestimated

Fig. 2.12 shows how the RMSE in the estimated ToA and DoA of the first path varies as the amplitude of the second path increases. Notice that SI-JADE and 2D Unitary ESPRIT are badly affected as soon as the power of the second path increases even slightly, where we can see a 10-fold increase in RMSE when the amplitude of the second path reaches 0.2; for both angle and delay estimates. However, only a slight degradation is obvious in the results for SAGE.

Hence, under-estimation of the path number is much worse for SI-JADE and 2D Unitary ESPRIT as it relies on the angle-delay subspace. This in turn depends greatly on accurate knowledge of L. So if there are more paths than the dimensions of the computed angle-delay subspace, the detection of certain multi-path components becomes difficult.

Another important source of model error is in the array model. The performance of DoA detection and estimation algorithms is ultimately limited by the noise in the array measurements and by errors in the array model [73]. In many cases, the array response is not exactly known. Changes in weather, the surrounding environment, and antenna location, may cause the response of the array to be significantly different from the time it was last calibrated. When unknown factors are considered, the array model should be parametrised not only by the DoA and ToA parameters but also some additional calibration-dependent parameters. These may include sensor element positions, gain

and phase offsets, mutual coupling, and element diversity. Given such a model, auto-calibration techniques [73] are often used to estimate the unknown (nuisance) model parameters simultaneously with the signal parameters.

2.7 Summary

We have presented a critical review of the joint parameter estimation techniques, ESPRIT, JADE and SAGE. Through extensive simulations in MATLAB, we have demonstrated that: 1) for all three algorithms, a significant improvement in DoA estimation and mild improvement in ToA estimation can be achieved as the number of antenna element increases. 2) SAGE's superior ToA estimation accuracy in a lower SNR environment and that it outperforms the other two algorithms in terms of ToA/DoA estimation variance and robustness to the channel model errors. 3) SAGE's computation complexity exhibits a linear relationship with the number of channel paths (while ESPRIT and JADE are independent) while being unaffected by the number of antenna elements.

We have also demonstrated certain limitations of these algorithms in the multipath environments, where the impact of multipath was clearly seen even when channel conditions are good. This remains as a challenging issue for parameter estimation algorithms applied in realistic channel environments. Although the SAGE algorithm exhibits strong potential in accurate channel parameter estimations, especially for positioning applications where the accuracy of the LoS component is of essence. However, a potential problem occurs where the effects of imprecise estimates are cascaded to other path estimates through the use of interference cancellation.

Chapter 3

Improving the SAGE Algorithm with Adaptive Interference Cancellation

3.1 Introduction

The Space Alternating Generalised Expectation maximisation (SAGE) [30, 31] algorithm is a well-accepted joint parameter estimation technique that has been used extensively in channel parameter estimation and Multiple-Input-Multiple-Output (MIMO) channel sounding scenarios [31, 36, 63]. In our previous review of joint channel parameter estimation techniques, the SAGE algorithm was shown to be the best performing among other popular algorithms, including JADE and 2D-ESPRIT [74]. However, we discovered that the performance of SAGE could be severely degraded by the error accumulated from the Interference Cancellation (IC) stage as each path in the channel is estimated sequentially. In this chapter, we propose the use of adaptive interference cancellation to improve SAGE's performance. We focus on applying SAGE to mobile positioning applications using Time of Arrival (ToA) and Direction of Arrival (DoA) techniques, in which the Line of Sight (LoS) signal component is of most importance. We study the particular case when the LoS component in the channel is weaker than other components where the standard SAGE algorithm has been shown to perform poorly.

The SAGE algorithm estimates a radio channel's parameters iteratively until pre-defined criteria are satisfied, on the assumption that the channel has more than one signal path. All the paths are processed sequentially during each iteration and, in order to estimate the parameters of an individual path, the interference from some/all the other paths in the channel is removed by the IC module. The path's parameters are then estimated from the "interference cleared" data. The contributions within the received data for this path is then reconstructed using the estimated parameters and the IC process removes it from the overall data before estimating the next path in the channel.

In principle, traditional IC can be performed either in *serial* (SIC) or *parallel* (PIC). Serial cancellation removes the interference from the first path up to and excluding the current path, hence the first path to be processed sees all of the interference from the remaining $L-1$ paths. Whereas each path downstream sees less and less as the cancellation progresses. On the other hand, parallel processing simultaneously removes the reconstructed interference produced by all other paths. The standard SAGE algorithm utilises PIC for the bulk of the processing as it is potentially quicker to converge, see (2.50). Whereas SIC is used in initialisation to obtain initial estimates of each path one after another, see (2.51).

Both SIC and PIC attempt to remove the reconstructed interferences completely, this is sometimes known as *brute force* cancellation. In a strong multipath environment, this can cause large errors in subsequent iterations since the interference removed can be an inaccurate estimate of the interfering path (or paths), and errors occurring at the IC stage for the first few paths will be accumulated and influence the process of the paths later on. Our previous work has demonstrated that, for weaker multipath components, the performance of SAGE degrades because of accumulated IC errors from stronger signal paths.

Recent work [75] has shown that the performance of a similar IC stage in a Multiple User-CDMA (MU-CDMA) system can be improved by performing partial interference cancellation at each iteration. Partial cancellation in MU-CDMA involves multiplying estimates of each user's symbol by a factor less than unity before cancellation. Divsalar *et al.* [76] proposed a partial cancellation approach by introducing a weight in each stage to determine the amount of cancellation. In early cancellation stages, the reliability of symbol detection is worse than at later stages, so the weights were increased with each

iteration. Xue *et al.* [77] later proposed an adaptive version where a set of weights is used to control the individual interferences caused by each user.

In this chapter, we extend the idea in [77] to the SAGE algorithm. Instead of attempting to remove each interfering path completely, we take away only a portion w of each interfering path. Since removing incorrectly estimates will add interference rather than remove it, it is desirable to cancel a fraction of the estimated interference if a channel parameter estimate is thought to be unreliable. We define a cost function that takes the weights w into consideration to minimise the squared Euclidean distance between the received signal and the weighted sum of the estimates of all paths' signal during the observation interval with respect to the weights. The normalised Least Mean Square (LMS) [78] algorithm is then used to adjust the fraction of the signal from each path that is removed in the next iteration. Monte-Carlo simulations in MATLAB are carried out to compare the performance of the standard SAGE algorithm with the proposed adaptive version employing adaptive IC, specifically in a weak-LoS multipath environment where the standard SAGE is shown to perform poorly.

3.2 The Least Mean Square Algorithm

The LMS algorithm is by far the most widely used algorithm in adaptive filtering. The main features that attracted the use of the LMS algorithm are low computational complexity, proof of convergence in stationary environment and unbiased convergence in the mean to the Weiner solution [79].

For a length $p+1$ finite impulse response filter with the filter input $\mathbf{x}(n) = [x(n), x(n-1), \ldots, x(n-p)]^T$, the filter output at the nth time sample is expressed as:

$$y(n) = \mathbf{w}^T(n)\mathbf{x}(n) \tag{3.1}$$

where $\mathbf{w}^T(n) = [w_0(n), w_1(n), \ldots, w_p(n)]^T$ is the time varying vector of filter coefficients. The output error is formed as the difference between the filter output and a training sequence $d(n)$, i.e. $e(n) = d(n) - y(n)$.

The LMS algorithm is a type of the steepest decent method that iteratively searches for the optimum set of filter weights $\mathbf{w}$ that minimises the mean square error $E\{|e(n)|^2\}$ by

recursively applying corrections according to [78]:

$$\begin{aligned} \mathbf{w}_{n+1} &= \mathbf{w}_n - \mu \nabla E\{|e(n)|^2\} \\ &= \mathbf{w}_n + \mu E\{e(n)\mathbf{x}^*(n)\} \end{aligned} \tag{3.2}$$

where $\nabla E\{|e(n)|^2\}$ denote the partial derivatives of $E\{|e(n)|^2\}$ w.r.t the coefficients $\mathbf{w}(n)$. While the step size μ controls the size of the incremental correction applied to the weight vector from one iteration cycle to the next.

The LMS algorithm significantly simplifies the steepest decent approach in (3.2) by replacing the expectation with an instantaneous sample mean. In this case, the weight vector update equation assumes a particularly simple form:

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \mu e(n)\mathbf{x}^*(n) \tag{3.3}$$

3.3 Adaptive Interference Cancellation

We can define an optimal cost function in terms of the Euclidean distance between the received signal $\mathbf{y}(t)$ and the weighted sum of all the estimates of all paths' signals, i.e.

$$\epsilon = \int_0^{T_o} \left| \mathbf{y}(t) - \sum_{\ell=1}^{L} w_\ell \mathbf{s}_\ell(t; \boldsymbol{\omega}_\ell) \right|^2 dt \tag{3.4}$$

where $\mathbf{s}_\ell(t; \boldsymbol{\omega}_\ell)$ is the contribution of the ℓth path to the baseband signals at the output of the antenna array as defined in (2.43) and $w_\ell \in [0, 1]$ is the weight for the ℓth path. We try to minimise ϵ with a set of optimal weights $\mathbf{w} = [w_1, \ldots, w_L]$. Neither the conventional brute force IC where $\forall \ell : w_\ell = 1$, nor the partial IC in [75] where $w_\ell = w_{\ell'}$ is the optimal solution to (3.4). However, for a suboptimal solution, a modified cost function can be formed in the sense of LS or MSE. We will demonstrate that the corresponding problem can be solved in a iterative manner through the LMS algorithm, and since the SAGE algorithm is iterative in nature too, the merging of one into the other is relatively straightforward.

Since the LMS algorithm is based on the MSE criteria, the cost function given in (3.4) is modified as follows:

$$\min_{\mathbf{w}^{(k)}} E\left[\left|\mathbf{y}(t)-\hat{\mathbf{y}}^{(k)}(t)\right|^2\right] = \min_{\mathbf{w}^{(k)}} E\left[\left|\mathbf{e}^{(k)}(t)\right|^2\right] \tag{3.5}$$

where $\mathbf{w}^{(k)} = [w_1^{(k)}, w_2^{(k)}, \ldots, w_L^{(k)}]^T$ is the weighting vector at the kth step. While $\hat{\mathbf{y}}^{(k)}(t)$ is the estimate of the received signal at the kth step, which is defined as:

$$\hat{\mathbf{y}}^{(k)}(t) = \sum_{\ell=1}^{L} w_\ell^{(k)} \mathbf{s}_\ell(t; \hat{\boldsymbol{\omega}}_\ell^{(k)}) \tag{3.6}$$

where the estimate of the contribution from the ℓth path during the kth stage $\mathbf{s}_\ell(t; \hat{\boldsymbol{\omega}}_\ell^{(k)})$ is reconstructed using its parameter estimates at stage k:

$$\mathbf{s}_\ell(t; \hat{\boldsymbol{\omega}}_\ell^{(k)}) = \mathbf{a}(\hat{\theta}_\ell^{(k)})\hat{\beta}_\ell^{(k)} g(t-\hat{\tau}_\ell^{(k)}) \tag{3.7}$$

At this stage, it is important to point out that unlike conventional LMS filtering, where the input sequence at each stage are L consecutive samples of some known source, the input data in this case are the set of L estimated contributions from each path, i.e. $\left[\mathbf{s}_1(t; \hat{\boldsymbol{\omega}}_1^{(k)}), \ldots, \mathbf{s}_L(t; \hat{\boldsymbol{\omega}}_L^{(k)})\right]$. Moreover, each estimated contribution $\mathbf{s}_\ell(t; \hat{\boldsymbol{\omega}}_\ell^{(k)})$ is a function of time as well as the iteration number k. Therefore, instead of using individual time samples, we utilise the entire observation of $\mathbf{s}_\ell(t; \hat{\boldsymbol{\omega}}_\ell^{(k)})$ at each iteration (i.e. data samples from the entire observation period T_o; and similarly for $\mathbf{y}^{(k)}(t)$ and $\mathbf{e}^{(k)}(t)$); while subsequent LMS iterations utilises a re-estimated observation of $\mathbf{s}_\ell(t; \hat{\boldsymbol{\omega}}_\ell^{(k+1)})$ based on the new estimate $\hat{\boldsymbol{\omega}}_\ell^{(k+1)}$.

With this in mind, the weighting vector can be adjusted via a normalised LMS algorithm as follows in matrix form:

$$w_\ell^{(k+1)} = w_\ell^{(k)} + \mu \times \text{sum}\left\{\frac{\left(\mathbf{S}_\ell^{(\mathrm{k})}\right)^* \diamond \mathbf{E}^{(\mathrm{k})}}{||\mathbf{S}_\ell^{(\mathrm{k})}||_{\mathrm{F}}}\right\} \qquad \text{for } \ell \in \{1, \ldots, \mathrm{L}\} \tag{3.8}$$

here the gradient estimate is the sum of the elements of the Hadamard product between the matrices $\left(\mathbf{S}_\ell^{(k)}\right)^*$ and $\mathbf{E}^{(k)}$; where $\mathbf{S}_\ell^{(k)}$ is a $M \times N$ matrix of the samples of the contribution of the ℓth path duration the kth iteration, samples are taken with a sampling period of T/P, where P is the oversampling factor. Similarly, $\mathbf{E}^{(k)}$ is a $M \times N$ matrix

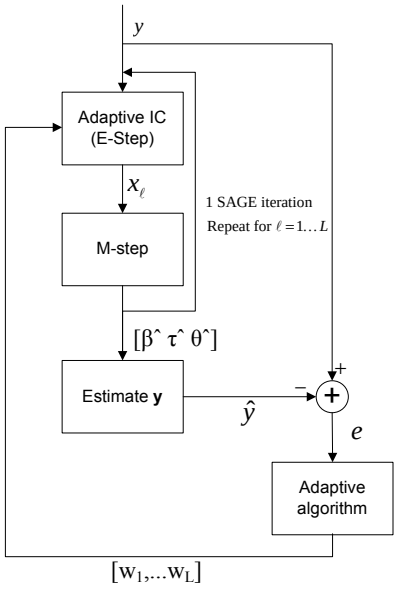


FIGURE 3.1: Schematic of SAGE algorithm with adaptive interference cancellation

of the samples of the error at the kth iteration in (3.5). Detailed derivation of the LMS gradient estimate in (3.8) can be found in Appendix A.

The step size μ plays an important role in the LMS algorithm. The normalised LMS simplifies the selection of μ by employing a time-varying step size. Subsequently, μ needs to satisfy $0 < \mu < 2$ to ensure convergence in this case. Another factor that affects the convergence rate of the LMS algorithm is its initial state. Since the LMS weights w_ℓ determines the amount of cancellation for the ℓth path, therefore it should also be an indication of the reliability of the path's parameter estimates. A simple initialisation for $w_\ell^{(0)}$ would be initial path amplitude $\hat{\beta}_\ell^{(0)}$, this is based on the assumption that parameter estimates of the stronger paths are more reliable than weaker ones.

Finally, the brute force IC step in (2.50) can be replaced with a adaptive cancellation approach:

$$\hat{\mathbf{x}}_\ell(t; \hat{\boldsymbol{\omega}}_\ell{}^{(k)}) = \mathbf{y}(t) - \sum_{\ell'=1, \ell' \neq \ell}^{L} w_{\ell'}^{(k)} \mathbf{s}_\ell(t; \hat{\boldsymbol{\omega}}_{\ell'}^{(k)}) \tag{3.9}$$

Figure 3.1 shows a schematic diagram of the proposed SAGE algorithm with adaptive IC. During one iteration, a fraction of the signal of each path is removed from the received data $\mathbf{y}(t)$ before estimating the channel parameters. The newly-estimated channel parameters are then used to obtain an estimate of $\mathbf{y}(t)$, i.e. the sum of contributions

from all paths. The error in the estimate of $\mathbf{y}(t)$ is then fed into the adaptive algorithm to produce updated weights for the next iteration.

3.3.1 M-Step Optimisation Interval

A second modification we make to the standard SAGE algorithm changes the optimisation interval for which the M-steps in (2.49) are evaluated. For each channel parameter, the standard SAGE algorithm obtains the current estimate by evaluating the maximisations in (2.49) over a interval that is centred at the estimate from the previous step, e.g. the ToA estimate for the ℓth path at the kth step $\hat{\tau}_\ell^{(k)}$ is obtained from:

$$\hat{\tau}_\ell^{(k)} = \underset{\tau \in \left[\hat{\tau}_\ell^{(k-1)} - \phi/k, \hat{\tau}_\ell^{(k-1)} + \phi/k\right]}{\arg\max} \{|z(\bullet)|\} \tag{3.10}$$

where ϕ is a pre-defined constant. Estimation of $\hat{\theta}_\ell^{(k)}$ is identical.

Notice as k increases, the search interval is narrowed which helps fine tuning as well as eventual convergence. However, if the difference between the initialised value $\hat{\tau}_\ell^{(0)}$ and its actual value is greater than search window in the very first iteration, i.e.

$$\left|\tau_\ell - \hat{\tau}_\ell^{(0)}\right| > \phi/1 \tag{3.11}$$

then it is not difficult to see that unless maximum progresses are made in the direction of the optimum within the first few SAGE iterations, then it is very difficult for the consecutive estimates to converge to the true value; since the value of ϕ/k becomes quite small after just a few iterations. It is possible to make the value of ϕ large enough to accommodate cases of poor initialisations. However, widening this search window too much is effectively disregarding the initialised data. This is another indication why the performance of the SAGE algorithm is highly dependent on its initial state and why the weaker paths are not well estimated.

Therefore, in difficult channel conditions, such as low SNR or high number of multipath, where the initial channel estimates are unreliable, a much better approach is to replace the narrowing search window of $\pm\phi/k$ in (3.10) with a smaller but non-varying window of $\pm\phi'$ instead. Provided ϕ' is not chosen too large, the loss in convergence speed is minimal. More importantly, it allows consecutive iterations certain freedom in the

optimisation interval such that in some case, convergence to actual estimates is possible even if initial values are poor.

3.4 Simulations

In this section, Monte-Carlo simulations were carried out in MATLAB to study the performance of the proposed adaptive SAGE algorithm; comparisons are made against the standard SAGE algorithm employing traditional brute force IC in a weak LoS multipath environment.

3.4.1 Weakness of Brute Force Interference Cancellation

We begin with a simplified 3 paths environment to highlight the problems that incorrect cancellation can cause in the standard SAGE algorithm. We specify the amplitudes of the paths according to $\{\beta_1 = 0.1\beta_3$ and $\beta_2 = 0.2\beta_3\}$, i.e. the amplitude of the LoS component β_1 is only 10% of that of the strongest path β_3. Since SAGE estimates the parameters of each path sequentially (in decreasing order of path strength), the LoS component in this channel will be processed last, and is expected to suffer the most from incorrect cancellation of the other paths. Figure 3.2 shows the distributions of the final path estimates from 100 simulations at two different SNR environment when path 2 was poorly initialised. From the results in the standard case (left column), it is evident that the incorrect estimation of path 2 (and subsequent removal of this path) severely effects the estimation of the weaker LoS component; such that there were almost no correct estimates of the LoS component (same can be said about path 2 itself). In fact, most of the estimates for the LoS component have converged close to the strongest path, which explains the unusually large distribution at path 3. We can observe a stark contrast in the results of the adaptive IC SAGE (right column), where at 10dB, distributions of all paths are clearly distinguishable.

3.4.2 Multipath Performance

Next, we compare the estimation accuracy of the two algorithms under a strong multipath environment. In the first instance, we utilise the weak LoS scenario, where the

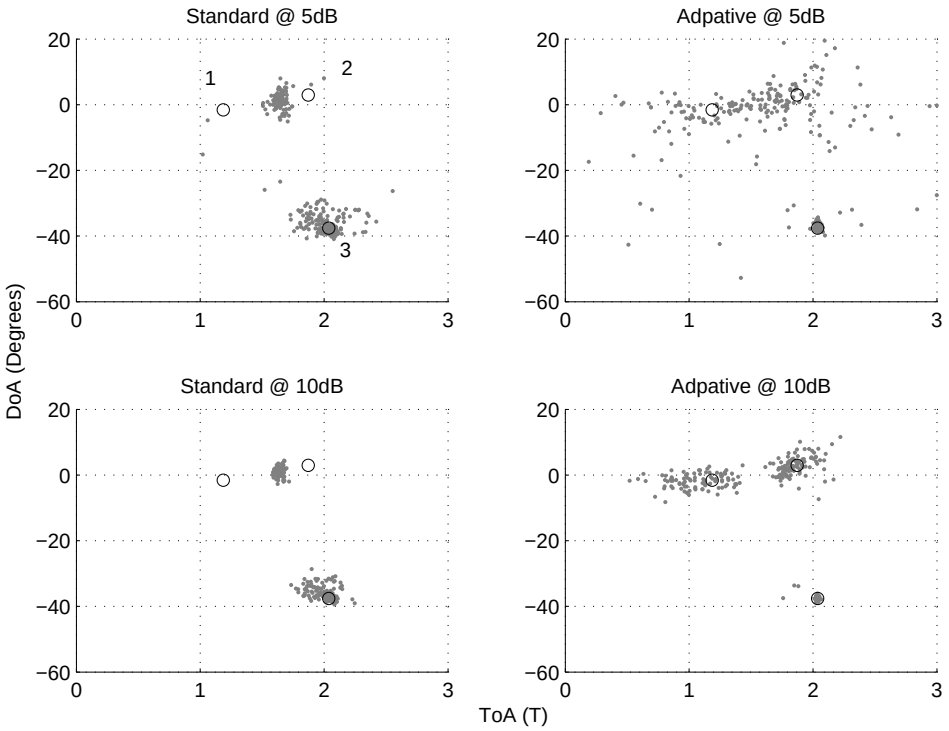
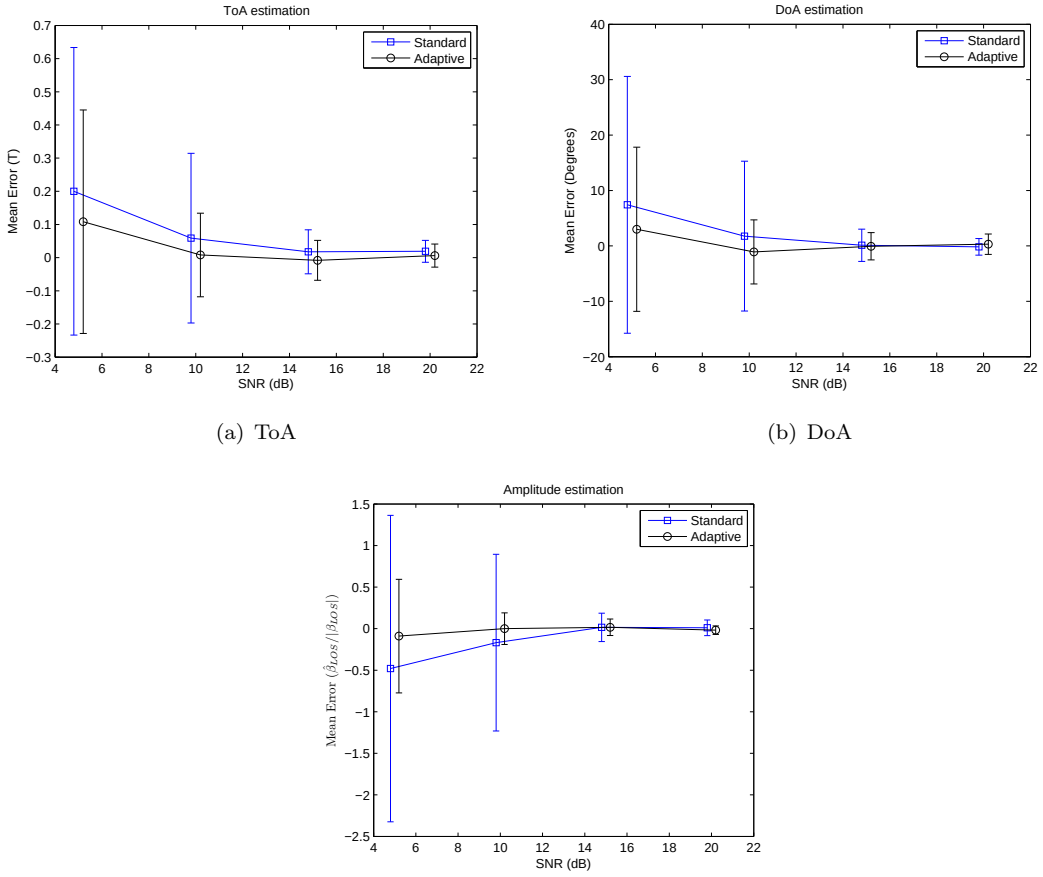


FIGURE 3.2: Distribution of final path estimates when path 2 was poorly initialised. Open circles represent actual path locations. Parameters for the 3 paths were: $\{\tau_1 = 1.2T, \theta_1 = -1.5°\}$,$\{\tau_2 = 1.9T, \theta_2 = 3.0°\}$,$\{\tau_3 = 2.0T, \theta_3 = -37.6°\}$. Multipath numbers are labelled in only 1 sub-figure to reduce clutter.

amplitude of the LoS component was limited to 10% of the strongest path, to compare the LoS component estimation accuracy of the adaptive method with the standard case. Secondly, general multipath identification peformance is compared when the channel contains a mixture of strong and weak multipath.

Assuming no prior knowledge of the location of the LoS component, hence choosing the path estimate with the shortest ToA value as the LoS estimate. Figure 3.3 shows the mean errors and standard deviation values in ToA, DoA and amplitude estimation of the LoS component against SNR values. We can observe noticeable improvement in ToA and DoA estimation at low SNR values, where up to 50% reduction in mean error can be seen in both estimates at 5dB; and the same level of reduction in standard deviation values is seen at 10dB in Figure 3.3(b). However, the biggest improvement occurs in amplitude estimation as shown in Figure 3.3(c). This is non-surprising since adaptive IC employing weights is analogous to iteratively applying an "correction factor"



(a) ToA

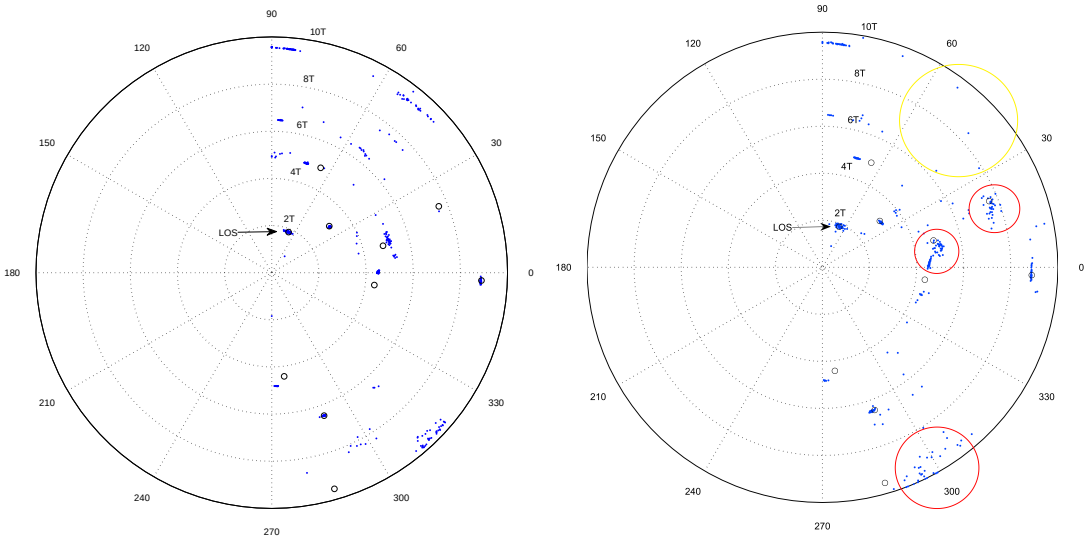
(b) DoA

(c) Amplitude

FIGURE 3.3: Mean error values of the weak LoS estimate from 100 Monte-Carlo simulations. Length of vertical bar represent ± 1 standard deviation of parameter estimates. $L = 8$.

to the closed form SAGE estimate of the amplitude in (2.49) to enhance the amplitude estimation.

Figure 3.4 shows the path distributions in a severe multipath environment containing a mixture of strong and weak paths. From initial inspection, there is no significant improvement in the adaptive method for the paths which are already identified by the standard SAGE, such as the 2 closest paths to the BS. However, closer inspection of the highlighted areas (circles) reveals areas of clear improvements for the much weaker paths. The 3 red circles indicates improvements in identifying multipath, whereas the large yellow circle shows an area of significant reduction in erroneous estimates.



(a) Standard (b) Adaptive

FIGURE 3.4: Distribution of multipath estimates from standard and adaptive interference cancellation. L=10; SNR=15dB.

3.5 Summary

We have presented an improved SAGE algorithm employing adaptive interference cancellation, where a set of weights is used to control the amount of cancellation at each iteration. These weights are obtained from the LMS algorithm, which tries to minimise the MSE between the received signal and its estimate. Simulation results in weak-LoS environments show considerable improvement over the standard SAGE algorithm where brute force cancellation is employed.

Chapter 4

Joint Channel Parameter Estimation Using Evolution Strategy

4.1 Introduction

The fundamental approach to optimisation is to formulate a single standard of measurement - a cost function that summaries the performance of a decision and iteratively improve this performance by selecting from among the available alternatives. Most classical optimisation methods generate a deterministic sequence of trial solutions based on the gradient or higher-order statistics of the cost function. Under regularity conditions on this function, these techniques can be shown to generate sequences that asymptotically converge to locally optimal solutions [80]. But these methods often fail to perform adequately when the cost function is highly non-linear or disturbed by random perturbations. Furthermore, locally optimal solutions often prove insufficient for real-world engineering problems.

Evolutionary Algorithms (EA) [80–83] are intelligent optimisation techniques based on the model of natural evolution. These types of algorithms exploit the collective learning process within a population of individuals, while each of the individuals represents a search point in the space of potential solutions to a given problem. After an arbitrary initialisation, the population evolves towards increasingly better regions of the search

space by means of simulated processes of *selection*, *mutation*, and *recombination*. The environment, which is the optimisation problem being solved, provides a fitness to each of the individuals in the population. The selection process favours individuals of higher fitness to reproduce more often than others. While new individuals are produced from the selected individuals via recombination and mutation. Recombination allows for exchange of information between individuals, while mutation introduces innovation into the population.

In this chapter, we propose a new joint channel parameter estimation technique by using an EA approach. The problem is formulated as the joint ML estimation of the channel parameters where typically, the high dimensional non-linear cost function is deemed to be too computationally expensive to be solved. In such cases, the performance of traditional iterative methods is often inadequate and does not find the global optimum. Our simulation results demonstrate the proposed method is extremely robust to errors in initialisation and shows superior performance at low Signal to Noise Ratio (SNR) environments. In addition, the computational complexity of the proposed method is demonstrated to be less than that of the traditional iterative ML approach.

4.2 Evolution Strategy

Three mainstream algorithms following the EA principles can nowadays be identified [81]:

- Evolution Strategy (ES) (Rechenberg, 1973; Schwefel, 1975)
- Genetic Algorithm (GA) (DeJong, 1975; Holland, 1975; Goldberg, 1989)
- Evolutionary Programming (EP) (Fogel et al., 1966; Fogel, 1991; 1992)

Here, we focus on Evolution Strategy and refer the interested reader to [80] and the references within for a more detailed comparisons of the three algorithms. All of them have clearly demonstrated their strong potential for solving complex optimisation problems in a variety of applications [83]. Among which Evolution Strategy is shown to be the least intricate solution given that it is able to self-adapt its core strategic parameters, thus eliminating the tedious task of adjusting vital controlling parameters for different conditions. In addition, no special emphasis is placed on the representation of the

optimisation variables, hence it is particularly suited to arbitrary precision real valued optimisation problems.

Evolution Strategy, piloted by Rechenberg in the 1960s and further explored by Schwefel was used as a technique of evolutionary experimentation for solving complex optimisation problems, mainly within engineering domains. Rechenberg reasoned that, since biological processes have been optimised by evolution, and evolution is a biological process itself, then it must be the case that evolution optimises itself [84].

The basic principle of ES is to mutate a parent solution by adding random perturbations in the hope of creating a fitter offspring. In fact, the original version of ES, named Two-Membered ES or $(1+1)$-ES[1], is based upon a population consisting of just one parent individual, and one offspring, created by means of adding normally distributed random numbers:

$$\mathbf{y} = \mathbf{x} + \mathcal{N}(0, \sigma) \tag{4.1}$$

where $\mathbf{x}$ and $\mathbf{y}$ are the parent and offspring solutions respectively. The fitter of the two is then selected as the ancestor for the next generation.

The $(1+1)$-ES can be designated as a kind of probabilistic gradient search technique - not, however, as a pure random or Monte-Carlo method; since the population principle has not really been used. The population principle was introduced later in the multi-membered ES. In the subsequent sections, any references made to ES refers only to the multi-membered type.

An ES strives to drive a population of candidate solutions to an optimisation problem towards increasingly better regions of the search space by means of variation and selection. The optimisation problems are typically of the type:

$$\min \left\{ f(\mathbf{x}) | \mathbf{x} \in M \subseteq \mathbb{R}^n \right\} \tag{4.2}$$

where $f : \mathbb{R}^n \mapsto \mathbb{R}$ is called the *objective function* and the set of M feasible region. Figure 4.1 illustrates the evolution loop that is cycled through repeatedly. The parameters λ and μ refer to the number of candidate solutions generated per generation and the number of those retained after selection. Executing variation (recombination/mutation)

[1] In ES literature, the addition sign is traditionally is used denote the selection scope

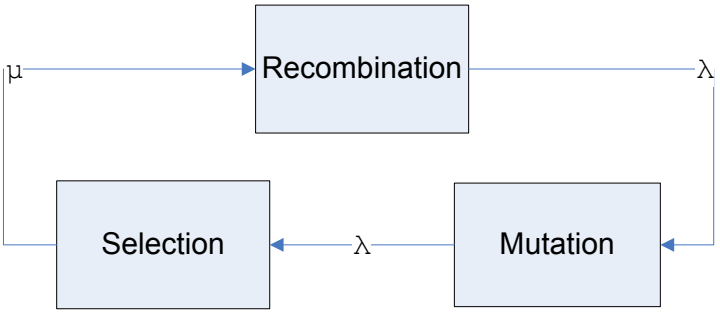


FIGURE 4.1: Given a function to be minimised, randomly create a set μ of candidate solutions of which an abstract fitness measure is applied. Then, $\lambda > \mu$ descendants are generated by means of recombination and subsequently subject to mutation. Next, μ of the fittest descendants are selected to form the parental population of the next generation.

leads to a set of new solutions that competes - based on their fitness, for a place in the next generation. The combined application of variation (creates necessary diversity) and selection (force pushing quality) generally leads to improved fitness values in consecutive populations [85].

4.3 Joint Channel Parameter Estimation

For our joint channel parameter estimation problem, we define the *objective function* as the Euclidean distance between the received signal $\mathbf{y}(t)$ at the antenna array and the sum of the estimates of all paths' signals:

$$\varepsilon(\mathbf{y};\boldsymbol{\omega}) \triangleq \int_0^{T_o} \left| \mathbf{y}(t) - \sum_{\ell=1}^{L} \hat{\beta}_\ell \mathbf{a}(\hat{\theta}_\ell) g(t-\hat{\tau}_\ell) \right|^2 dt \tag{4.3}$$

here, T_o is the observation period, and the combined vector of channel parameters $\boldsymbol{\omega} \triangleq [\boldsymbol{\omega}_1 | \boldsymbol{\omega}_2 | \ldots, | \boldsymbol{\omega}_L]$ contains the channel parameters of all paths, where $\boldsymbol{\omega}_\ell = [\beta_\ell, \tau_\ell, \theta_\ell]$ contains the channel parameters for each multipath. Then, the least square estimate of $\boldsymbol{\omega}$ is the value of this vector for which $\boldsymbol{\omega} \mapsto \varepsilon(\mathbf{y};\boldsymbol{\omega})$ is a minimum,

$$\hat{\boldsymbol{\omega}} \in \arg\min_{\boldsymbol{\omega}} \{\varepsilon(\mathbf{y};\boldsymbol{\omega})\} \tag{4.4}$$

this is equivalent to the ML estimate of $\boldsymbol{\omega}$ given the observation $\mathbf{y}(t)$. Then our objective is to optimise for the ML problem in (4.4) by the evolution of a population of multiple

candidate parameter estimates $\hat{\boldsymbol{\omega}}$, each of which is assessed according to its *fitness* using (4.3). For reasons of notational simplicity, we will use the notation ω_i to indicate the ith element of the combined vector of channel parameter estimates $\boldsymbol{\omega}$ in the subsequent sections, no distinctions will be made to the type of channel parameters, e.g. τ or θ etc.

The parameter space dimension n for the minimisation problem in (4.4) is $n \triangleq 3L$, i.e. there are 3 unknown channel parameters associated with each of the L paths. This is considered to be computationally prohibitive for most practical number of paths. Up to now, iterative ML methods such as EM and SAGE are commonly used to search for approximate solutions to (4.4). In this work, we will show that our ES approach for solving (4.4) directly is computationally less demanding than these iterative methods; while at the same time does not suffer from poor initialisation and residue error problems.

4.3.1 Representation

In this thesis, instead of using the standard "comma/plus" ES notation, which can be potentially misleading, we use the notations $(\mu_\rho \odot \lambda)$ and $(\mu_\rho \oplus \lambda)$ to represent the characteristics of an ES - μ number of parents produces λ offspring which are reduced again to the μ parents of the next generation. In the reproduction process, each of the offspring is the recombination of ρ number of parents. While the operators $\odot$ and $\oplus$ define the survival selection scope for the next generation. In subsequent sections, it is often necessary to refer to both type of selection scopes, in such cases, we will use the notation $(\mu_\rho \bullet \lambda)$ to indicate an ES utilising either of the above mentioned selection types. The proposed ES method in this thesis can be defined by the following 11-tuple:

$$(\mu_\rho \bullet \lambda) - \text{ES} \triangleq \left(\boldsymbol{\Psi}^{(k)}, \mu, \lambda, \rho, n_\sigma, n_\phi, \mathbf{rec}, \mathbf{mut}, \mathbf{sel}, \gamma', \gamma, \varphi\right)$$

$\boldsymbol{\Psi}^{(k)} = (\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_\mu)^{(k)} \in I^\mu$	$I = \mathbb{R}^n \times \mathbb{R}_+^{n_\sigma} \times [-\pi, \pi]^{n_\phi}$	candidate solutions
$\mu \in \mathbb{N}$	$\mu \geq 1$	number of parents
$\lambda \in \mathbb{N}$	$\lambda \geq \mu$	number of offspring
$\rho \in \mathbb{N}^3$	$1 \leq \rho \leq \mu$	recombination factor
$n_\sigma \in \mathbb{N}$	$1 \leq n_\sigma \leq n$	number of step sizes
$n_\phi \in \mathbb{Z}$	$0 \leq n_\phi \leq (2n - n_\sigma)(n_\sigma - 1)/2$	number of rotation angles
$\mathbf{rec} : I^\mu \mapsto I$		recombination operator

$\mathbf{mut} : I \mapsto I$		mutation operator
$\mathbf{sel} : I^{\mu+\lambda} \mapsto I^{\mu} \vert I^{\lambda} \mapsto I^{\mu}$		selection operator
$\gamma', \gamma \in \mathbb{R}_+$		step length variabilities
$\varphi \in \mathbb{R}_+$	$0 \leq \varphi \leq \pi/4$	correlation variabilities

Each population individual at the kth generation is represented by a vector:

$$\boldsymbol{\psi}_j^{(k)} = \left(\boldsymbol{\omega}, \boldsymbol{\sigma}, \boldsymbol{\phi}\right)^{(k)} \in \boldsymbol{\Psi}^{(k)}, \qquad j \in \{1, \ldots, \mu\} \tag{4.5}$$

$\boldsymbol{\omega} \in \mathbb{R}^n$	combined vector of channel estimates, the only part of $\boldsymbol{\psi}_j^{(k)}$ entering the objective function
$\boldsymbol{\sigma} \in \mathbb{R}_+^{n_\sigma}$	step sizes (standard deviations of the Normal distribution used for simulating mutations)
$\boldsymbol{\phi} \in [-\pi, \pi]^{n_\phi}$	rotation angles used to define the correlation between the individual step sizes

In the proposed method, no emphasis is placed on the genetic representation of the individuals. Hence the candidate solutions $\boldsymbol{\omega}$ are directly represented by the actual channel parameter values referred to in ES terms as *object variables*. The entire collection of μ candidate solutions of the channel parameter estimates $[\boldsymbol{\omega}_{(1)}, \ldots, \boldsymbol{\omega}_{(\mu)}]^T$ forms the $\mu \times n$ object variable part of the population matrix:

$$\boldsymbol{\Omega} = \begin{bmatrix} \boldsymbol{\omega}_{(1)} \\ \boldsymbol{\omega}_{(2)} \\ \vdots \\ \boldsymbol{\omega}_{(\mu)} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\omega}_{(1)1} & \boldsymbol{\omega}_{(1)2} & \cdots & \boldsymbol{\omega}_{(1)L} \\ \boldsymbol{\omega}_{(2)1} & \boldsymbol{\omega}_{(2)2} & \cdots & \boldsymbol{\omega}_{(2)L} \\ \vdots & & & \\ \boldsymbol{\omega}_{(\mu)1} & \boldsymbol{\omega}_{(\mu)2} & \cdots & \boldsymbol{\omega}_{(\mu)L} \end{bmatrix} \tag{4.6}$$

where we have used the notation $\boldsymbol{\omega}_{(j)\ell}$ to represent the vector containing the jth instance of the path ℓ's channel parameters. After grouping of similar channel parameter terms and expanding becomes:

$$\boldsymbol{\Omega} = \begin{bmatrix} \beta_{(1)1} & \cdots & \beta_{(1)L} & \theta_{(1)1} & \cdots & \theta_{(1)L} & \tau_{(1)1} & \cdots & \tau_{(1)L} \\ \beta_{(2)1} & \cdots & \beta_{(2)L} & \theta_{(2)1} & \cdots & \theta_{(2)L} & \tau_{(2)1} & \cdots & \tau_{(2)L} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \beta_{(\mu)1} & \cdots & \beta_{(\mu)L} & \theta_{(\mu)1} & \cdots & \theta_{(\mu)L} & \tau_{(\mu)1} & \cdots & \tau_{(\mu)L} \end{bmatrix} \tag{4.7}$$

where each row[2] represents one candidate solution to the minimisation problem in (4.4).

The vectors $\boldsymbol{\sigma}$ and $\boldsymbol{\phi}$ in (4.5) are known as the *strategy parameters* of the individuals. They determine the variances and its correlations of the n-dimensional Normal mutation probability density that is used to mutate the channel parameter estimates $\boldsymbol{\omega}$. Hence, besides representing the object variable vector, each individual may additionally include n_σ different standard deviations σ_i:

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\sigma}_{(1)} \\ \boldsymbol{\sigma}_{(2)} \\ \vdots \\ \boldsymbol{\sigma}_{(\mu)} \end{bmatrix} = \begin{bmatrix} \sigma_{(1)1} & \cdots & \sigma_{(1)n_\sigma} \\ \sigma_{(2)1} & \cdots & \sigma_{(2)n_\sigma} \\ \vdots & \ddots & \vdots \\ \sigma_{(\mu)1} & \cdots & \sigma_{(\mu)n_\sigma} \end{bmatrix} \tag{4.8}$$

where each row is the n_σ number of standard deviations that is used to mutate each candidate solution in the matrix $\boldsymbol{\Omega}$; and n_ϕ different rotation angles ϕ_i:

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\phi}_{(1)} \\ \boldsymbol{\phi}_{(2)} \\ \vdots \\ \boldsymbol{\phi}_{(\mu)} \end{bmatrix} = \begin{bmatrix} \phi_{(1)1} & \cdots & \phi_{(1)n_\phi} \\ \phi_{(2)1} & \cdots & \phi_{(2)n_\phi} \\ \vdots & \ddots & \vdots \\ \phi_{(\mu)1} & \cdots & \phi_{(\mu)n_\phi} \end{bmatrix} \tag{4.9}$$

where each one of the rotational angles ϕ_i describes the correlation between 2 pairs of standard deviations in $\boldsymbol{\sigma}_{(j)}$. Therefore, each candidate solution can be associated with up to $n(n+1)/2$ number of strategy parameters, however, this setting is rarely used.

It is intuitive that the success of our method in finding a solution close to the true optimum is largely dependent on the choice of the mutation strengths σ_i. In addition, as the solution comes closer and closer to the optimum, the mutation strength should reduce proportionally. This phenomenon is exploited by the self-adaptive nature of our method, where in addition to the channel parameters, these strategy parameters are neither constant nor explicitly controlled, but undergo a logarithmic-normally distributed variation mechanism, where they are, through the same process of recombination and mutation, evolved during the search process.

[2] All entries in $\boldsymbol{\Omega}$ are normalised such that they are uniformly distributed between $(0, 1)$. This makes the subsequent processing of $\boldsymbol{\Omega}$ much simpler.

Together, the matrices $\mathbf{\Omega}$, $\mathbf{\Sigma}$ and $\mathbf{\Phi}$ form the conceptual $\mu \times (n + n_\sigma + n_\phi)$ population matrix $\mathbf{\Psi} = \big[\mathbf{\Omega}|\mathbf{\Sigma}|\mathbf{\Phi}\big]$, where each row corresponds to one complete ES individual of the form in (4.5). Although we have kept the channel parameters and its associated strategy parameters as separate entities, it is still important to visualise them together as the matrix $\mathbf{\Psi}$; such that the jth row of $\mathbf{\Sigma}$ and $\mathbf{\Phi}$ is used to mutate the jth row in $\mathbf{\Omega}$ and eventually, will survive/decease together.

4.3.2 Mutation

Mutation is the primary source of genetic variation in ES [86]. Given an individual space $I = \mathbb{R}^n \times \mathbb{R}_+^{n_\sigma} \times [-\pi, \pi]^{n_\phi}$, the mutation operator $\mathbf{mut} : I^\lambda \to I^\lambda$ is applied to all individuals separately, hence it is sufficient to present its reduced form:

$$\text{mutation:} \qquad \mathbf{mut} : I \to I \tag{4.10}$$

Various types of mutation based on the normal (Gaussian) distribution are used in the proposed method. In the simplest case of uncorrelated mutation with a single step size σ, the same distribution is used to mutate each element of the channel parameters ω_i. Mutations are realised by adding some $\Delta\omega_i$ to each ω_i, where the $\Delta\omega_i$ values are randomly drawn using the given Gaussian $\mathcal{N}(\xi, \sigma)$, with the following probability density function:

$$p(\Delta\omega_i) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(- \frac{(\Delta\omega_i - \xi)^2}{2\sigma^2} \right) \tag{4.11}$$

in practice, the mean ξ is always set to zero.

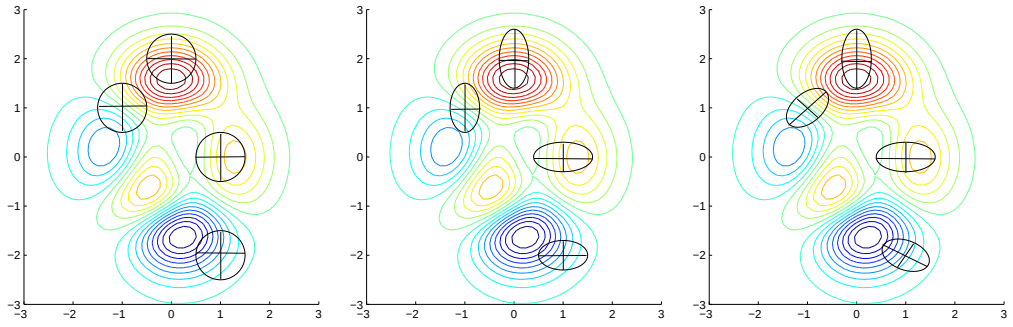
The mutation operator produces a new individual $\boldsymbol{\psi}' = \mathbf{mut}(\boldsymbol{\psi})$ by first mutating the standard deviation (following a log-normal distribution[3]) and then modifying the channel parameters using the newly modified probability density function characterised by σ' :

$$\begin{aligned} \sigma' &= \sigma \exp(\gamma \mathcal{N}(0,1)) \\ \omega_i' &= \omega_i + \sigma' \mathcal{N}_i(0,1) \end{aligned} \tag{4.12}$$

where $\mathcal{N}(0,1)$ are independent random samples from the standard Gaussian distribution, while $\mathcal{N}_i(0,1)$ denotes a separate draw from the standard Gaussian distribution for each

[3] if $x \sim \mathcal{N}(\mu, \sigma^2)$ then $\exp^x \sim logN(\mu, \sigma^2)$ for $x \in (0, +\infty)$

variable i. It is clear that the mutation operator prefers small changes and due to its symmetry it introduces no bias: the expected value is zero and the mutants ω_i' are distributed symmetrically around the parental state.



(a) Isotropic (b) Non-isotropic (c) Correlated

FIGURE 4.2: Illustration of mutation hyperellipsoids in the case of simple isotropic mutations with $n_\sigma = 1$ (a), non-isotropic mutations with $n_\sigma = 2$ (b), and correlated mutations with $n_\sigma = 2, n_\phi = 1$ (c). In each graphic four individuals and their ellipsoids of equal probability density to place an offspring are shown on top of the fitness landscape ($f : \mathbb{R}^2 \to \mathbb{R}$) represented by lines of equal objective function value.

The isotropic mutation operator defined in (4.12) has the advantage that it needs only one endogenous strategy parameter for its control. Since there is only one σ, the mutation step size is the same in each data dimension, see Figure 4.2 (a). However, there are situations where it can be beneficial to have mutation vectors those surfaces of constant density are ellipsoidal, where up to n number of standard deviations are used to mutate each solution vector, and the mutation components follow the multivariate Normal distribution:

$$p(\Delta\boldsymbol{\omega}) = \frac{1}{(2\pi)^{n/2}(\det(\mathbf{C}))^{1/2}} \exp\left(\frac{1}{2}\Delta\boldsymbol{\omega}^T\mathbf{C}^{-1}\Delta\boldsymbol{\omega}\right) \tag{4.13}$$

here, $\mathbf{C}$ is the positive definite and symmetric covariance matrix with the entries $\mathbf{C} = \boldsymbol{\sigma}^2\mathbf{I}$. The mutation mechanism using n separate standard deviations then becomes:

$$\begin{aligned} \sigma_i' &= \sigma_i \exp(\gamma'\mathcal{N}(0,1) + \gamma\mathcal{N}_i(0,1)) \\ \omega_i' &= \omega_i + \sigma_i'\mathcal{N}_i(0,1) \end{aligned} \tag{4.14}$$

where the parameters $\gamma' \sim 1/\sqrt{2n}$ and $\gamma \sim 1/\sqrt{2\sqrt{n}}$ in (4.12) and (4.14) are known in ES terms as the global and local learning rate respectively.

The motivation for using n_σ standard deviations is the wish to treat each dimension differently. This is based on the trivial observation that the fitness landscape can have different slopes in one direction than in another direction. The utilisation of multiple step sizes allows for bigger mutations on the coordinate axis which lies closer to the optima, see Figure 4.2 (b). The added degree of freedom provides for better flexibility, but the increased number of parameters causes a linear increase in computational complexity [84], since the mutation of each additional standard deviation requires a separate multiplication.

Finally, in the most general case of non-zero covariances, the mutation ellipsoids may have arbitrary orientation in the search space and can therefore adapt to any advantageous direction of the search, which is useful for example in the case of narrow valleys where an orientation of mutation ellipsoids along the valley is most appropriate, see Figure 4.2 (c).

The axes of the mutation ellipsoids are parallel with the coordinate axes if $\mathbf{C}$ is a diagonal matrix. If, however, the other elements c_{ij} for $i \neq j$, the covariances of the mutation steps sizes are non-zero, then the ellipsoids are rotated in space. The random components ω_i become mutually dependent, or correlated [87]. However, it is not advisable to incorporate the covariances directly into the representation and apply mutations according to:

$$\boldsymbol{\omega}' = \boldsymbol{\omega} + \mathcal{N}(0, \mathbf{C}) \tag{4.15}$$

because it is difficult to guarantee that the coordinate system remains orthogonal or, equivalently, the covariance matrix remains positive definite. This is why the rotation angles ϕ_i are used to performing the rotation alignments of the mutation ellipsoids. The covariances are given by the rotation angles ϕ_i which describe the rotations that need to be done to transform an uncorrelated mutation vector $\Delta\boldsymbol{\omega}_u$ to a correlated one $\Delta\boldsymbol{\omega}_c$.

Algorithmically, the generation of a correlated realisation $\Delta\boldsymbol{\omega}_u$ from an uncorrelated one $\Delta\boldsymbol{\omega}_c$ can be achieved by multiplication of $\Delta\boldsymbol{\omega}_u$ by n_ϕ rotation matrices $\mathbf{R}_{pq}(\phi_i)$, for $i = 1, \ldots, n_\phi$. Multiplication by such a matrix performs a coordinate transformation w.r.t axes number p and q and angle ϕ_i. Since n_ϕ rotations are needed to represent all

correlations, the complete relation between $\Delta\boldsymbol{\omega}_u$ and $\Delta\boldsymbol{\omega}_c$ results as [85]:

$$\Delta\boldsymbol{\omega}_c = \left(\prod_{p=1}^{n_\sigma - 1} \prod_{q=p+1}^{n_\sigma} \mathbf{R}_{pq}(\phi_i) \right) \Delta\boldsymbol{\omega}_u \tag{4.16}$$

with $i = \frac{1}{2}(2n_\sigma - p)(p+1) - 2n_\sigma + q$ and the rotation matrix $\mathbf{R}_{pq}(\phi_i)$ is given by a $n \times n$ unit matrix except for the following entries holds: $r_{pp} = r_{qq} = \cos(\phi_i)$, and $r_{pq} = -r_{qp} = -\sin(\phi_i)$:

$$\mathbf{R}_{pq}(\phi_i) = \begin{bmatrix} 1 & 0 & & & & \dots & & & & 0 \\ 0 & 1 & & & & & & & & \\ & & \ddots & & & & & & & \\ & & & 1 & & & & & & \\ & & & & \cos(\phi_i) & & & & -\sin(\phi_i) & \\ & & & & & 1 & & & & \\ \vdots & & & & & & \ddots & & & \vdots \\ & & & & & & & 1 & & \\ & & & & \sin(\phi_i) & & & & \cos(\phi_i) & \\ & & & & & & & & & 1 \\ & & & & & & & & \ddots & \\ & & & & & & & & & 1 & 0 \\ 0 & & & & & \dots & & & & 0 & 1 \end{bmatrix} \tag{4.17}$$

An efficient way of calculating (4.16) is the multiplication from right to left, i.e. n_ϕ matrix-vector multiplications as suppose to $n_\phi - 1$ matrix-matrix followed by 1 matrix-vector multiplications. However, incorporating correlated mutations increases computational complexity quadratically, in addition, for every increment in the number of multipath, the search space is tripled; hence, correlated mutations are not recommended unless the number of multipath is small.

The complete correlated mutation mechanism employed in our method can then be described as:

$$\begin{aligned} \sigma_i' &= \sigma_i \exp(\gamma' \mathcal{N}(0,1) + \gamma \mathcal{N}_i(0,1)) \\ \phi_j' &= \phi_j + \varphi \mathcal{N}_j(0,1) \\ \boldsymbol{\omega}' &= \boldsymbol{\omega} + \mathcal{N}(0, \boldsymbol{\sigma}', \boldsymbol{\phi}') \end{aligned} \tag{4.18}$$

where $\mathcal{N}(0, \boldsymbol{\sigma}', \boldsymbol{\phi}')$ denotes a realisation of a random vector distributed according to the generalised n-dimensional Gaussian distribution with expectation 0 and covariance matrix determined by $\boldsymbol{\sigma}'$ and $\boldsymbol{\phi}'$.

Common to all of the mutation mechanisms mentioned above, it is possible for the standard deviations σ to become practically zero by the multiplicative process and for the rotation angles ϕ to leave the range $[-\pi, \pi]$ of feasible values. To prevent both events, the following rules should be enforced:

$$\begin{aligned} \sigma_i' < \epsilon_\sigma &\mapsto \sigma_i' = \epsilon_\sigma \\ |\phi_j'| > \pi &\mapsto \phi_j' = \phi_j' - 2\pi \times \text{sign}(\phi_j') \end{aligned} \tag{4.19}$$

Altogether, the above mutation mechanism enables the proposed ES method to evolve its own strategy parameters during the search, exploiting an implicit link between advantageous changes of object variables and useful strategy parameters. Meaning that, an individual, represents a good $\boldsymbol{\omega}'$ that survived selection and a good set of strategy parameters $\boldsymbol{\sigma}'$ and $\boldsymbol{\phi}'$ that proved successful in generating this good $\boldsymbol{\omega}'$ from $\boldsymbol{\omega}$. This mechanism of *self-adaptation* of strategy parameters allows for an adaptation of these parameters without the need for finding an appropriate exogenous control mechanism [88].

4.3.3 Recombination

Although mutation is the primary search factor employed in ES, the recombination operator is indispensable, too, for the strategy to self-adapt the strategy parameters properly [83]. Besides mutation, recombination also works on object variables and strategy parameters, and different recombination operators may be used to manipulate object

variables and standard deviations. Recombination always creates just one offspring individual per application of the operator to the parent population and is applied as often as necessary to produce a complete offspring population.

A variety of different recombination schemes are currently used in ES, and the operators are sexual as well as panmictic. What follows is a description of the generalised recombination operator employed in the proposed method for creating an individual $\tilde{\boldsymbol{\psi}} = \mathbf{rec}(\boldsymbol{\psi})$ from the population $\boldsymbol{\Psi}^{(k)} \in I^{\mu}$:

$$\text{recombination} \qquad \mathbf{rec} : I^{\mu} \rightarrow I \tag{4.20}$$

The notation $(\mu_{\rho} \bullet \lambda)$ is used to describe multi-recombinant ES. Here the parameter ρ determines the number of parents that multi-combine to form one offspring. In contrast to the biology standard, where $\rho = 2$, it is worth using multi-mixing, i.e. $\rho > 2$, for ES applications [82]. There are two different recombination patterns - *intermediate* $(\mu_{\rho_I} \bullet \lambda)$ and *discrete* $(\mu_{\rho_D} \bullet \lambda)$ recombination. Empirically, discrete recombination on object variables and intermediate recombination on strategy parameters have been observed to yield best results, and the recombination of strategy parameters has shown to be essential for the self-adaptation mechanism to work.

Intermediate recombination is typically applied to the strategy parameters $\boldsymbol{\sigma}$ and $\boldsymbol{\phi}$; descendant are produced by some kind of averaging. In $(\mu_{\rho_I} \bullet \lambda)$ strategies ρ individuals $\boldsymbol{\sigma}_{m_\nu}, \nu = 1 \ldots, \rho$ from the parent pool $\{\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_\mu\}$ are chosen at random. The averaging is simply the determination of the centre of mass:

$$\begin{aligned} \tilde{\boldsymbol{\sigma}} &= \frac{1}{\rho} \sum_{\nu=1}^{\rho} \boldsymbol{\sigma}_{m_\nu} \\ m_\nu &= \text{random}\{1, \ldots, \mu\} \\ m_1 &\neq m_2 \neq, \ldots, \neq m_\rho \end{aligned} \tag{4.21}$$

where the ρ random indices m_ν are all unique. However, this is rather elaborate, and a more practical version could allow "self-fertilisation" by sampling the ρ m_ν-numbers from a simple random number generator.

A special case of the $(\mu_{\rho_I} \bullet \lambda)$ strategies occurs when $\rho = \mu$ holds, i.e. there is no random mating. This means all parents are involved in the recombination:

$$\tilde{\boldsymbol{\sigma}} = \frac{1}{\mu} \sum_{m=1}^{\mu} \boldsymbol{\sigma}_m \tag{4.22}$$

hence, each offspring is generated from the centre of mass individual in (4.22) plus a random mutation.

Discrete recombination is typically applied to the channel parameters $\boldsymbol{\omega}$; descendant are produced by randomly choosing components from the selected individuals. An offspring $\tilde{\boldsymbol{\omega}}$ is constructed from ρ different, randomly chosen parents $\boldsymbol{\omega}_{\eta_\nu}$:

$$\begin{aligned} \tilde{\mathbf{x}} &= \sum_{i=1}^{n} (\mathbf{e}_i^T \boldsymbol{\omega}_{m_i}) \mathbf{e}_i \\ m_i &= \text{random}\{\eta_1, \ldots, \eta_\rho\} \\ \eta_\nu &= \text{random}\{1, \ldots, \mu\} \end{aligned} \tag{4.23}$$

that is, the coordinate i of the $\tilde{\boldsymbol{\omega}}$-vector is given by random choice of one of the coordinate i-values from the mates of the index set $\{\eta_1, \ldots, \eta_\rho\}$; where $\mathbf{e}_i$ represents the standard unit vector basis of length n.

A special case of the $(\mu_{\rho_D} \bullet \lambda)$ strategies also occurs when $\rho = \mu$ holds, i.e. each coordinate i of the offspring $\tilde{\boldsymbol{\omega}}$ can be chosen randomly from the entire parent population:

$$\begin{aligned} \tilde{\boldsymbol{\omega}} &= \sum_{i=1}^{n} (\mathbf{e}_i^T \boldsymbol{\omega}_{m_i}) \mathbf{e}_i \\ m_i &= \text{random}\{1, \ldots, \mu\} \end{aligned} \tag{4.24}$$

which is sometime known as "dominant recombination".

An example of discrete recombination of the channel parameters using $\rho = 4$ parents is shown below:

$$\begin{array}{lllllll} \boldsymbol{\omega}^{(1)} \text{ (Parent 1)} & \omega_1^{(1)} & \omega_2^{(1)} & \omega_3^{(1)} & \omega_4^{(1)} & \omega_5^{(1)} & \omega_6^{(1)} \\ \boldsymbol{\omega}^{(2)} \text{ (Parent 2)} & \dot{\omega}_1^{(2)} & \omega_2^{(2)} & \omega_3^{(2)} & \dot{\omega}_4^{(2)} & \omega_5^{(2)} & \omega_6^{(2)} \\ \boldsymbol{\omega}^{(3)} \text{ (Parent 3)} & \omega_1^{(3)} & \dot{\omega}_2^{(3)} & \omega_3^{(3)} & \omega_4^{(3)} & \omega_5^{(3)} & \dot{\omega}_6^{(3)} \\ \boldsymbol{\omega}^{(4)} \text{ (Parent 4)} & \omega_1^{(4)} & \omega_2^{(4)} & \dot{\omega}_3^{(4)} & \omega_4^{(4)} & \dot{\omega}_5^{(4)} & \omega_6^{(4)} \\ \textbf{Recombinant:} & \omega_1^{(2)} & \omega_2^{(3)} & \omega_3^{(4)} & \omega_4^{(2)} & \omega_5^{(4)} & \omega_6^{(3)} \end{array} \tag{4.25}$$

where the newly formed recombined offspring is made up of the dotted components denoted by $\dot{\omega}$ from the pool of 4 parents in (4.25).

Since it is often the case that both recombination types are used in an ES implementation, it is an accepted convention not to specify explicitly the recombination types in these cases, and the notation $(\mu_\rho \bullet \lambda)$ is often sufficient. While the notations $(\mu_{\rho_I} \bullet \lambda)$ and $(\mu_{\rho_D} \bullet \lambda)$ are normally reserved for ES instances employing exclusively intermediate or discrete recombinations types. Furthermore, when the global recombination setting is used ($\rho = \mu$), the recombination factor ρ is normally dropped from the notations to avoid the potentially confusing usage of $(\mu_\mu \bullet \lambda)$. Hence, in the following work, we will often use the simplified notation $(\mu \bullet \lambda)$ to describe an ES implementation employing global discrete recombination for channel parameters, and global intermediate recombination for the strategy parameters.

An implementational characteristic of recombination in ES is the property to "blow up" the population size from μ individuals in the parent population to λ individuals in the offspring population by applying recombination λ times, this also implicitly models the reproduction process.

4.3.4 Selection

The selection operators used in the proposed method are completely deterministic. Schwefel distinguishes between:

$$\begin{aligned} (\mu \oplus \lambda) - \text{selection} : \quad & \mathbf{sel}^{\oplus} : I^{\mu+\lambda} \rightarrow I^{\mu} \\ (\mu \odot \lambda) - \text{selection} : \quad & \mathbf{sel}^{\odot} : I^{\lambda} \rightarrow I^{\mu} \end{aligned} \tag{4.26}$$

where the former selects the μ best individuals out of the union of parents and offspring to form the next parent generation, while the latter selects the μ best individuals out of the offspring only.

Although the elitist $(\mu \oplus \lambda)$-selection guarantees a monotonously improving performance, it has several disadvantages when compared to $(\mu \odot \lambda)$-selection, which restricts the lifespan of individuals to just one generation:

- The $(\mu \oplus \lambda)$-selection preserves (outdated) solutions and is not able to follow the moving optimum in a changing fitness environment.
- The $(\mu \odot \lambda)$-selection discards all parents and is therefore in principle able to leave (small) local optima, hence is advantageous in the case of multimodal topologies.
- $(\mu \oplus \lambda)$-selection hinders the self-adaptation mechanism w.r.t strategy parameters to work effectively, because misadapted strategy parameters may survive for a large number of generations when they cause a fitness improvement by chance.

For the above mentioned reasons, $(\mu \oplus \lambda)$ selection is rarely used in ES literature, and our proposed ES method employs $(\mu \odot \lambda)$ selection only.

Traditionally, the selective pressure in ES is very high because λ is typically much higher than μ. The empirical investigation in [83] indicates a ratio of $\mu/\lambda \approx 1/7$ to be optimal, where μ should not be chosen too small, e.g. a widely used setting is $(15 \bullet 100)$-selection. We will demonstrate later that the selection pressure can be influential to the performance of the proposed method.

Algorithm 4.1 Outline of proposed ES method

$k := 0$
set $\big(\mu, \lambda, \rho \in \{1, \ldots, \mu\}, n_\sigma \in \{1, \ldots, n\}, n_\phi \in \{0, \ldots, n(n-1)/2\}\big)$
$[\mathbf{\Omega}]_{i,j}^{(0)} \in U(0,1) \quad [\mathbf{\Sigma}]_{i,j}^{(0)} \in U(0, \sigma_{\max}) \quad [\mathbf{\Phi}]_{i,j}^{(0)} \in U(-\pi, \pi)$
while not terminated **do**
    for $j = \{1, \ldots, \lambda\}$ **do**
        $\tilde{\boldsymbol{\psi}}_j^{(k)} := \mathbf{rec}\big(\mathbf{\Psi}^{(k)}\big)$ {recombine}
        $\boldsymbol{\psi}_j'^{(k)} := \mathbf{mut}\big(\tilde{\boldsymbol{\psi}}_j^{(k)}\big)$ {mutate}
        $\varepsilon_j^{(k)} := \varepsilon\big(\boldsymbol{\psi}_j'^{(k)}\big)$ {evaluate fitness}
    end for
    $\mathbf{\Psi}^{(k+1)} := \mathbf{sel}\big([\varepsilon_1^{(k)}, \ldots, \varepsilon_\lambda^{(k)}], \mathbf{\Psi}'^{(k)}\big)$ {select}
    if $\sum_{i=1}^{n} \Big[[\bar{\mathbf{\Omega}}]_{:,i}^{(k)} - [\bar{\mathbf{\Omega}}]_{:,i}^{(k-1)}\Big] \leq \delta$ **then**
        Terminate
    end if
    $\hat{\boldsymbol{\omega}}^{(k)} = \Big[[\bar{\mathbf{\Omega}}]_{:,1}^{(k)}, [\bar{\mathbf{\Omega}}]_{:,2}^{(k)}, \ldots, [\bar{\mathbf{\Omega}}]_{:,n}^{(k)}\Big]$
    $k := k + 1$
end while

In summary, the above proposed ES method can be described by the pseudocode in Algorithm 4.1. Recombination, mutation and selection are represented by the high-level operators $\mathbf{rec} : I^\mu \to I$, $\mathbf{mut} : I \to I$ and $\mathbf{sel} : I^\lambda \to \lambda^\mu$ respectively; where I denotes the space of individuals, i.e. $I = \Re^n \times \Re_+^{n_\sigma} \times [-\pi, \pi]^{n_\phi}$.

- Line 3: Since elements of the matrix $[\boldsymbol{\Omega}]_{i,j} \in (0,1)$ represent the normalised channel parameter estimates, then we can initialise the entries of $[\boldsymbol{\Omega}]_{i,j}$ with random samples from $U(0,1)$, where $U(0,1)$ is the standard uniform distribution between 0 and 1.

- Line 3: It follows that the mutations $\Delta\omega_i$ generated by (4.14) should be bounded by $-1 \leq \Delta\omega_i \leq +1$ for $\forall i \in \{1,\ldots,n\}$. Since 95% of all values in a Normal distribtuion falls under 2 standard deviations from the mean, the bound on $\Delta\omega_i$ can be achieved with high probability if we initialise $\boldsymbol{\Sigma}$ according to $[\boldsymbol{\Sigma}]_{i,j} \in U(0,\sigma_{\max})$, where $\sigma_{\max} \leq 1/2$.

- Line 8: Fitness evaluations for all λ offspring should be vectorised for speed. See Appendix B for details.

- Line 11: The termination criterion is the difference in the means of the columns of $\boldsymbol{\Omega}$ between the current and the previous generation: $\sum_{i=1}^{n}\left[[\bar{\boldsymbol{\Omega}}]_{:,i}^{(k)} - [\bar{\boldsymbol{\Omega}}]_{:,i}^{(k-1)}\right] \leq \delta$; where $[\bar{\boldsymbol{\Omega}}]_{:,i}^{(k)}$ represents the mean of the ith column of $\boldsymbol{\Omega}$ in generation k, and δ is a predefined constant.

- Line 14: Consequently, the set of n column means is used as the final parameter estimate once the termination criterion is met, i.e. $\hat{\boldsymbol{\omega}} = \left[[\bar{\boldsymbol{\Omega}}]_{:,1}^{(k)}, [\bar{\boldsymbol{\Omega}}]_{:,2}^{(k)}, \ldots, [\bar{\boldsymbol{\Omega}}]_{:,n}^{(k)}\right]$

4.4 Simulations

In this section, simulations in MATLAB were performed to investigate the performance of the proposed ES method in a number of scenarios where traditional methods do not perform well. In addition, the influence of the recombination, learning rates γ, γ', offspring size λ and selection pressure λ/μ are analysed. Comparisons are made against the SAGE algorithm where appropriate.

Unless otherwise stated, the proposed method was simulated using a global recombinant non-isotropic ES with settings: $\mu = 15$, $\lambda = 100$, $n_\sigma = n$, $n_\phi = 0$ and $\rho = \mu$; while the results were averaged over 50 independent runs.

4.4.1 Initialisation

In this section, we compare the level of dependence on accurate initialisation of our method against the SAGE algorithm. This is of interest because both of these algorithms would exhibit excellent performance if accurate initialisation is provided. However, in reality, it is not always possible to obtain accurate initialisation.

The test channel consists of 6 strong paths. Both algorithms were initialised with the channel parameters' true values ± an offset, the amount of offset was increased from 20% to 80%, e.g. if the true value of one path was $[\beta = 1, \theta = 10, \tau = 0.5]$, then one example of an initial value with a 20% error could be $[\beta^{(0)} = 1+0.2, \theta^{(0)} = 10-2, \tau^{(0)} = 0.5-0.1]$.

Figure 4.3 shows the distribution of SAGE and the ES estimates for four levels of errors in initialisation. It is clear that even for strong paths (with good SNR conditions), some degradation is evident at 20% error for SAGE. At 40% error, path estimates become very inconsistent. While at even higher levels, some of the paths were completely undetected. On the other hand, the performance of the ES approach is very slighly degraded with increasing levels of initialisation errors. Even at 80% error (which is equivalent to random initilisation), all of the paths are still clearly distinguiable. In fact, between 40% and 80% errors, the performance of the ES method is approximately the same, and no further degradation is expected even if the amount of errors was increased more.

4.4.2 Estimation and Positional Accuracy

The performance of many existing channel parameter estimation methods is inadequate when the amplitude of the multipath are small. Here, we investigate the performance of the proposed method under these conditions. As well as the positional estimates derived from the LoS measurements.

Figure 4.4 shows the estimation accuracy of the LoS component in a weak LoS channel (where the amplitude of the LoS component is small compared to the other paths).The proposed ES method shows a clear improvement in both estimation accuracy and consistency within the low SNR range. In fact, between 0dB and 5dB, there is an 3-4 fold reduction in standard derivations and >5 fold reduction in mean errors with the proposed method. At 0dB, the ES approach exhibits a maximum mean error of just 5%,

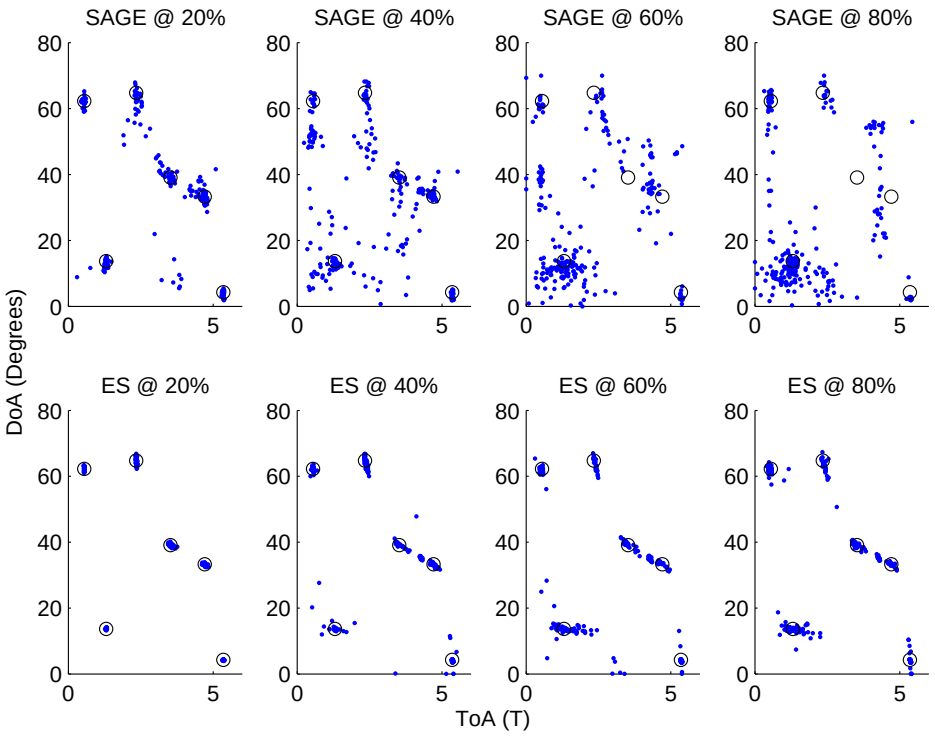


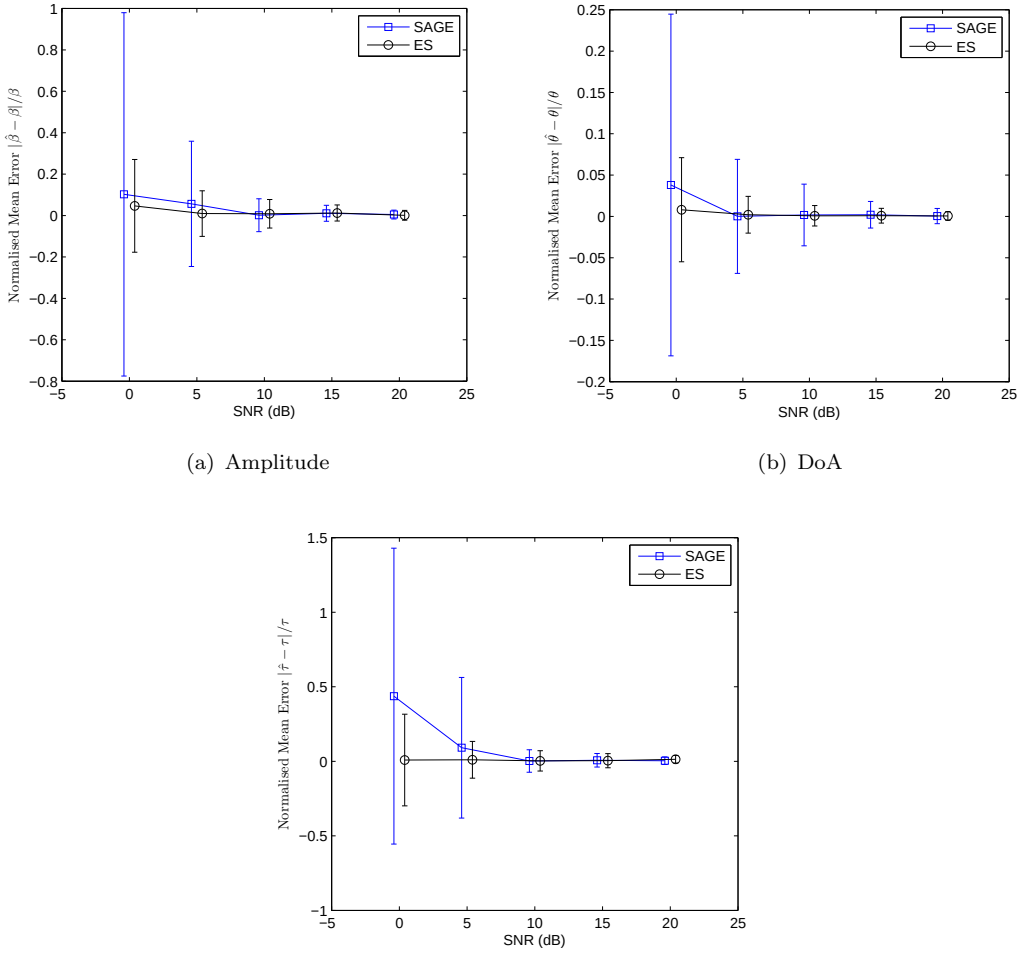
FIGURE 4.3: Distribution of SAGE and ES path estimates for different levels of errors in initialisation. Open circles represent actual path locations. $(15 \odot 100)$-ES was used. SNR=15dB.

and maximum standard deviation of 30% across all estimates. These values are further reduced to 2% and 12% respectively at 5dB.

Now, we can compare the potential positioning accuracies derived from the ToA/DoA estimates of the LoS component in the weak LoS conditions. In Table 2.1, we compared the positional accuracies of the surveyed algorithms using the standard errors in range estimation for both strong and weak LoS channels. The same comparison between the ES and SAGE results are shown in Table 4.2 for the weak LoS channel only, where we can observe considerably smaller standard errors for ES at the low SNR range.

		0dB	5dB	10dB	15dB	20dB
Weak LoS	SAGE	$269.7m$	$122.9m$	$21.0m$	$12.2m$	$6.9m$
	ES	$79.7m$	$31.9m$	$17.6m$	$12.3m$	$6.7m$

TABLE 4.2: Combined standard deviation $(\sigma_{\hat{x}} + \sigma_{\hat{y}})$ in range estimates using joint ToA/DoA estimates from SAGE and ES in a weak LoS channel.



(a) Amplitude

(b) DoA

(c) ToA

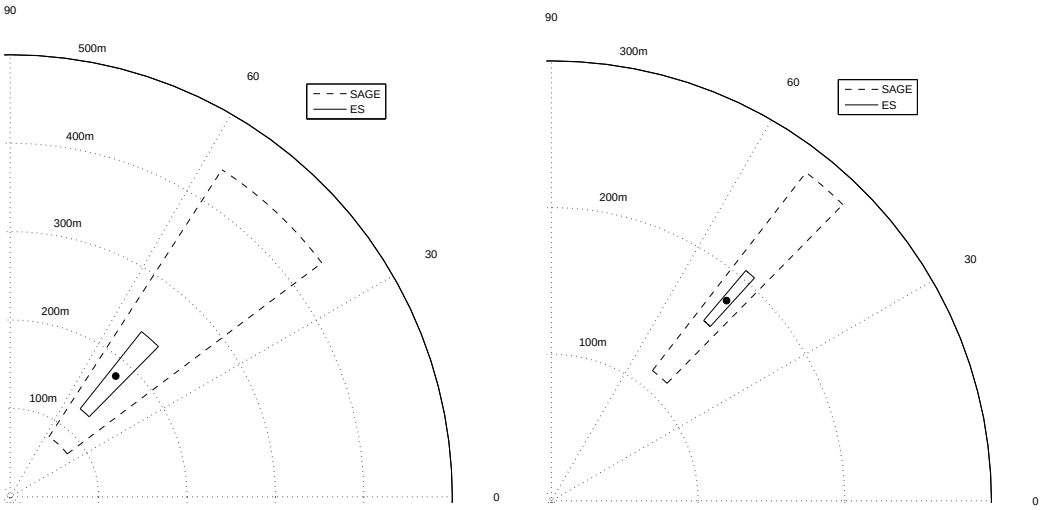
FIGURE 4.4: Estimation accuracy of the weak LoS path. Length of error bar (from top to bottom) represent 2 standard deviation values. For comparative purposes, both algorithms were initialised with the parameter estimates from SAGE's own initialisation routine. $(15 \odot 100)$-ES was used. L=6. Results at each SNR are staggered to avoid obstruction between plots.

However, a more intuitive way of assessing the impact on the positioning accuracies is to calculate the area enclosed by the range and DoA estimate intervals. Using some simple geometry, we define this area as the enclosure between the 2 linear LoPs defined by the mean DoA estimate ($\bar{\theta}$) $\pm$ one standard deviation (σ_θ) and the 2 circular LoPs defined by the mean ToA estimate ($\bar{\tau}$) $\pm$ one standard deviation (σ_τ):

$$\pi(r_\top^2 - r_\perp^2)(\theta_\top - \theta_\perp)/360 \tag{4.27}$$

where $r_\top = (\bar{\tau}+\sigma_\tau)cT$ and $r_\perp = (\bar{\tau}-\sigma_\tau)cT$ denote the upper and lower range estimates; while $\theta_\top = \bar{\theta} + \sigma_\theta$ and $\theta_\perp = \bar{\theta} - \sigma_\theta$ are the upper and lower bearing estimates. c is the

speed of light. Hence, we expect the positioning estimates to be normally distributed at the centre of the enclosure, while the entire area covers 68% of the estimates.



(a) 0dB (b) 5dB

FIGURE 4.5: Positioning accuracies of SAGE and ES in weak LoS conditions.

Figure 4.5 shows the positional areas for 0dB and 5dB for both SAGE and ES. The black dot represents the location of the MS while the measuring BS is located in the origin. We can see that the large ToA bias shown in the SAGE estimates at 0dB is quite evident in the positioning estimates, such that the centre of the enclosure is not the location of the MS. More significantly, the 3-4 fold reduction in SD values in Figure 4.4 shown by ES ToA/DoA estimates at 0dB and 5dB becomes a significant reduction in positioning terms; such that the area of the enclosures in both SNR values are approximately 15 times smaller.

4.4.3 Computational Complexity

The computational complexity of the proposed method is directly dependent on the number of offspring λ that is created. Since the fitness of each of the offspring must be evaluated per generation, this tends to be the computational bottleneck for any ES methods running on single CPU machines. This section compares the computational complexity of the ES approach (using two typical offspring settings, $\lambda = 100$ and $\lambda = 200$, and keeping $\mu = 15$) with the SAGE algorithm.

Figure 4.6 shows the average CPU processing time per iteration/generation of both algorithms as a function of the number of channel multipath. We choose not to compare the time for convergence since the convergence rate of ES is highly dependent on a number of external settings, such as the learning parameters γ, γ' and the selection pressure λ/μ [81].

We can see that the $(15 \odot 100)$-ES is typically much faster than the SAGE algorithm when parameter space dimension is not too big ($L < 16$), while it remains faster when L is large. $(15 \odot 200)$-ES is slower than SAGE only when L is large ($L > 16$), otherwise, it is comparable to the computational complexity of SAGE.

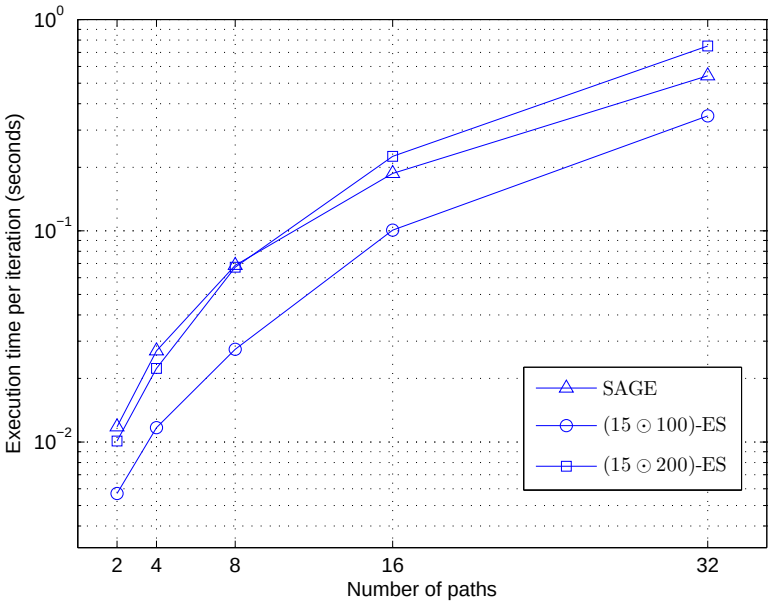


FIGURE 4.6: Computational complexities of SAGE and proposed ES method

4.4.4 Recombination

As previously mentioned, there are two popular recombination operators used in ES, namely discrete recombination $(\mu_{\rho_D} \bullet \lambda)$ and intermediate recombination $(\mu_{\rho_I} \bullet \lambda)$. The current trend of using discrete recombination for object variables and intermediate recombination for strategy parameters originates from the suggestion by Schwefel. Although they are capable of applying to both objective and strategy parameters, no theoretical or experimental criteria for how to choose an appropriate recombination for

each have been shown [89]. In this section, we present a full overview of experimental results according to the types of recombinations applied to both object and strategy parameters.

By using the two recombination operators on both object variables and strategy parameters, we obtain four different combinations: Discrete/Intermediate (DI), Discrete/Discrete (DD), Intermediate/Discrete (ID) and Intermediate/Intermediate (II); of which the first operator indicates the recombination type used for object variables and second for strategy parameters.

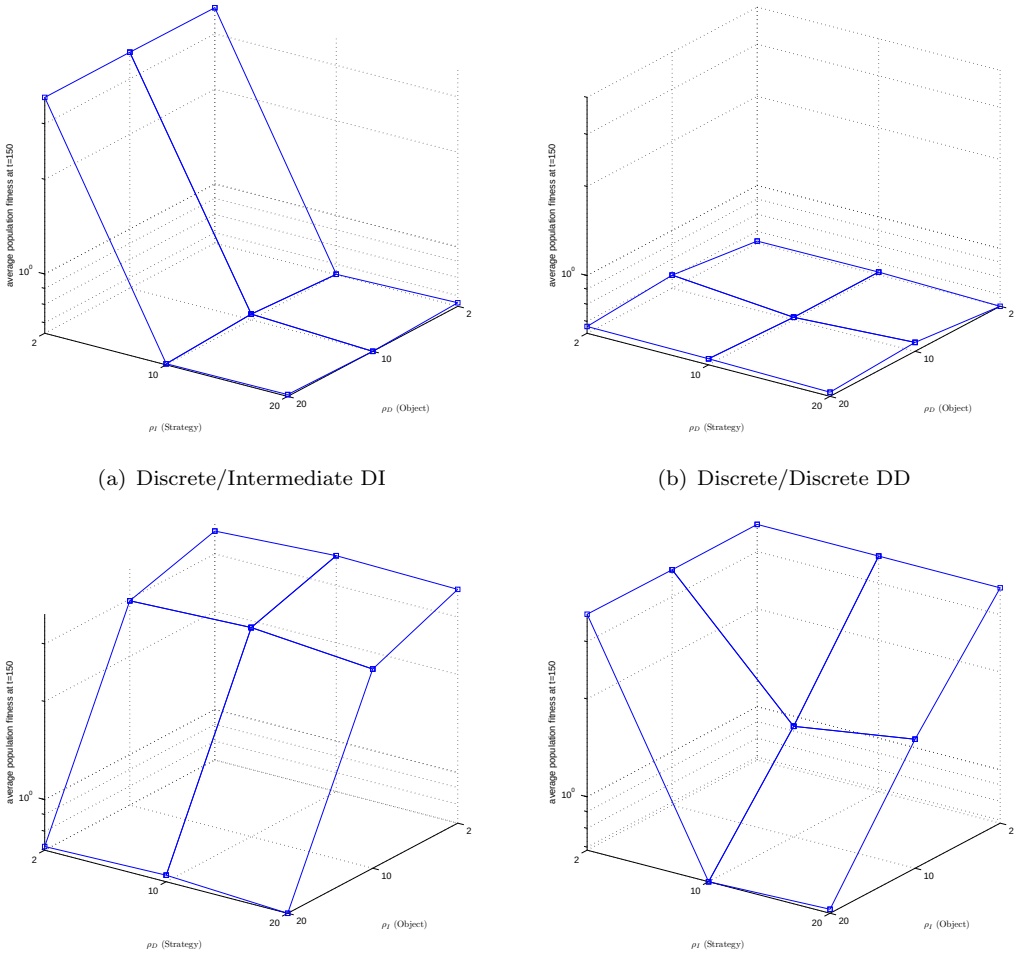
Figure 4.7 shows the average population fitness after 150 generations employing the above mentioned four different combinations while the number of parents ρ that multi-combine are varied from 2 (sexual) to μ (panmictic). We define average population fitness as the average fitness of the entire population of μ candidates at each generation.

Straightaway, we notice that the choice of recombination strategy is highly critical, where many of the different combinations will not lead to ES convergence. In fact, from the results in DI, ID, and II, we can conclude that whenever intermediate recombination is used (regardless of whether it is used for object or strategy recombination), a high value of ρ is required, ideally panmictic recombination where $\rho = \mu$. In addition, from the DI and ID examples, it is apparent that once an appropriate level of ρ has been reached, increasing it further has no clear benefit.

To get a better insight of these results, we look at the searching powers of different types of recombinations. Schwefel has counted the number of different recombination results for the above schemes, which is stated as:

Given a parent population of μ individuals with n-dimensional object variables each, the combinatorially possible different number of recombination results of object variables is:

$$\begin{aligned} \mu + \binom{\mu}{2}(2^n - 2) \quad & \text{discrete recombination} \\ \mu^n \quad & \text{panmictic discrete recombination} \\ \mu(\mu+1)/2 \quad & \text{intermediate recombination} \\ \Big(\mu(\mu+1)/2\Big)^n \quad & \text{panmictic intermediate recombination} \end{aligned} \tag{4.28}$$



(a) Discrete/Intermediate DI

(b) Discrete/Discrete DD

(c) Intermediate/Discrete ID

(d) Intermediate/Intermediate II

FIGURE 4.7: Average population fitness after 150 generations for all possible recombination types. L=4, SNR=15dB.

where the results for this case are shown in Table 4.3. Notice only 210 different combinations are available through the use of sexual intermediate recombination, which is far too small a search space to be of any use. On the other hand, sexual discrete recombination gives 7.8×10^5 different results, which is already much bigger than most usable ES population sizes, such that only a small fraction of the possible results can be tested. Hence, larger recombinant numbers is not expected to be any more beneficial.

	Discrete	Intermediate
$\rho = 2$	7.8×10^5	2.1×10^2
$\rho = 20$	4.1×10^{15}	7.4×10^{27}

TABLE 4.3: Number of combinatorially possible recombination results for sexual/panmictic discrete/intermediate recombination given $\mu = 20$, $n = 12$.

4.4.5 Learning Rate

In this section, the influence of the learning parameters γ and γ' is investigated. It was reported that for a non-recombinant self-adaptive ES, the learning parameters should be chosen as [90]:

$$\begin{aligned} \gamma' &\sim 1/\sqrt{2n} \\ \gamma &\sim 1/\sqrt{2\sqrt{n}} \end{aligned} \tag{4.29}$$

where the constant of proportionality is relatively unrestricted. But we will show that this is not the case for the proposed recombinant ES approach.

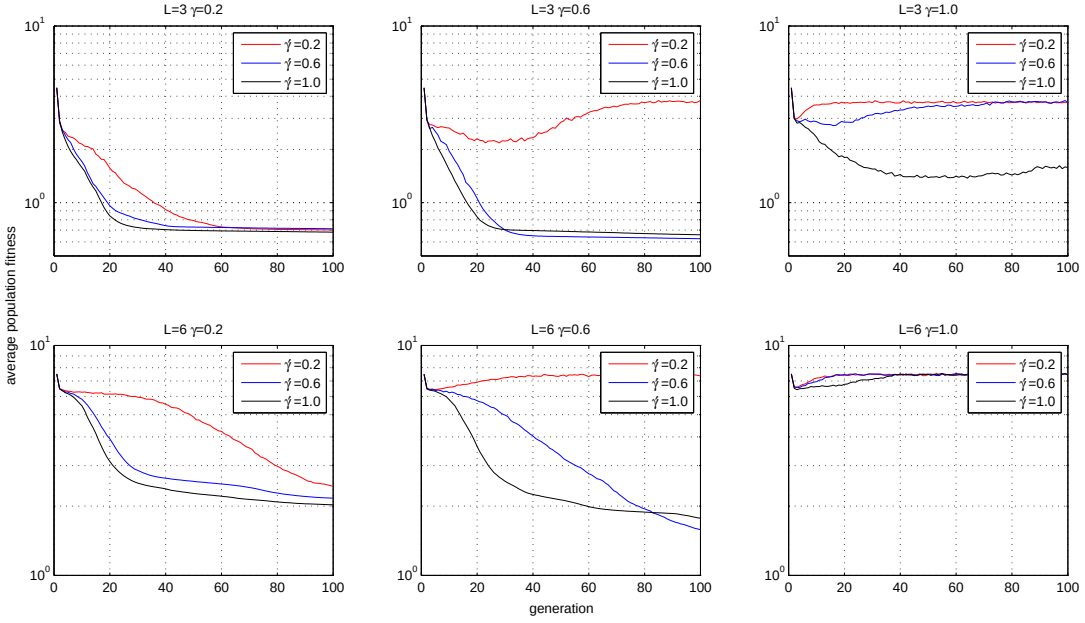


FIGURE 4.8: Average population fitness at each generation as a function of both the global γ' and local γ learning parameters for $L = 3$ (top row) and $L = 6$ (bottom row). SNR=15dB.

Figure 4.8 shows the effects of the learning parameters on the convergence of the proposed method, where we have plotted the average population fitness at each generation using various combinations of γ and γ' values for $L = 3$ (top row) and $L = 6$ (bottom row). We can clearly observe a region in the $\gamma\gamma'$ space where our method does not converge, in particular, when the local learning parameter γ is high. It is also apparent from the low and medium γ settings that $\gamma' > \gamma$ is more preferable. From (4.14), we can see

that the global learning parameter γ' controls the overall amount of mutation applied to the population, while the local parameter γ introduces variation to each individual. Hence, a large γ compared to γ' setting would create instability in the mutation of the standard deviations and the convergence performance is likely to suffer.

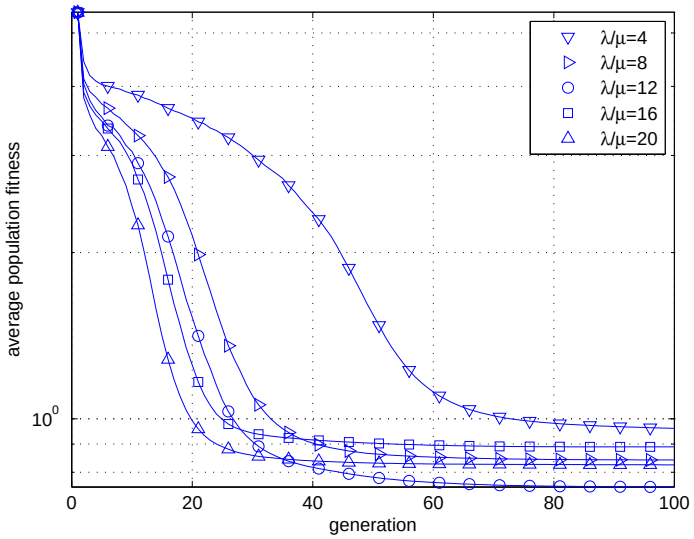
4.4.6 Selection Pressure

In this experiment, we aim to investigate the effect of selection pressure i.e. the λ/μ ratio on the performance of the ES approach. Figure 4.9 shows the average population fitness during evolution for five different selection pressure settings for $L = 4$ (a) and $L = 6$ (b). Selection pressure is varied by keeping the number of offspring λ constant while changing the number of parents μ, as we will show in the next section that the value of λ also affects the ES performance.

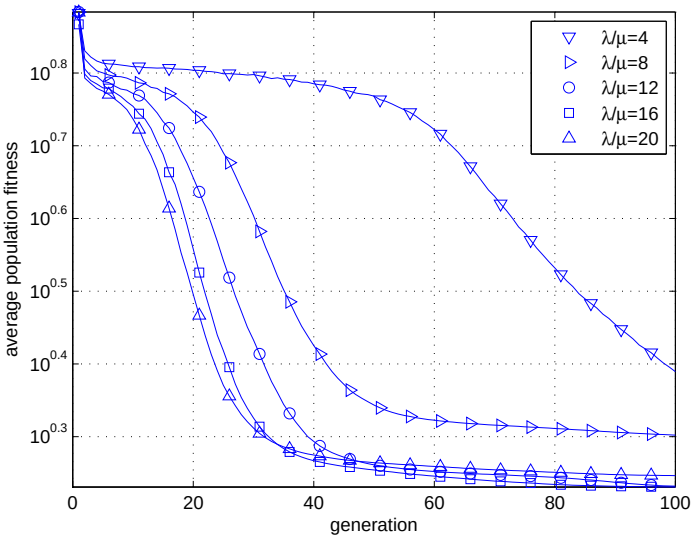
It is quite evident that in both cases ($L = 4, L = 6$), $\lambda/\mu \geq 8$ is required to achieve an acceptable convergence rate, this is consistent with the recommendation based on empirical measurements from [83]. In addition, we can also observe that during the early generations (between 10 and 30), it is quite beneficial to have a high selection pressure since it appears to be proportional to the rate of convergence. However, a very high selection pressure can occasionally lead to prematurely convergence, as in the case of $\lambda/\mu = 16, \lambda/\mu = 20$, and not necessarily towards the global optimum. Hence, the selection pressure controls the characteristics of the ES search. Decreasing μ emphasizes on path-oriented search and convergence velocity, while increasing μ leads to more column-oriented search [85].

4.4.7 Offspring Size

It is well known in ES literature that the estimation performance can be improved by increasing the offspring population size λ. Since promising areas of search space are normally sampled repeatedly by the ES, and there are usually many similar solutions in the population, when the population is large, the influence of noise in evaluating an individual is very likely to be compensated by that of a similar individual [91]. Define R_∞ as the steady state residue distance of the population to the optimum solution in the parameter space, then in the case of the non-recombinative $(1 \odot \lambda)$-ES, R_∞ has been



(a) L=4



(b) L=6

FIGURE 4.9: Average population fitness for five different λ/μ values: $\lambda = 300$, $\mu = \lceil 300/\{4, 8, 12, 16, 20\}\rceil$. SNR=15dB.

proven to decrease with increasing λ. For the global recombinant $(\mu \odot \lambda)$-ES our method utilises, a similar relationship is suggested in [92]:

$$R_\infty \geq \sqrt{\sigma_\eta N \Big/ \Big(\frac{\lambda}{\mu}\frac{1}{\sqrt{2\pi}} \exp\Big[-\frac{1}{2}\Big(\Phi^{-1}\Big(1-\frac{\mu}{\lambda}\Big)\Big)^2\Big]\Big)} \tag{4.30}$$

here, σ_η is the standard deviation of the fitness noise, and Φ^{-1} is the inverse function to the cumulative distribution function of the standard normal variate $\mathcal{N}(0,1)$, which

tends to 1 as λ increases. The divisor term is the asymptotic progress coefficient for large λ [93].

Figure 4.10 shows the average population fitness of the proposed ES approach using five different offspring sizes λ for $L = 4$ (bottom) and $L = 8$ (top). Where in each case, the selection pressure $\lambda/\mu = 10$ is kept constant (hence, μ is increased accordingly with λ).

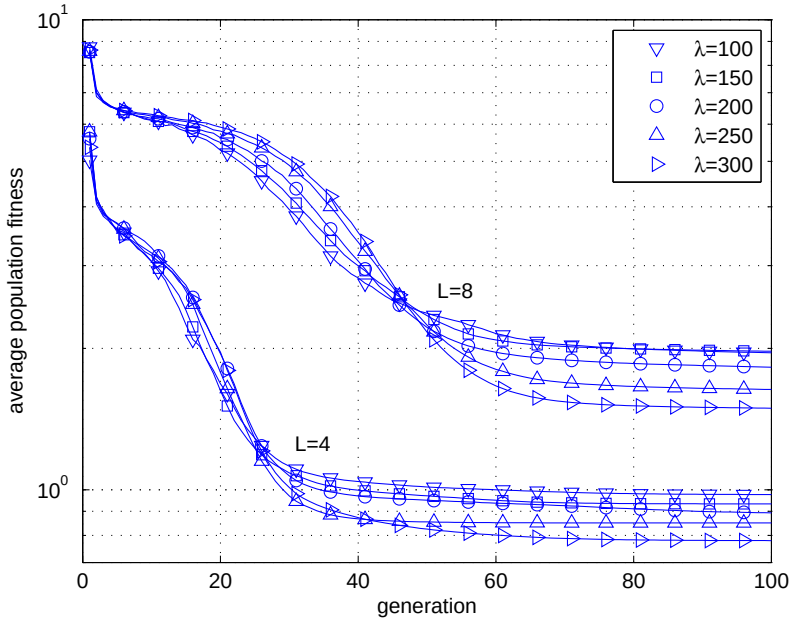


FIGURE 4.10: Average population fitness for five different λ values. In each case, selection pressure is kept constant at $\lambda/\mu = 10$. SNR=15dB.

As expected, we can observe a performance gain when the offspring size is increased. This is a useful property since one can achieve a desired accuracy of estimation by just increasing the offspring size alone, provided the increase in computational complexity is manageable and justified. The performance gain in the case of $L = 4$ is less compared to $L = 8$, which suggests even the lowest setting of $\lambda = 100$ is sufficient to achieve near optimum performance in this case, and further increments in λ is probably unjustified.

4.5 Summary

We have proposed a joint channel parameter estimation technique using an ES approach. The proposed method is shown to be extremely robust to initialisation and low SNR

environments. In addition, the computational complexity of the proposed method is demonstrated to be less than that of the traditional iterative ML approach. The vast search powers of a population based intelligent optimisation method opens the possibility of jointly estimating other important parameters, e.g. the number of channel paths and channel gains.

Chapter 5

Joint Detection and Estimation of Channel Parameters Using Hierarchically Organised Evolution Strategy

5.1 Introduction

One common assumption in most research areas within channel parameter estimation is that the number of channel multipath L is known a priori. However, in practice this number must be estimated first; either by a separate mechanism or jointly estimated with the channel parameters; the latter is sometime known as the joint detection and estimation problem. The joint approach is considered the more optimum of the two, but at the cost of additional computational complexity. We have demonstrated in Chapter 2 the dependence of these channel parameter estimation methods on accurate knowledge of L, where even for small errors in $\hat{L}$, the estimate of the true multipath number, can lead to significant performance degradation. In addition, if $\hat{L} < L$, these estimation methods will typically detect only the strongest $\hat{L}$ multipath components, thus omitting the remaining $L - \hat{L}$ (weakest) paths in the process. This is clearly unacceptable if the LoS component happens to be weak.

There exists numerous research in the field of joint detection and channel parameter estimation, especially in the field of direction finding with active antenna array [94]. The methods in [95, 96] jointly optimise Rissanen's Minimum Description Length (MDL) principle and the Maximum Likelihood (ML) estimator of the channel parameters, which is computationally prohibitive. Sub-optimal stochastic methods based on Evolutionary Algorithms have also been proposed, where a population with variable sized candidate solutions are evolved through specially adapted mechanisms of recombination and mutation. Within this population, candidate solutions representing the best choice of L and the corresponding channel parameters as measured by fitness are selected for reproduction. Recombination where two parents reproduce by exchanging different sized segments is employed in [97, 98], producing in the processes two new offspring with different length. While in [99, 100], the mutation process randomly add/remove bits from each offspring. However, these approaches are unstable and could easily stagnate, since only a limited number of candidate solutions are available at each generation, which is shared between a number of (possibly high) problem dimensions.

In this chapter, we propose an efficient sub-optimal method of jointly detecting and estimating the channel parameters based on Hierarchically Organised Evolution Strategies. As the name suggests, the proposed method uses multiple hierarchies of ES to simultaneously optimise for L and the channel parameters. During the evolution process, a set of $\hat{L}$ values are explored within the Upper Level Strategy (ULS); while the Lower Level Strategy (LLS), which consists of multiple sub-populations, searches for the channel parameters corresponding to the set of $\hat{L}$ values as chosen by the upper level.

5.2 Hierarchically Organised Evolution Strategies

Multi-population Evolutionary Algorithms have been invented several times to solve difficult optimisation problems. In the field of Evolution Strategies, Rechenberg proposed the Hierarchically Organised ES, also referred to as Meta-ES or Nested ES as early as 1978 [101]. Essentially, the ES is parallelised by dividing a large population into several sub-populations - each of them "living" on a different process. After one or more generations some individuals migrate to a neighbouring sub-population, or in other words: the distributed optimisation processes exchange information during the search

[102]. From this point onwards, we shall use the simpler term Meta-ES when referring to Hierarchically Organised ES.

A Meta-ES can be described by generalising formally the ES bracket notation to [103]:

$$\Big[\underbrace{\mu'_{\rho'} \bullet \lambda'}_{\text{ULS}} \underbrace{(\mu_\rho \bullet \lambda)^\chi}_{\text{LLS}}\Big] \tag{5.1}$$

where λ' instances of the standard $(\mu_\rho \bullet \lambda)$-ESs in (4.5) runs independently (in parallel) for a number of χ generations in isolation. The idea is to have these λ' independent populations furnished with different search properties, such as different mutation strengths, different search spaces partitions, or other object/strategy parameters which are normally fixed in advance. After χ generations, the amount of progress that has been made on the various populations are compared. The search properties of the most successful $\mu'_{\rho'}$ populations are subject to variation, and new set of species is set up and run with those new search properties. Thus, the evolutionary optimisation happens on two levels: the search space of the LLS is that of the optimisation problem at hand; that of the ULS is the search property space of the LLS. Variation and selection are used on both levels [104].

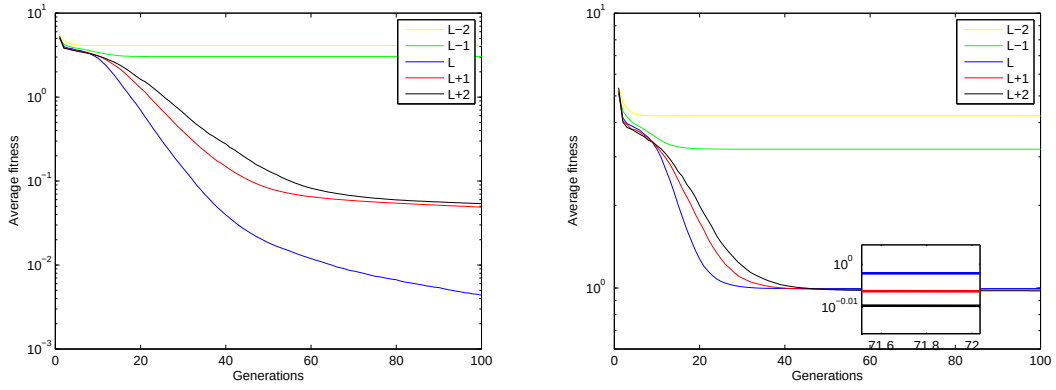
Meta-ES has traditionally been used as an alternative to the mutative mechanism in step length adaptation, where a single standard deviation σ is modified using two subpopulations each utilising $\sigma \times \alpha$ and σ/α respectively, where α is a pre-defined constant normally set to $\alpha = 1.3$. It has been shown that in mutative self-adaptation, opportunistic individuals that make short steps are often rewarded, hampering long term progress [104]. In other words, opportunistic individuals can increase their likelihood of short term success by using below-optimal step lengths. The use of the isolation period χ in between step length adaptation in Meta-ES aids long term progress rather than short term success, since the ability of mutation strength to generate successful steps is much easier to judge after several time steps than after a single one.

5.3 Joint Detection and Estimation of Channel Parameters

It is very difficult to perform the joint detection and estimation of channel parameters using a standard $(\mu \bullet \lambda)$-ES with deterministic selection as defined in 4.3.4. For a given optimisation problem in noisy environment, an ES population will converge to a better fitness if the problem dimension is over-estimated. Figure 5.1 shows the average population fitness as defined by (5.2) during an evolution process for different degrees of over- and under-estimation of the number of multipath L under ideal (no noise in the received data $\mathbf{y}(t)$) and noisy conditions,

$$\varepsilon(\mathbf{y};\boldsymbol{\omega}) \triangleq \int_0^{T_o} \left| \mathbf{y}(t) - \sum_{\ell=1}^{L'} \hat{\beta}_\ell \mathbf{a}(\hat{\theta}_\ell) g(t-\hat{\tau}_\ell) \right|^2 dt \tag{5.2}$$

where for an under-estimated problem dimension, the population consists of candidate solutions representing case of $L' < L$; similarly for the over-estimated case: $L' > L$.



(a) Ideal (b) Noisy

FIGURE 5.1: Average population fitness at each generation from a standard $(100 \odot 15)$-ES algorithm when the number of path is under/over estimated under ideal and noisy conditions. The labels indicate the amount of deviation from actual number of path $(L = 3)$. SNR=15dB for the noisy case.

We can observe, under ideal conditions (Figure 5.1(a)), the value of ε in (5.2) is lowest only when the correct value of L is used. In fact, if the optimisation process carried on indefinitely, the steady state ε value when L is correct converges to zero. When $\mathbf{y}(t)$ contains noise (Figure 5.1(b)), we can still observe a clear fitness distinction when L is under-estimated, but this is not the case for over-estimation. It is not difficult to see

that steady state ε value in the noisy case when L is correct is simply the energy of the noise contained in $\mathbf{y}(t)$, i.e. $\int_0^{T_o} |\boldsymbol{\eta}(t)|^2 dt \approx E_o T_o$. In fact, the magnified sub-figure shows having a higher problem dimension actually improves the steady state fitness beyond this value. In other words, if we were to estimate L by measuring the fitness while incrementing $\hat{L}$ (initialised at a small value, i.e. $\hat{L}^{(0)} = 1$), our estimate would approach and then surpass the actual value. More importantly, the fitness of the population utilising the correct value of L is only the lowest for a very small proportion of the evolution process; in this case, between 10-40.

This behaviour is a consequence of the noise associated with the fitness space, i.e. there is noise involved in the process of evaluation of the fitness function. Hence, when evaluating the fitness of a candidate solution $\mathbf{x}$, it is not the ideal fitness $f(\mathbf{x})$ that we obtain, but a perceived fitness that is normally distributed with mean $f(\mathbf{x})$ and standard deviation $\sigma_f(\mathbf{x})$ [105]. This form of noise has been termed fitness noise. Since the perceived fitness may be different to the ideal fitness, the fitness noise is capable of deceiving the selection mechanism as it can lead to inferior candidate solutions being selected based their perceived fitness while superior ones are discarded [106].

Nevertheless, we can observe a few other properties which are potentially useful in the estimation of L:

- The convergence rate of the fitness values are lower when $\hat{L}$ is high; more importantly, it is much slower when $\hat{L}$ is over-estimated. For the example in Figure 5.1(b), over-determined populations converged in around 40 generations compared to 20 generations when the correct value is used.
- The improvement in steady state fitness value from increasing $\hat{L}$ is substantially reduced when $\hat{L}$ becomes over-estimated.

It is possible to jointly estimate L and the channel parameters $\boldsymbol{\omega} \triangleq [\boldsymbol{\omega}_1 | \boldsymbol{\omega}_2 | \ldots, | \boldsymbol{\omega}_L]$ by observing any/all of the above mentioned properties, during the parallel execution of $L_{\max}$ copies of the standard ES, where $L_{\max}$ is the pre-specified upper limit on the number of multipath, i.e. $1 \leq L \leq L_{\max}$. However, this is only feasible for very small numbers of L. In addition, since we have no prior knowledge of the number of paths, then conservative estimates of $L_{\max}$ would need to be relatively large to guarantee one of the executing ES algorithms will be using the correct L.

Employing Meta-ES allows us to compare the population fitnesses of λ' instances ($1 < \lambda' < L$) of the standard ES each employing a different $\hat{L}$ value. Therefore, the problem is now to formulate an evolution strategy that explores different values of $\hat{L}$ while at the same time evolving multiple populations of candidate solutions to estimate $\mathbf{\Omega}$; by exchanging of information between the λ' instances of the ES algorithm.

From initial inspection, it appears that the first property is the easiest to observe, especially the convergence rate during the early generations. However, using only λ' instances of the standard ES within the lower level, then the ULS must alternate the set of $\hat{L}$ values which the lower level ES explores, hence we cannot *continuously* observe the fitness for any chosen set of $\hat{L}$ values; only a snapshot of χ generations at a time. After χ generations, it is possible that an entirely different set of $\hat{L}$ values are chosen for exploration. Essentially, evolving multiple ES instances for χ generations at a time, presents us with a vertical snapshot of Figure 5.1 at regular intervals (albeit of much fewer $\hat{L}$ values, since λ' should be relatively small). Hence provided we can navigate through the possible $\hat{L}$ values while maintaining sufficiently accurate channel parameter estimates, it is possible to utilise the third property as a guide towards the optimum $\hat{L}$ value.

While it is possible for us to utilise the standard Meta-ES approach and optimise both $\hat{L}$ and σ via the ULS, as demonstrated in [103] where a $[1 \odot 4(\mu_I \odot \lambda)^{\chi}]$-ES is used to adapt both the optimal population size μ and step length σ. It is more efficient in this case to optimise only $\hat{L}$ at the upper level and keep the mutative self-adaptation mechanism at the lower level. The reasons are three fold:

- From results in previous chapter, there were no serious issues with premature convergence and stagnation from using mutative self-adaptation, therefore there is no need to change the adaptation mechanism.
- Optimisation at the ULS is designed for one dimensional variables, i.e. a single mutation strength σ only (as suppose to a separate σ_i for each channel parameter), hence only isotropic mutations are possible (see previous chapter for details on isotropic/non-isotropic mutations).
- In order to optimise for a single variable in the ULS, at least two instances of the lower strategies are required, and a minimum of four is necessary to handle two

variables. Hence, λ' should be kept to a minimum to reduce the computational burden.

Thus, our proposed joint detection and channel parameter estimation method is based on a $[1 \odot 3(\mu \odot \lambda)^{\chi}]$-ES, where 3 consecutive $\hat{L}$ values are explored at any time. Traditional deterministic selection based on fitness ordering in the upper level is modified to a type of conditional selection. In the following description, certain syntax and symbols of the LLS are identical to Chapter 4, hence will not be re-defined here.

5.3.1 Upper Level Strategy

The estimate of the number of multipath $\hat{L}$ is the single parameter that is adapted by the ULS. Thus, while the LLS faces an $3 \times \hat{L}$-dimensional optimisation problem, that of the upper level strategy is one-dimensional. As a consequence, a very simple algorithm can be used. Algorithm 5.1 shows the pseudocode of the proposed $[1 \odot 3(\mu \odot \lambda)^{\chi}]$-ES algorithm in which we explore three consecutive $\hat{L}$ values, i.e. $[\hat{L}-1, \hat{L}, \hat{L}+1]$ during each generation of the ULS. In the following descriptions, we denote $\mathbf{\Omega}^{(k)} \in \mathbb{R}^{\mu \times 3\hat{L}}$ and $\mathbf{\Sigma}^{(k)} \in \mathbb{R}^{\mu \times 3\hat{L}}$ as the matrices of channel parameter estimates and mutation strengths at the kth ULS generation. Their structures are identical to (4.7) and (4.8), apart from the width which is defined by $\hat{L}$ instead. In addition, $\mathbf{\Omega}_p^{(k)}$, $\mathbf{\Sigma}_p^{(k)}$ and $\varepsilon_p^{(k)}$ for $p \in \{1, 2, 3\}$ denotes the matrix of channel parameter estimates, matrix of mutation strengths and average population fitness from the pth instance of the lower strategy at the kth ULS generation respectively. The key points are summarised below:

- Our deterministic approach for exploring the L domain is based on two simple and concrete rules for changing the current estimate of $\hat{L}$.
    1. *Looking forward*: if there is a significant improvement in fitness from the next higher value of $\hat{L}$, then it is very likely we are still under-estimating $\hat{L}$, then we should increase $\hat{L}$ to the higher value and compare its neighbouring values.
    2. *Looking back*: if over the last two $\hat{L}$ increments, we have observed insignificant fitness improvement, then we have most likely over-estimated $\hat{L}$.
    3. In all other cases, we cannot be certain of current estimate, we keep $\hat{L}$ unchanged and compare again after the next generation.

Algorithm 5.1 $[1 \odot 3(\mu \odot \lambda)^{\chi}]$-ES

Require: $2 \leq \hat{L} \leq L_{\max} - 1$

1: $k := 0$
2: *Initialise* $\left([\mathbf{\Omega}^{(k)}]_{i,j} \in U(0,1), [\mathbf{\Sigma}^{(k)}]_{ij} \in U(0,0.3), \hat{L} \in U(1,\ldots,\hat{L}_{max})\right)$
3: **while** not terminated **do**
4: $\left[\varepsilon_1^{(k)}, \mathbf{\Sigma}_1^{(k)}, \mathbf{\Omega}_1^{(k)}\right] := LLS\left(\mu, \lambda, \chi, \mathbf{\Sigma}^{(k)}, \mathbf{\Omega}^{(k)}, \hat{L} - 1\right)$
5: $\left[\varepsilon_2^{(k)}, \mathbf{\Sigma}_2^{(k)}, \mathbf{\Omega}_2^{(k)}\right] := LLS\left(\mu, \lambda, \chi, \mathbf{\Sigma}^{(k)}, \mathbf{\Omega}^{(k)}, \hat{L}\right)$
6: $\left[\varepsilon_3^{(k)}, \mathbf{\Sigma}_3^{(k)}, \mathbf{\Omega}_3^{(k)}\right] := LLS\left(\mu, \lambda, \chi, \mathbf{\Sigma}^{(k)}, \mathbf{\Omega}^{(k)}, \hat{L} + 1\right)$
7: **if** $\varepsilon_1^{(k)} - \varepsilon_3^{(k)} < \varepsilon_\Delta$ **then** {decrement $\hat{L}$ by 1}
8: $\hat{L} := \hat{L} - 1$
9: $\mathbf{\Omega}^{(k)} := \mathbf{\Omega}_1^{(k)}$
10: $\mathbf{\Sigma}^{(k)} := \mathbf{\Sigma}_1^{(k)}$
11: **else if** $\varepsilon_2^{(k)} - \varepsilon_3^{(k)} > \varepsilon_\Delta$ **then** {increment $\hat{L}$ by 1}
12: $\hat{L} := \hat{L} + 1$
13: $\mathbf{\Omega}^{(k)} := \mathbf{\Omega}_3^{(k)}$
14: $\mathbf{\Sigma}^{(k)} := \mathbf{\Sigma}_3^{(k)}$
15: **else** {$\hat{L}$ unchanged}
16: $\mathbf{\Omega}^{(k)} := \mathbf{\Omega}_2^{(k)}$
17: $\mathbf{\Sigma}^{(k)} := \mathbf{\Sigma}_2^{(k)}$
18: **end if**
19: check for termination
20: $k := k + 1$
21: **end while**

This type of conditional selection is a popular approach for dealing with fitness noise in EAs, where typically, an offspring individual is accepted if and only if its fitness is better than that of its parents by at least a predefined threshold [91].

- In lines 4-6, three runs of the lower level $(\mu \odot \lambda)$-ES are conducted in parallel. The runs last for χ LLS generations each and all of them use the current mutation strength $\mathbf{\Sigma}^{(k)}$ and channel parameter estimates $\mathbf{\Omega}^{(k)}$ from the ULS as their initial search values. One instance uses the current multipath number estimate $\hat{L}$, while the remaining two tries $\hat{L} - 1$ and $\hat{L} + 1$ respectively. At the end of χ LLS generations, the final search points, mutation strengths and the fitness value of the final search points are saved.

- In line 7-10, the fitness level of the strategies employing the lower $(\hat{L}-1)$ and upper $(\hat{L} + 1)$ values are compared. If the improvement in fitness from an increment of two channel paths are less than a pre-defined criteria ε_Δ, which we call the *upper level fitness criteria*, then it is most likely we have over-estimated $\hat{L}$, and $\hat{L}$ should

be smaller. Consequently, the search parameters $\left[\mathbf{\Omega}^{(k)}, \mathbf{\Sigma}^{(k)}\right]$ of the ULS are set to the parameters from the ES instance with the lower $\hat{L}$ value.

- In line 11-14, if the fitness gain from adding one additional path is greater than ε_Δ, then it is most likely we have underestimated $\hat{L}$, and that further increments to $\hat{L}$ could be beneficial. Consequently, the search parameters $\left[\mathbf{\Omega}^{(k)}, \mathbf{\Sigma}^{(k)}\right]$ of the ULS are set to the parameters from the ES instance with the higher $\hat{L}$ value.

- The requirement at beginning of algorithm is enforced to make sure at each generation, the LLS are using three *consecutive* $\hat{L}$ values. Hence, if the best $\hat{L}$ value becomes 1, then the strategy exploring $\hat{L}-1$ is pointed to $\hat{L}+2$ instead. A similar approach is taken when p becomes $\hat{L}_{\max}$, i.e. the strategy using $\hat{L}+1$ is changed to $\hat{L}-2$.

5.3.2 Lower Level Strategy

For a chosen number value of $\hat{L}$, the lower level strategy $LLS(\bullet)$ (Algorithm 5.2) optimises for the best channel parameter estimates $\mathbf{\Omega}^{(k)}$, assuming a problem dimension of $3 \times \hat{L}$. The function is identical to the non-isotropic global recombinant $(\mu \odot \lambda)$-ES employed in Chapter 4. However, since the three instances of the LLS are all initialised with the same set of starting search values $\left[\mathbf{\Sigma}^{(k)}, \mathbf{\Omega}^{(k)}\right]$, but different $\hat{L}$ values, therefore the sizes of the matrices $\left[\mathbf{\Omega}^{(k)}, \mathbf{\Sigma}^{(k)}\right]$ must be modified according to $\hat{L}$. This is shown in lines 1-8. In Line 9, the standard $(\mu \odot \lambda)$-ES is run for χ generations to obtain a temporary estimate of the channel parameters.

Algorithm 5.2 Function: $\text{LLS}\left(\mu, \lambda, \chi, \mathbf{\Sigma}, \mathbf{\Omega}, \hat{L}\right)$

1: $\delta n := length(\mathbf{\Sigma}) - 3 \times \hat{L}$
2: **if** $\delta n > 0$ **then**
3: remove last δn columns of $\mathbf{\Sigma}$ and $\mathbf{\Omega}$
4: **end if**
5: **if** $\delta n < 0$ **then**
6: add $|\delta n|$ columns to $\mathbf{\Sigma}$ with samples drawn from $U(0,1)$
7: add $|\delta n|$ columns to $\mathbf{\Omega}$ with samples drawn from $U(0,0.3)$
8: **end if**
9: $[\varepsilon', \mathbf{\Sigma}', \mathbf{\Omega}'] := ES(\mu, \lambda, \mathbf{\Sigma}, \mathbf{\Omega}, \hat{L})$ (execute for χ generations)
10: **return** $\varepsilon', \mathbf{\Sigma}', \mathbf{\Omega}'$

5.4 Simulations

In this section, we investigate empirically the performance of the proposed meta-ES method in MATLAB. Since the channel parameters are estimated via the LLS, which is identical to the $(\mu \odot \lambda)$-ES presented in the previous chapter, hence the estimation performance of the channel parameters $\boldsymbol{\omega}$ remains unchanged. Here, we will concentrate primarily on the estimation of the channel multipath number L.

The proposed meta-ES method utilises two additional predefined strategy parameters that will be influential to the estimation performance; the isolation amount χ and the upper level fitness criteria ε_Δ. We will start by investigating the level of effect these parameters have on the proposed method, this is followed by some general results in multipath environments.

As mentioned earlier, the average steady state fitness can no longer be used as a performance measure, since a lower fitness in this case does not necessarily equates to better estimates. Strictly speaking, the long term behaviour of an algorithm is of utmost interest in convergence analysis: One wants to know about the number of iterations that are necessary to reach (or come close to) the optimum (or the optimiser). However, the full dynamics of ES turns out to be intractable even in the simplest cases [107]. Fortunately, due to its simple Markovian character, much of the dynamics of ES can be recovered by observing statistical parameters and their expected rates of change from one generation to another; one such performance measure is the expected quality gain [108]:

$$\bar{Q} = E\Big\{ \langle F^{(k)} \rangle_\mu - \langle F^{(k-1)} \rangle_\mu \Big\} \tag{5.3}$$

which is a measure of the expected difference in fitness of the population of candidate solutions at consecutive time steps. Here, $\langle F^{(k)} \rangle_\mu$ denotes the average fitness of μ candidate solutions at the kth generation. We can estimate $\bar{Q}$ by calculating $(\langle F^{(1)} \rangle_\mu - \langle F^{(k_{\text{end}})} \rangle_\mu)/k_{\text{end}}$ for each run then average over multiple independent runs.

5.4.1 Choice of Upper Level Fitness Criteria

On initial inspection, the choice of the upper level fitness criteria ε_Δ appears to be central to the workings of the proposed method, since ε_Δ controls the freedom in which

the ULS explores the $\hat{L}$ search space. Figure 5.2 shows the estimated quality gain and the average final $\hat{L}$ values for 3 different multipath channels as a function of ε_Δ. It is not surprising that there is a strong correlation between the expected quality gain and the average $\hat{L}$ value; both are at the optimum for $\varepsilon_\Delta < 0.2$. Within the higher range of ε_Δ values ($\varepsilon_\Delta > 0.3$), it becomes increasingly difficult for the algorithm to explore different values of $\hat{L}$, and the method fails to converge to the actual value. However, what is more significant is that the optimum range of ε_Δ values are quite similar for different problem dimensions, i.e. between 0.1 and 0.2 the proposed method works well for all the tested L values. Hence, provided ε_Δ is kept reasonably small, the proposed meta-ES method is expected to perform well for all channel orders.

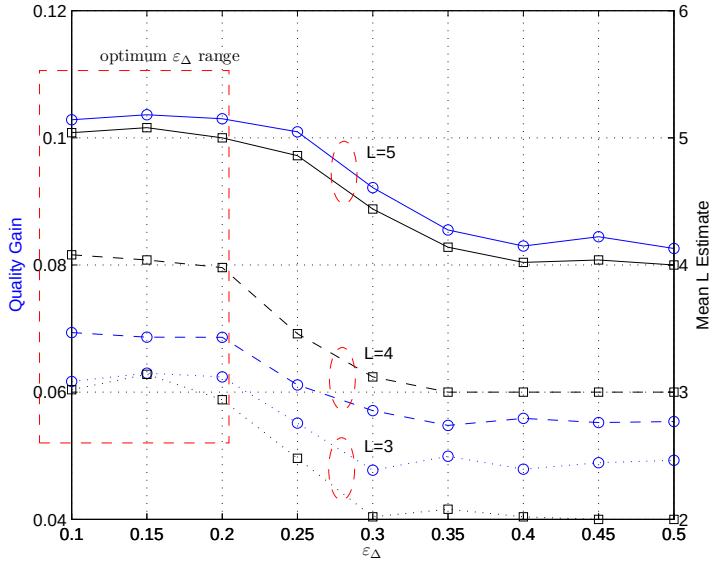


FIGURE 5.2: Estimated expected quality gain (circles) and average final L estimate (squares) as a function of ε_Δ. SNR=15dB

5.4.2 Effects of Isolation Period

It is well known that the isolation period χ is a critical parameter in step length adaptation using Meta-ES. In these cases, chosen periods that are too short can result in sub-optimal step lengths and the same type of problems that effect mutative self-adaptation. Isolation periods that are too long prevent fast adaptation of the step length and slow down convergence [101]. This issue is so important that the idea of adding another level to the hierarchy of ES, for the sole purpose of optimising the length of isolation period,

has been proposed in [104]. We will see that a similar role is played by the isolation period in our proposed method.

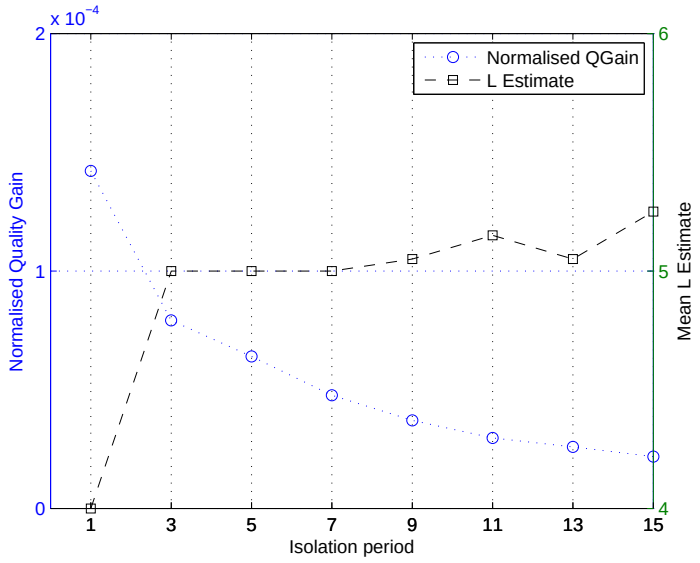
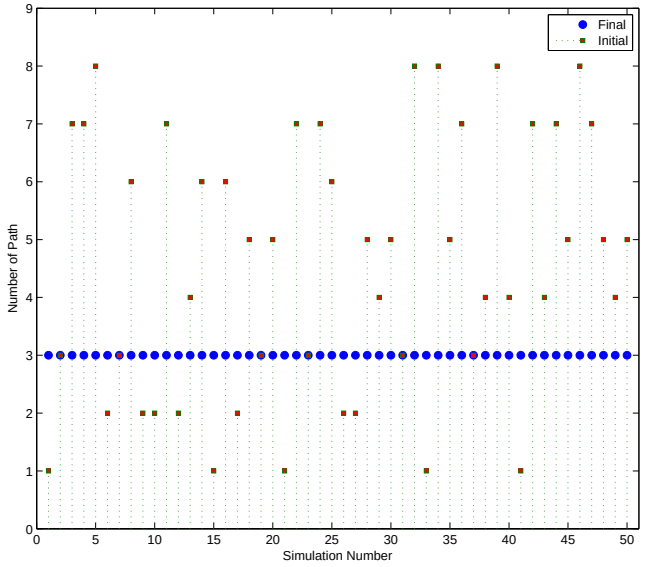
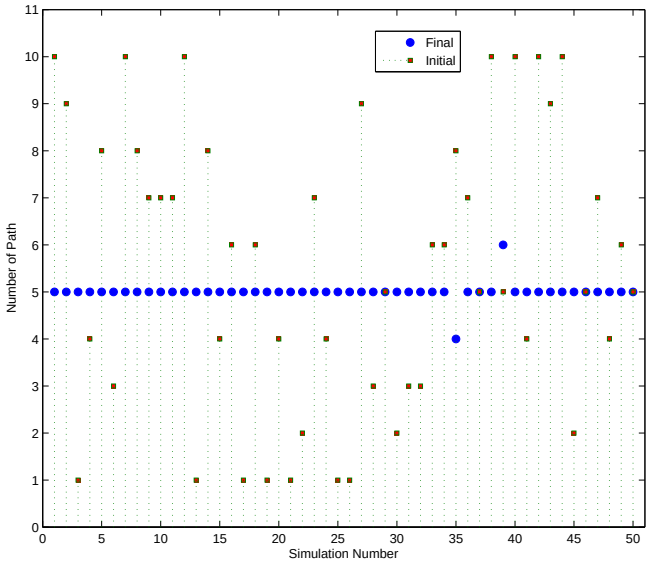


FIGURE 5.3: Normalised quality gain (circles) and average final L estimate (squares) as a function of the isolation period χ. Actual $L = 5$. SNR=15dB

If we intend to vary the isolation period, the expected quality gain $\bar{Q}$ must be normalised prior to comparison. We define the normalised quality gain per evaluation of the objective function as $\bar{Q}/(\chi\lambda'\lambda)$, which relates the gain in fitness to the amount of computational effort invested to achieve it. Here, the number of objective function evaluations per time step is $\chi\lambda'\lambda$. The first thing we notice from Figure 5.3 is that the normalised quality gain decreases monotonically with increasing isolation period, and the algorithm is operating at its highest efficiency at $\chi = 1$. However, this value of χ represent too short a time interval during which the LLS can make any significant progress in, and the algorithm cannot converge to the correct value of L. More interestingly, once the isolation period is greater than 1, the detection performance of the proposed method is relatively stable; such that increasing χ from 3 to 15 has only resulted in an average $\hat{L}$ deviation of less than 0.2. Hence, the only concrete bond for the isolation period to guarantee good $\hat{L}$ estimate is $\chi > 1$; but considering large χ values are detrimental to the overall operating efficiency, smaller values are recommended.



(a) L=3



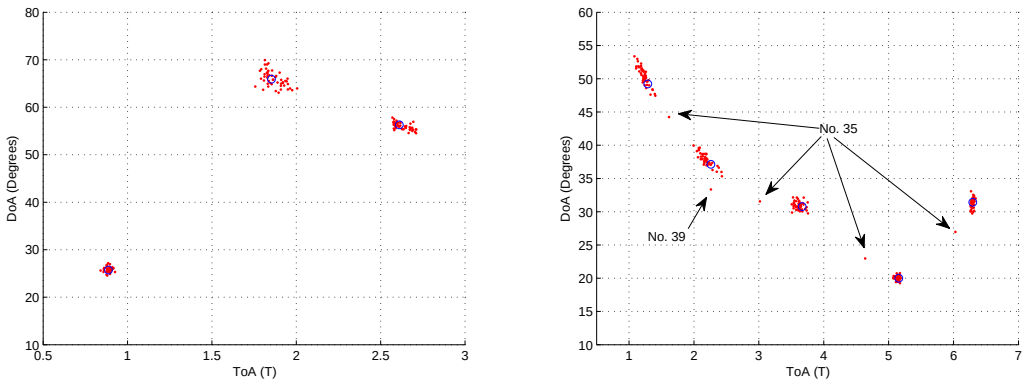
(b) L=5

FIGURE 5.4: Initial and final L estimates for 50 independent runs. Initial estimates for $\hat{L}$ are integer samples drawn from $U[1, 8]$ and $U[1, 10]$ for $L = 3$ and $L = 5$ respectively. SNR=15dB, $\varepsilon_\Delta = 0.15$ and $\chi = 5$ for all cases.

5.4.3 Joint Detection and Estimation

Figures 5.4 and 5.5 show the joint detection and estimation results for 2 multipath channels, where the number of multipath are $L = 3$ and $L = 5$ respectively. In Figure 5.4, the initialised and final $\hat{L}$ values from 50 independent trials are plotted. For each

simulation (numbered by the x-axis), its initial and final $\hat{L}$ values are shown by the vertical positions of the red squares and blue circles respectively. It is quite evident that our method places no constraint on the initial $\hat{L}$ estimates; it is able to estimate the correct value of L regardless of the initialised values in almost all cases. Figure 5.4(b) shows only two cases of errors in simulations No. 35 and 39, where a small under- and over- estimation of $\hat{L}$ occurred. In both channels, we have chosen $\hat{L}_{\max}$ - the upper limit on the $\hat{L}$ search space - to be greater than $2L$. This is just to demonstrate the versatility of the proposed method, and it is by no means a requirement. Just as important, Figure 5.5 shows that the distributions of corresponding channel parameter estimates, which are also randomly initialised, are all close to actual. We observe two sets of slightly erroneous data in the case of $L = 5$ in Figure 5.5(b), which correspond to the cases of when $\hat{L}$ was under- and over-estimated by 1 in simulations No. 35 and 39. Notice how under-estimating $\hat{L}$ is much worse since non of the 4 paths were estimated accurately; on the other hand, over-estimation produces accurate estimates of all 5 paths, plus 1 addition random path.



(a) L=3 (b) L=5

FIGURE 5.5: Distribution of channel parameter estimates from 50 independent runs. Initial parameter estimates - elements of matrix $\mathbf{\Omega}$ are samples drawn from $U(0,1)$ for all cases. Circles represent actual multipath locations. SNR=15dB, $\varepsilon_\Delta = 0.15$ and $\chi = 5$ for all cases.

5.5 Summary

We have proposed a joint detection and channel parameter estimation method based on Hierarchically Organised Evolution Strategies. The proposed method uses a set of

simple deterministic rules for the adaptation of the channel multipath number in the upper level strategy; while at the same time, multiple instances of the same non-isotropic recombinative ES searches for the optimum channel parameter estimates within the lower level. Empirical result shows the proposed method does not require delicate tuning of strategy parameters, and is also highly independent on initialisation values.

Chapter 6

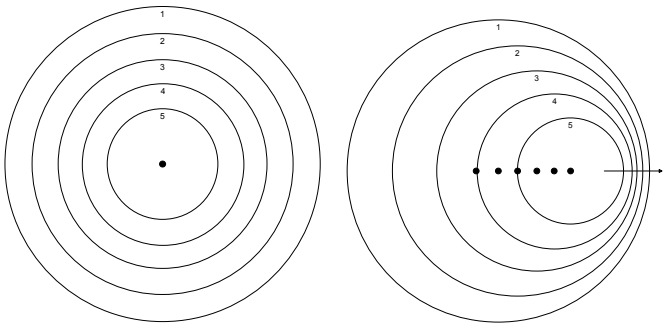
Doppler Estimation

6.1 Introduction

The Doppler effect due to motion is an important characteristic of the wireless channel that has been exploited by positioning technologies for decades. In a pulsed Doppler radar, the velocities of targets can be estimated directly by sensing Doppler frequencies. While in an airborne radar system, where the objective is detect other aircrafts, the Doppler frequencies are used to differentiate between unwanted echoes from slow moving (or stationary) objects from the ground and that from a target aircraft [109].

Furthermore, it is entirely possible to estimate the position and velocity of a moving source using only Doppler frequency measurements at several locations. Since each receiver location experiences a slightly different Doppler shift, then multiple Doppler shifted frequency measurements therefore provides a measure of the change in range during the observation interval [110]. A total of 5 spatially separate measurements is sufficient to solve for the target location (in 2 dimensional space), velocity and transmission frequency (if unknown) [111]. Two application examples are in passive sonar, where sonobuoys intercept the acoustic energy radiating from a moving source and in the tracking of a projectile which emits a tone [112]. If the transmission frequency is known, then 3 temporally separated measurements at a single sensor is adequate to determine the range, velocity and direction of motion of the moving target [113].

In a hybrid positioning system, Doppler, ToA and DoA measurements at same or different BSs can be combined to reduce the number of receivers required for a given location



(a) Stationary (b) Non-stationary

FIGURE 6.1: A wave when radiated from a point source when stationary and when moving; Wave is compressed in the direction of motion, spread out in the opposite direction, and unaffected in the direction normal to motion.

accuracy, and it will facilitate in the resolution of ambiguities [114]. While in a scattering rich environment, the Doppler information can be used to enhance the positional accuracy obtainable when strictly no LoS measurement is available. The authors in [115] uses the symmetry between Doppler shifts from opposite directions to identify scatters which lie on the same plane as MS, consequently, triangulation using these scatters are used to estimate the location of the MS.

In this chapter, we incorporate Doppler shift into our system model and extend the proposed ES channel parameter estimation method in Chapter 4 to jointly estimate the amplitude, ToA, DoA and Doppler frequency of each multipath. Doppler estimation brings additional challenges which must be addressed accordingly, furthermore, a modification to the existing ES approach is made which addresses a potential weakness of the standard ES algorithm.

6.2 The Doppler Effect and Doppler Frequency

The Doppler effect is a shift in the frequency of a wave radiated, reflected, or received by an object in motion. As illustrated by Figure 6.1, a wave radiated by a point source is compressed in the direction of motion and is spread out in the opposite direction. In both cases, the greater the object's speed, the greater the effect will be. Only at right angles to the motion is the wave unaffected. Since frequency is inversely proportional to wavelength, the more compressed the wave is, the higher its frequency is, and vice versa. Therefore, the frequency of the wave is shifted in direct proportion to the object's

velocity. The difference between between the transmitted and received frequencies due to motion is known as the Doppler shift, or Doppler frequency; in the subsequent sections, both terms are used interchangeably.

The Doppler frequency is essentially a measure of the change in the phase difference between the transmitted and received signals [109]. At any one instance, the phase of the received wave lags that of the transmitted wave by the transit delay τ, e.g. if τ is 1000 times the period of T of transmitted wave plus some fraction ϕ, then the phase between the received and transmitted wave will only differ by a fraction of one cycle - ϕ. Hence, if the transit time τ changes due to motion, then ϕ will change accordingly, and the Doppler shift is proportional to the rate of change of ϕ.

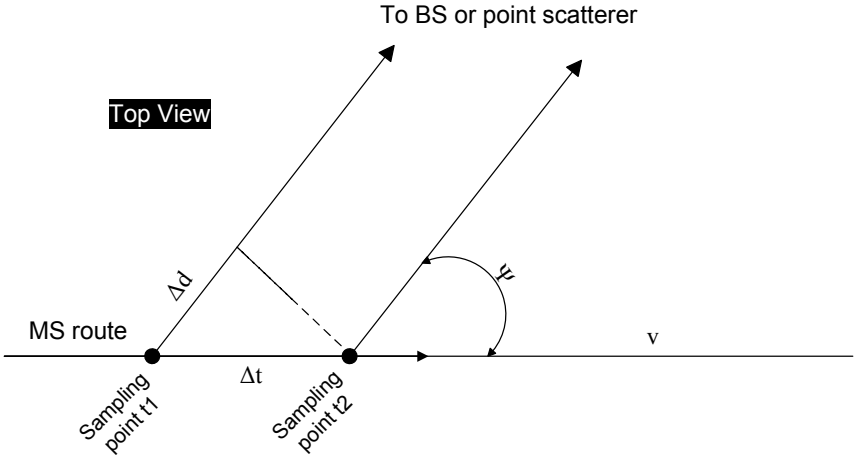


FIGURE 6.2: Parallel ray assumption. v is the speed of the mobile and ψ is the direction of arrival of the received data relative to the direction of motion of the mobile.

Assuming the transmitted waves from the MS to the BS or a point scatterer remain parallel for all points of the mobile route (Figure 6.2), this is approximately true for short route length [116]. The Doppler shift results from the fact that the transmitter or receiver movement over a short time interval Δt causes a slight change in distance $\Delta d = v\Delta t \cos\psi$ that the transmitted signal needs to travel to the receiver. The phase change due to this path length difference is $\Delta\phi = 2\pi v\Delta t \cos\psi/\lambda$. The Doppler frequency is then obtained from the relationship between the signal frequency and phase:

$$f_D = \frac{1}{2\pi}\frac{\Delta\phi}{\Delta t} = v\cos\psi/\lambda \tag{6.1}$$

where ψ is the arrival angle of the received data relative to the direction of motion, v is the MS velocity, and $\lambda = c/f_c$ is the signal wavelength ($c = 3 \times 10^8$m/s is the speed of light). For $|\psi| < 90°$, the MS is moving towards the BS, then the Doppler frequency is positive; while for $\psi = \pm 90°$, the Doppler frequency is zero; finally for $|\psi| > 90°$, the MS is moving away from the BS, the Doppler frequency will be negative.

6.3 System Model

A typical wireless mobile multipath propagation channel is illustrated in Figure 6.3, where data symbols are modulated by a known pulse shape at the mobile device in motion and transmitted through the spatial multipath channel. In addition to the ToA, DoA, and path attenuation, each path is further parametrised by its Doppler shift component f_D.

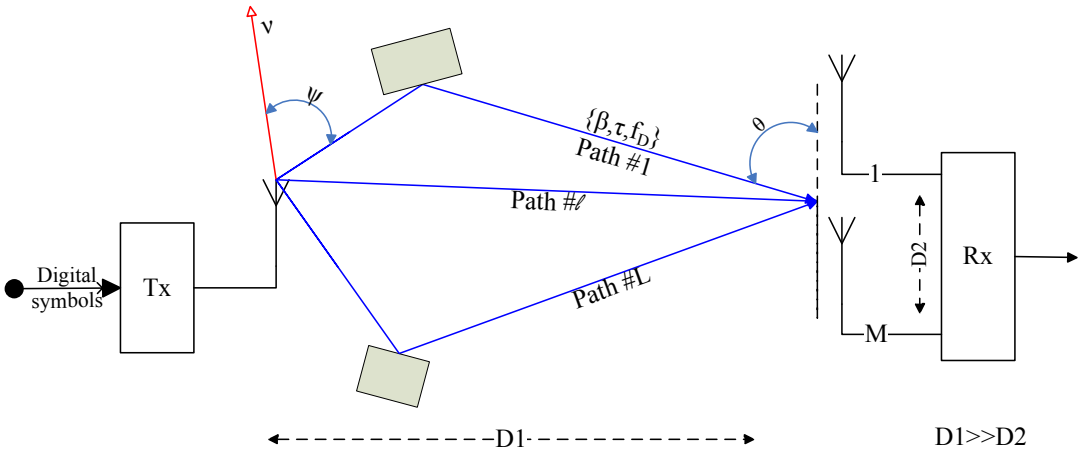


FIGURE 6.3: Illustration of a mobile wireless multipath channel environment. Each path is parametrised by its ToA τ_ℓ, DoA θ_ℓ, Doppler shift $f_D = v/\lambda \cos(\psi_\ell)$ and complex path attenuation β_ℓ. v is the speed of the mobile and ψ_ℓ is the direction of arrival of the received data relative to the direction of motion of the mobile.

Taking into the Doppler frequency into account, we can re-write the contribution of the ℓth wave to the M baseband signals at output of the antenna array as:

$$\begin{aligned} \mathbf{s}_\ell(t;\boldsymbol{\omega}_\ell) &\triangleq [s_{1,\ell}(t;\boldsymbol{\omega}_\ell), \ldots, s_{M,\ell}(t;\boldsymbol{\omega}_\ell)]^T \\ &= \beta_\ell \mathbf{a}(\theta_\ell) \exp\{j2\pi f_{D,\ell} t\} u(t-\tau_\ell) \end{aligned} \tag{6.2}$$

where $u(t)$ is the transmitted signal and the vector $\boldsymbol{\omega}_\ell \triangleq [\beta_\ell, \theta_\ell, \tau_\ell, f_{D,\ell}]$ contains the parameters of the ℓth wave. Each wave is characterised by its DoA θ_ℓ, ToA τ_ℓ, Doppler shift, $f_{D,\ell}$, and a complex path attenuation β_ℓ. Consequently, for the joint amplitude, DoA, ToA, Doppler shift estimation problem, the *objective function* becomes:

$$\epsilon(\mathbf{y};\boldsymbol{\omega}) \triangleq \int_0^{T_o} \left| \mathbf{y}(t) - \sum_{\ell=1}^{L} \hat{\beta}_\ell \mathbf{a}(\hat{\theta}_\ell) \exp\{j2\pi f_{D,\ell}\} u(t-\hat{\tau}_\ell) \right|^2 dt \tag{6.3}$$

where T_o is the observation period at the receiver, and the combined vector of channel parameters $\boldsymbol{\omega} \triangleq [\boldsymbol{\omega}_1 | \boldsymbol{\omega}_2 | \ldots, | \boldsymbol{\omega}_L]$ contains the channel parameters of all paths. While the parameter space dimension now becomes $n \triangleq 4L$.

In the subsequent sections, we will sometimes drop the multipath subscript ℓ from the Doppler shift frequency component $f_{D,\ell}$ where appropriate to improve readability.

6.4 Implementation

With the addition of the Doppler estimates $f_{D,\ell}$, the object variables part of the population matrix in (4.7) takes the form of:

$$\boldsymbol{\Omega} = \begin{bmatrix} \beta_{(1)1} & \cdots & \beta_{(1)L} & \theta_{(1)1} & \cdots & \theta_{(1)L} & \tau_{(1)1} & \cdots & \tau_{(1)L} & f_{D(1)1} & \cdots & f_{D(1)L} \\ \beta_{(2)1} & \cdots & \beta_{(2)L} & \theta_{(2)1} & \cdots & \theta_{(2)L} & \tau_{(2)1} & \cdots & \tau_{(2)L} & f_{D(2)1} & \cdots & f_{D(2)L} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \beta_{(\mu)1} & \cdots & \beta_{(\mu)L} & \theta_{(\mu)1} & \cdots & \theta_{(\mu)L} & \tau_{(\mu)1} & \cdots & \tau_{(\mu)L} & f_{D(\mu)1} & \cdots & f_{D(\mu)L} \end{bmatrix} \tag{6.4}$$

where each row represent one candidate solution to the joint parameter estimation problem. The modification to the matrix of strategy parameters $\boldsymbol{\Sigma}$ in (4.8) is trivial.

6.4.1 Ambiguity Function

For the moment, consider only the problem of estimating the ToA and Doppler components, one application area which considers this type of problem frequently is radar [109, 117]. In radar parameter estimation, the estimated delay component is the round trip time and not the ToA, but the underlying principles are the same. One important property that arises frequently in radar signals design is the ambiguity function.

The optimal receiver here is the correlator that performs time correlation over the observation period T_o between the received signal $y(t)$ and the required signal with a form $u(t-\tau)e^{j2\pi ft}$, where τ and f are unknown non-random parameters we are trying to estimate. The time correlation is followed by envelop detection, whose role is to eliminate the unknown phase term, that is,

$$\Lambda(\tau, f) = \left| C(\tau, f) \right|^2 \tag{6.5}$$

with

$$C(\tau, f) = \int_{T_o} y(t) u^*(t-\tau) e^{-j2\pi ft} dt \tag{6.6}$$

The signal $C(\tau, f)$, is the result of the correlation, over the observation time T_o, of the received signal $y(t)$ with a replica of the transmitted signal $u(t)$, with time delay τ and frequency shift f. The signal $C(\tau, f)$ therefore appears as a surface that is dependent on two parameters, τ and f.

Assuming the actual delay and Doppler shift are τ_a and f_a respectively, in situations where the noise is negligible compared to the transmitted signal, then the received signal is reduced to:

$$y(t) = u(t-\tau_a) e^{j2\pi f_a t} \tag{6.7}$$

and $C(\tau, f)$ becomes

$$C(\tau, f) = \left| \int_{T_o} u(t-\tau_a) u^*(t-\tau) e^{j2\pi(f_a - f)t} dt \right|^2 \tag{6.8}$$

Following a change of variables to bring the target to the origin:

$$\begin{aligned} \tau' &= \tau - \tau_a \\ f' &= f - f_a \end{aligned}$$

and

$$z = t - \tau + \tau' \tag{6.9}$$

then, equation (6.8) becomes

$$C(\tau', f') = \left| \int_{T_o} u(z)u^*(z-\tau')e^{-j2\pi f' z} dz \right|^2 \tag{6.10}$$

Equation (6.10) is known as the ambiguity function;[1] the function inside the magnitude signs is the time-frequency autocorrelation function of $u(t)$, which is a measure of the degree of similarity between a complex envelope and a replica of it that is shifted in time and frequency [118]. To minimise potential errors in estimation, the ideal ambiguity function is one at the origin and zeros else where e.g. a Dirac delta function at the origin without any side lobes, however, this requires an infinite transmitted bandwidth and an infinite observation period. Nevertheless, close approximations can be achieved by using more sophisticated transmission signal $u(t)$. The ambiguity function has a number of important properties:

- The amplitude of the global maximum depends solely on the SNR and not on the form of $u(t)$. In contrast, the surface of $C(\tau', f')$, dependent on $u(t)$, can have several maxima in the presence of multipath, all of which have comparable amplitude to the global maximum, thus creating ambiguity as to the position of the multipath in the delay-frequency plane [117].

- The shape of the surface $C(\tau', f')$ will determine the system resolution capability. The narrower the correlation peaks, the easier it is to discriminate between the multipath.

- The ambiguity function has a dual symmetry with respect to the time axis τ', and Doppler frequency axis f'. The volume under the surface circumscribed by $C(\tau', f')$ and the origin is equal to one for normalised $u(t)$. This means the disturbance caused by a multipath is constant throughout the time-Doppler plane. The only available flexibility is to define $u(t)$ in such a way as to minimise this disturbance in certain areas of the plane.

Consider the sequence of rectangular pulses shown in Figure 6.4. It is characterised by the pulse duration T, the inter-pulse spacing T_p, and the total number of pulses K.

[1] sometimes referred to as Woodward's ambiguity function because of his pioneering work with it

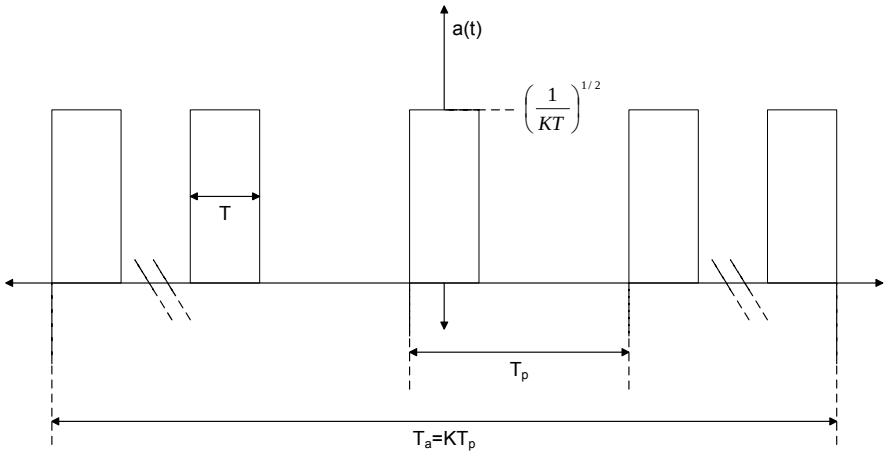


FIGURE 6.4: Rectangular pulse train

This sequence is frequently used in radar and sonar systems as it is easy to generate, the optimum receiver is easy to implement and the parameters can be varied to match different operating conditions [118].

By employing coded pulse sequences such as a rectangular pulse train for our transmitted signal $u(t)$, it is possible to control the width of the major peak in both directions of the ambiguity function [109]. This is shown in Figure 6.5 using an approximate contour representation of $C(\tau', f')$.

- We can decrease the width of the major peak in the frequency domain by increasing T_a (or K).
- We can decrease the width of the major peak in the time domain by decreasing T (corresponds to an increased bandwidth).

However, employing constant repetition signals of this type has the disadvantage of including subsidiary peaks caused by the periodic structure. Although the distances between these side peaks can be controlled using parameters of $a(t)$ as shown in Figure 6.5, a much better alternative is to suppress their amplitudes by either using non-uniform repetition rates or pseudo-random binary sequences of a_k to modulate the signal $a(t)$.

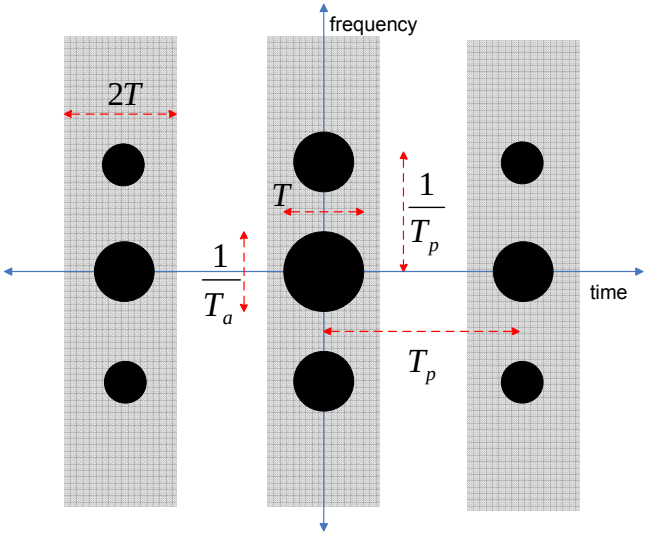


FIGURE 6.5: An approximate contour plot (centred at the origin) for a pulse train. The dark shaded areas indicate where $C(\tau', f')$ is significant. In the light shaded area, $C(\tau', f')$ is small. In the unshaded area $C(\tau', f')$ is zero.

6.4.2 Finite Data Length

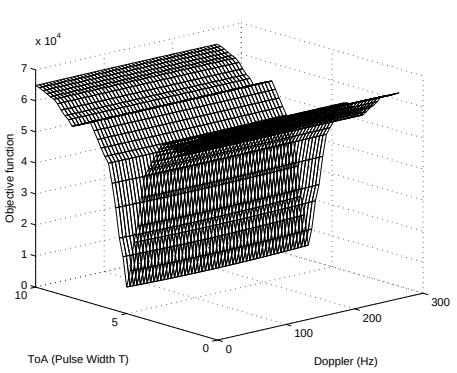
One important issue regarding the joint estimation of temporal and frequency domain parameters is the processing of the received data. Since a shift in frequency is equivalent to a continuous shift in the phase of the wave in propagation, thus detection of Doppler shift is easier if the received data is observed over a large time period, i.e. Doppler resolution $\propto$ observation interval. On the other hand, ToA resolution is largely determined by the symbol period and sampling rate, and not directly effected by the length of the observation interval. Hence, in order to accurately estimate both the ToA and Doppler frequency, it is necessary to closely sample the received data over a large observation period.

Considering the average pedestrian and vehicular speeds, the observation span at the receiver can be in the region of a few milliseconds during which the LMDP will remain approximately constant, e.g. even for a mobile device travelling at $100kph$ ($28ms^{-1}$), its change in position is only approximately $17cm$ during a time interval of $6ms$. Therefore, if possible, it is desirable to observe the received signal continuously over this time period to maximise Doppler resolution as well as the effective SNR. However, even at a modest sampling rate, observing continuously over a few milliseconds requires the storage and

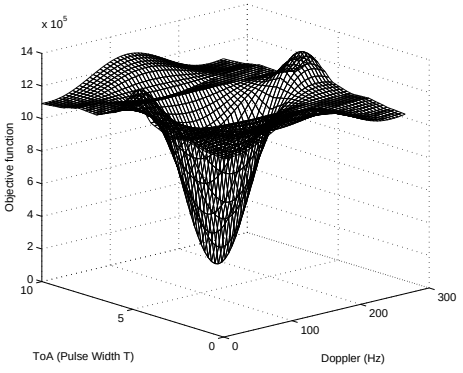
subsequent processing of a large amount of data (sampling at $20MHz$ over $6ms$ creates 120×10^3 bits of data).

In this chapter, the received data is sampled based on the scheme in [31] where it is observed over a window consisting of I regularly spaced time intervals of duration T_a each. The spacing between the centres of two consecutive observation intervals is denoted by T_f, where $T_f > T_a$. Hence, the observation duration and the observation span are IT_a and $(I-1)T_f + T_a$ respectively. The reason why the received signal is not continuously observed, i.e. $T_f = T_a$, is that the absolute Doppler frequency of the impinging waves is considerably smaller than the inverse of the burst duration. Hence, selecting $T_f > T_a$ makes it possible to increase the observation span, which leads to an enhancement of the Doppler resolution, while limiting the growth of the observation duration and therefore of the amount of measured data to be stored. By the Sampling Theorem $1/T_f$ must be larger than twice the maximum occurring Doppler frequency, while I must be confined in such a way that the observation span is smaller than the time interval during which the parameters of the waves remain approximately constant.

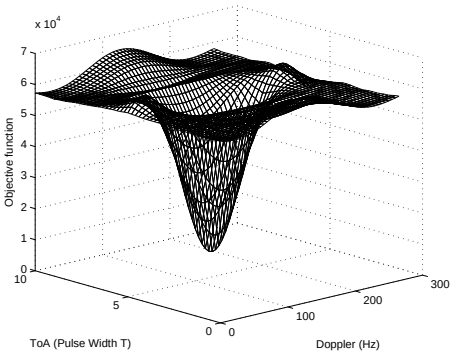
In Figure 6.6, we plot the observed objective function in (6.3) of a single path channel as a function of its ToA and Doppler frequency estimates, for three different observation duration and spans. In (a), the received data is continuously observed for $0.32ms$, producing 6400 samples as a result; however, the observation span is not long enough to provide sufficient Doppler resolution in order to detect the correct Doppler frequency. Ideally, the observation span should be the same as the time duration for which the channel parameters are assumed to be approximately constant; but observing continuously for such a long period will inevitably produce a large number of samples, as shown in (b). Finally in (c), $I = 4$ regularly spaced observations of duration $T_a = 0.08ms$ were taken, a spacing of $T_f = 2ms$ between consecutive observations ensures that the observation span is same as in (b) while the observation duration remains unchanged. Hence, sufficient Doppler resolution is achieved using the minimum number of samples. In all three cases, the received data is sampled at $20Mhz$, and the DoA is assumed to be known.



(a) Observation duration - $0.32ms$, Observation span - $0.32ms$, 6400 samples



(b) Observation duration - $6.1ms$, Observation span - $6.1ms$, 122000 samples



(c) Observation duration - $0.32ms$, Observation span - $6.1ms$, 6400 samples

FIGURE 6.6: Plot of objective function as a function of ToA and Doppler frequency for a single path channel. DoA is assumed to be known. Actual parameters are: $\tau = 5T$ and $f_d = 125Hz$. SNR=$5dB$

6.4.3 Bounded Mutations

In standard mutative self-adaptive ES, a complete offspring is produced by the addition of a normally distributed mutation component z_i to the recombinant parent individual according to:

$$\begin{aligned} \omega_i' &= \omega_i + z_i \\ z_i &\sim \sigma_i \mathcal{N}_i(0,1) \end{aligned} \tag{6.11}$$

naturally, the amount of mutation is controlled by the mutation strengths σ_i, which are normally at their largest in the beginning of the evolution process. Therefore, a certain proportion of the offspring are likely to be placed outside the search limits during this period. This is shown in Figure 6.7, where the percentage of the invalid offspring

immediately following mutation at each generation for shown for 4 different initial σ_i values.

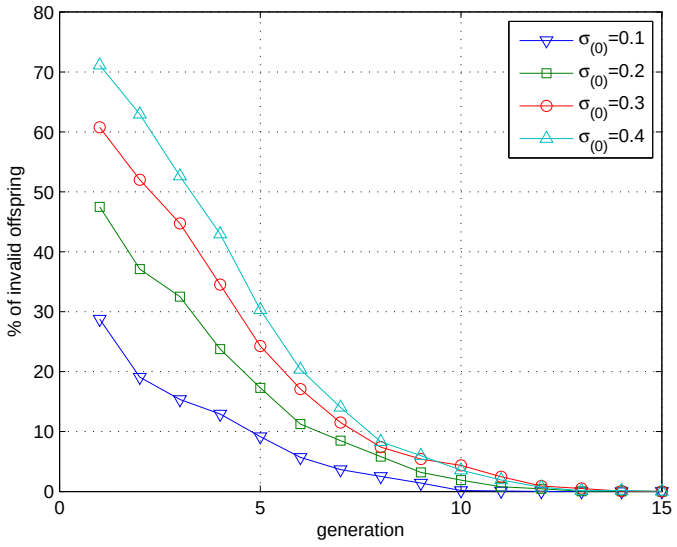


FIGURE 6.7: Percentage of invalid offspring during the early generations for 4 different initial σ_i values.

The standard ES approach to deal with these invalid offspring is to map the offending offspring back to the value of the bound:

$$\forall i : \omega_i = \max\left\{\omega_\perp, \min\left\{\omega_\top, \omega_i\right\}\right\} \tag{6.12}$$

such that all values are bounded within $\omega_\perp \leq \omega_i \leq \omega_\top$ as desired. Since we have normalised the values of ω_i using the pre-defined values of $\tau_{\max}, \theta_{\max}$ etc, then the lower $\omega_\perp$ and upper $\omega_\top$ limits here are 0 and 1 respectively.

However, applying the cap in (6.12) to any invalid offspring is not ideal for two reasons:

- The lower $\omega_\perp$ and upper $\omega_\top$ bounds are normally chosen in such a way that no multipath parameters are expected to lie outside (or even close) these limits, hence, mapping invalid estimates back to these extreme values will not produce much superior estimate!
- Self-adaptation in ES is based on the indirect link between the parameter estimates and its associated mutation strength, i.e. a good estimate implies the mutation

strength that helped to create it must also be good. However, in cases when the actual parameter are in fact located near either of the bounds, then an offspring that over-stepped the boundary which is then subsequently mapped back to the boundary will likely to be select for reproduction; which means the mutation strength associated with this offspring (which is likely to be too big, since it created an invalid offspring) will also be selected.

Therefore, we propose to include the following additional bound which operate on the applied mutations $z_i \sim \mathcal{N}(0, \sigma_i^2)$ prior to the caps in (6.12). The idea is to keep the number of invalid offspring in the early iteration as small as possible so that the "quality" of the population is high, at the same time not modifying the p.d.f of the applied mutations:

$$\begin{aligned} \omega_i + z_i < \omega_\perp \quad &\mapsto \quad z_i = -z_i \\ \omega_i + z_i > \omega_\top \quad &\mapsto \quad z_i = -z_i \end{aligned} \tag{6.13}$$

hence, the signs of any component $\forall i : z_i$ that will mutate an offspring to outside the search boundary are reversed. The p.d.f of z_i dictates that small mutations are much more likely to occur than larger ones, which means that the probability of a negated mutation shifting the offspring outside the opposite boundary is relatively small, e.g. $p(\omega_i + z_i > \omega_\top) > p(\omega_i - z_i < \omega_\perp)$. In addition, by modifying the sign and not the value of z_i, we have not altered its distribution, since $p\big(z_i = z'\big) = p\big(z_i = -z'\big)$.

6.5 Simulations

The performance of the proposed joint amplitude, ToA, DoA, Doppler shift estimation method have been assessed by means of Monte Carlo simulations. The transmitted signal $u(t)$ consists of a periodically repeated burst signal $a(t)$, i.e. $u(t) = \sum_{i=-\infty}^{\infty} a(t - iT_a)$. The burst signal is of the form $a(t) = \sum_{k=0}^{K-1} a_k p(t - kT_p)$, where $[a_0, \ldots, a_{K-1}]$ denote the known binary sounding sequence (taking values ± 1 with equal probability) of length K and the rectangular shape pulse respectively. The pulse shape during T_p is related to T_a according to $T_a = KT_p$. We have elected to sacrifice the time resolution capability somewhat by choosing $T = T_p$; this is done to reduce simulation time by using a lower sampling frequency and still maintain a reasonable pseudo-random sequence length K.

We use a ULA which comprises of $M = 4$ equi-distance elements with a spacing of $\lambda/2$. The pseudo-noise sounding sequence consists of $K = 80$ rectangular pulse $p(t)$ with symbol period $T_p = 1\mu s$. Hence the duration of each repeated burst of sounding sequence is $T_a = 80\mu s$. At the BS, the received data is sampled at an interval of $T_s = 50ns$ for a duration of T_a seconds at a time, while the interval between successive observations is $T_f = 2ms$. $I = 4$ such intervals are normally taken, hence giving an effective observation interval of $T_o = 6.1ms$.

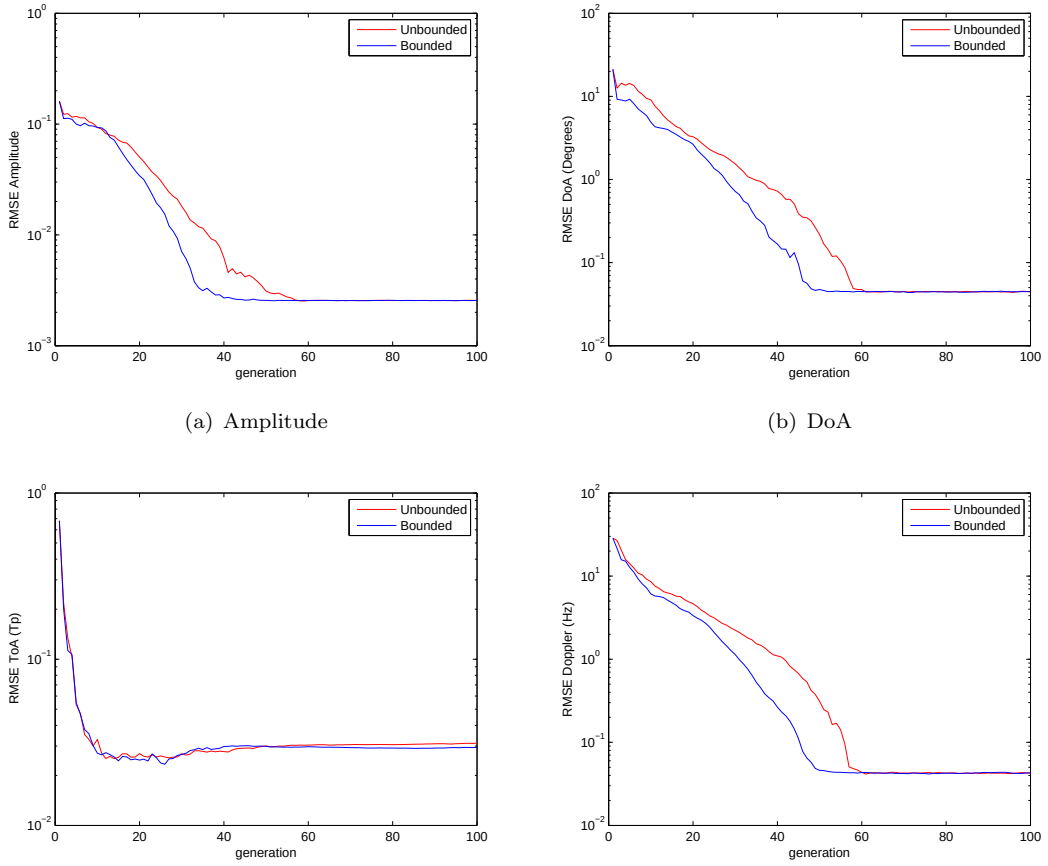
Unless otherwise stated, the proposed ES method is initialised according to: $[\mathbf{\Omega}]_{ij} \in U[0,1]$; $[\mathbf{\Sigma}]_{ij} \in U[0,0.3]$;

6.5.1 Special Case - Estimation of One Wave

In this section, we present some empirical analysis on the convergence time of the proposed method in a single path channel, to be specific, we are interested in the effect of the quality of initial estimates on convergence. But first, we investigate the possible gain in performance from introducing the proposed bounded mutations step to the standard ES method. The effects of reducing the number of invalid offspring on the ES performance can be seen in Figure 6.8, where the RMSE values for all the parameter estimates are shown at each generation in a single path channel. Hence, by keeping the number of invalid offspring to a minimum in the early generations (0% in this example, but not always possible), we have ensured that the number of potentially good offspring is high, and the algorithm is able to reach convergence significantly quicker.

We ascertained in Chapter 4 that a key benefit of the ES method is its ability to convergence to optimum estimates regardless of the initial state. In this experiment, we try to get some insight on the effects of initial values on the convergence time of the proposed method. Since the convergence rate of ES is highly dependent on the selection pressure λ/μ and the mutation learning rates γ and γ', our primary interest here is the relative performances when the initial conditions vary, so these parameters are kept constant here. The effects of selection pressure λ/μ have been discussed in Chapter 4.

Figure 6.9 shows the RMSE values for all parameter estimates at each generation based on initial estimates containing 20%, 40% and 60% errors. It is worth nothing here

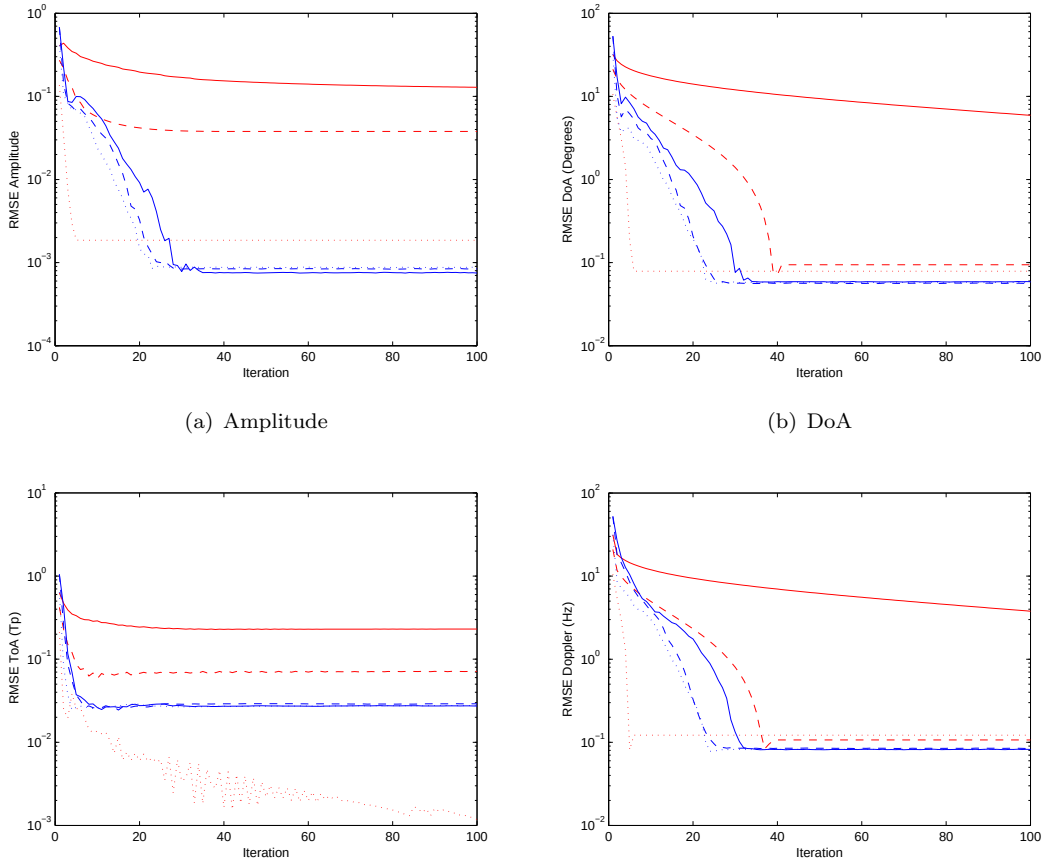


(a) Amplitude (b) DoA

(c) ToA (d) Doppler

FIGURE 6.8: RMSE values for all channel parameter at each generation for bounded and unbounded mutations. SNR=10dB. $L = 1$.

that the magnitude of the errors are not important, since the same levels of errors will have very different performance effects under vary channel conditions. Hence, it is more informative to consider them simply as good, average and poor operating conditions. We can observe that under good conditions (red dotted lines), SAGE is capable of convergence under < 20 iterations. This concurs with the results in [31] when the mulipath are well resolved. Under the same conditions, the convergence rate of ES is unable to match that of SAGE, there are numerous reasons for this. As the extrinsic property of ES suggests, even when the current population lies near the vinicity of the optimum, a significant proportion of offspring produced via mutations will not be towards the direction of the optimum at each generation, hence rate of progress is somewhat sacrifised for space exploration. In addition, the initial mutation strength is a fixed parameter that is often chosen to be large as possible while not generating too many samples which lie outside the search range, regardless of how good the starting



(a) Amplitude (b) DoA

(c) ToA (d) Doppler

FIGURE 6.9: RMSE values for all channel parameters at each iteration number based on initial estimates with 20% error (dot), 40% error (dash), and 60% error (solid). Red-SAGE, Blue-ES. L=1. SNR=10dB.

point might be. Of course, if it is possible to know in advance how accurate the initial values are, one could adjust the initial mutation strength accordingly to reduce degree of divergence from the initial value, but this is clearly not possible. However, this type of ideal operating conditions are very rarely observed in real life.

We should explain here why the ToA result for SAGE under good conditions appears to be considerably better than all others (red dotted lines in Figure 6.9 (c)). Since we have used a relatively low sampling frequency, it is not possible for the estimators to achieve ToA estimation with very high precision. In fact, differences in delay which are less than $0.05T_p$ here are indistinguishable and the steady state RMSE values attained by the ES ToA estimates are already the lowest possible. The number precision in which the optimiser in SAGE operates is not limited, hence the SAGE ToA estimates appears to be improving indefinitely in this case. Had we used a very small lower bound on

the mutations in our ES method, we would observe the same thing; but this is of no particular benefit.

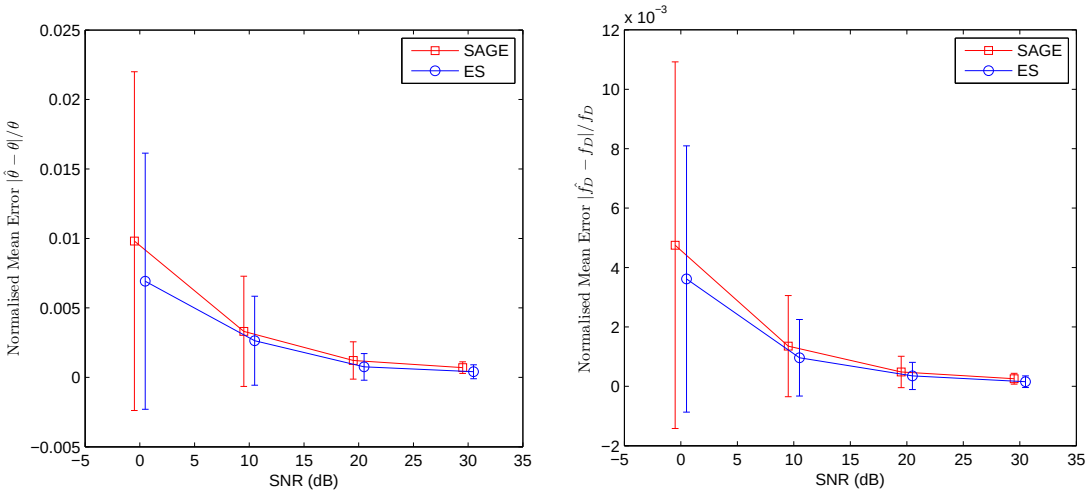
When more errors are introduced in the initial estimates, convergence rate of SAGE declines dramatically (red dashed lines), this is particularly noticeable in the DoA and Doppler estimates, in addition, there is also a significant increase in the steady state error of its amplitude and ToA estimates. At this point, we have reached a limit at which the standard SAGE algorithm can tolerate in initial errors. Upon further increases, it is obvious that a non-convergence state has been reached (red solid lines).

On the other hand, under the same conditions, the proposed ES method exhibits very little differences in convergence time (blue lines), i.e. the differences in convergence time for all 3 test scenarios spans less than 10 iterations. In addition, in conformance with previous results in Chapter 4, the same steady state estimation error is obtained regardless of initial conditions. These are particularly useful properties since it is possible to obtain consistent result from the proposed method under a variety of conditions without the need to fine tune control parameters.

6.5.2 Multipath Performance

In this section, the performance of the proposed ES method is analysed in a multipath environment, we start by comparing the estimation accuracy of the LoS component with the SAGE algorithm under good operating conditions. As previously mentioned, it is not possible to obtain good estimation results from SAGE unless reasonably accurate initialisations are provided, hence this section makes comparisons against the SAGE algorithm under good operating conditions, where accurate initialisation is available for a number of strong specular multipath.

Figure 6.10 shows the normalised mean errors and standard deviations in DoA and Doppler estimates of the LoS component. We can observe that even under these ideal conditions, where the estimation performance of SAGE is quite good (single digit percentage of errors at the low SNR range), a performance gain in the proposed method can still be observed. Where roughly 1/4 reduction in both mean error and standard deviations can be seen at 0dB.



(a) DoA (b) Doppler

FIGURE 6.10: Normalised mean errors in DoA and Doppler estimates of the LoS component. $(10 \odot 50)$-ES. $L = 3$

Of course, the real advantage of ES as demonstrated previously is its total independence from initialisation, even for high dimensional problems. Figure 6.11 shows the distributions of the ToA, DoA and Doppler estimates from the proposed ES method in a strong multipath channel using random initialisations, where all 6 multipath are contained in a delay spread of ~ 3 symbol periods. On the x-axis, the ticks and labels indicates the actual values of the channel parameters in all three sub-figures. Straight away, we can observe the proposed method exhibits excellent resolution capabilities, such that all 6 components are clearly distinguishable in each of the parameter domains. Multipath which closely resides in any single parameter dimension are resolved with ease, as shown by the the two DoA components which differ by only 1 degree, as well as the group of three Doppler frequencies that span only 13Hz.

6.6 Summary

In this chapter, we have extended our work in Chapter 4 to incorporate mobility in our system and presented a joint amplitude, ToA, DoA, Doppler shift estimation method based on the ES algorithm. A bounded mutation mechanism is introduced to reduce the number of invalid offspring produced at each generation. Simulation result shows by keeping the number of invalid offspring at a minimum, a significant gain in convergence is possible compared to standard ES algorithm. In comparisons with the SAGE

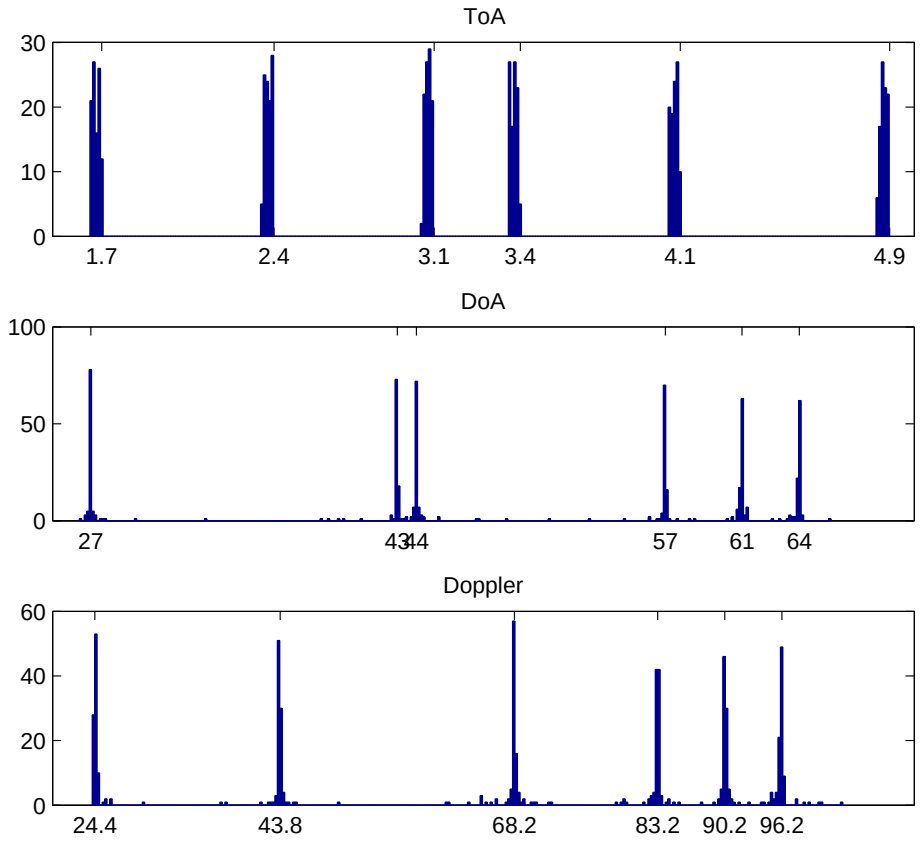


FIGURE 6.11: Distributions of ToA, DoA and Doppler estimates using random initialisation for $(15 \odot 100)$-ES in a strong multipath channel. The x-axis labels indicates position and value of actual channel parameters. SNR=20dB. L=6.

algorithm, the proposed method is shown to have superior estimation accuracy as well as better immunity from poor initialisation. Finally, excellent resolution capabilities of the proposed method is demonstrated in a strong multipath environment.

Chapter 7

Conclusion

In the next generation of communications networks, the telecommunications applications require various types of context information of the environments, persons, and devices to offer flexible and adaptive services in wireless networks. Applications that exhibits location aware intelligence offers significantly improved quality of service to the mobile subscriber, as well as enhanced safety, security, and operational efficiency to the location context dependent business and organisations. In this context, accurate, reliable and real-time mobile wireless positioning technologies are required, of which the estimation of location dependent channel parameters is an integral part of the positioning process. Due to the unique nature of wireless positioning using mobile signal measurements, small errors in parameter estimation typically leads to large errors in final position estimates. Consequently, the estimation algorithms adopted by wireless positioning systems must provide accurate parameter estimates, even under challenging conditions. It is clear that methods which jointly estimates more than one type of channel parameters have significant advantages in comparison to traditional techniques, not only they are capable of resolving multipath which are closely spaced, the availability of multiple channel parameters at each base station enables the utilisation of various emerging hybrid and fingerprinting positioning techniques. The intelligent joint parameter estimation techniques discussed in this thesis are highly robust and accurate methods which are strong candidate for the application of wireless mobile positioning.

7.1 Summary of Contributions

The main focus of the work in this thesis has been on the intelligent joint estimation techniques of location dependent channel parameters. The main contributions of this thesis are:

- A comprehensive critical review of the current and most popular methods for the joint estimation of location dependent channel parameters was given. Empirical analysis on a number of key issues such as estimation accuracy, multipath resolution, computational complexity and model error handling have been addressed.

- An adaptive interference cancellation technique has been proposed to improve the performance of the SAGE algorithm under low SNR and/or weak multipath conditions, where traditionally, the performance of the standard SAGE algorithm is poor due to the nature of the brute force interference cancellation approach. Simulation results shows the proposed method is less susceptible to residual errors accumulated from interference cancellation and the estimation accuracy of the weaker multipath are significantly improved.

- An intelligent joint channel parameter estimation technique based on the application of Evolution Strategy is proposed. Being a type of an Evolutionary Algorithm, the proposed method utilises a population of candidate solutions to simultaneously search for the maximum likelihood estimates of the concerned channel parameters. The proposed ES method is shown to overcome some key weaknesses such as low SNR performance and dependency on initialisation of the traditional iterative maximum likelihood solutions like SAGE, while also being less computationally demanding (under certain conditions). A comprehensive empirical analysis of the tuning parameters relating to the proposed method demonstrates its flexibility as well as its ability to be self manageable.

- A two tier Hierarchically Organised Evolution Strategy is proposed to jointly estimate the number of multipath as well as the channel parameters. The proposed method utilises the lower level strategies to jointly estimate the channel parameters while the upper level performs conditional selection of the lower levels to search for the channel order. The proposed method is shown to be highly independent of

initial states (for both channel parameters and channel order), while at the same time insensitive to tuning parameters.

- The proposed ES method was also extended to further estimate the Doppler shift of each multipath. During which a modified mutation bound was introduced to reduce the number of invalid offspring produced in the standard ES algorithm. The proposed changes improve the convergence rate of the standard ES significantly under certain conditions.

7.2 Future Work

Further research can be carried out building on the work presented in this thesis. The following are some of the areas which deserve attention in future research.

- The true potential of Evolution Strategies cannot be realised unless it is subject to parallel implementations. Evaluation time of the objective function tends to be the most computationally demanding step in any EA, and being a totally independent operation, significant gains in computation time can be expected if the evaluations of offspring can be somewhat parallelised. A brief overview of EA parallelisation architecture is given in [83]. Although the implementations of our proposed intelligent estimation techniques are vectorised wherever possible, it is by no means an efficient parallel implementation.

- We noticed that the convergence rate of ES remains almost unaffected by the quality of initial estimates. This is undesirable if the channel conditions are good and accurate initialisations are available. Since ES estimation and convergence performance are both directly related to the population (μ) and offspring (λ) sizes, it makes sense to adopt a mechanism of adjusting these values dynamically depending on current conditions. A recent example is proposed in [119] for operation in noisy environments, where offspring size is increased when the algorithm struggles to make significant progress. A similar strategy can be applied directly in good conditions to maximise convergence.

- Our intelligent channel parameter estimation methods proposed in this thesis could easily be adapted to perform other channel parameter estimation, e.g. channel

impulse response, delay spread, channel coherence time, etc. In particular, a most useful application is the task of joint channel order and channel estimation, where our proposed joint detection and estimation meta-ES method in Chapter 5 is an ideal candidate.

- Conventional positioning techniques can be viewed as a two-step procedure: parameter measurement followed by position estimation. These approaches have the disadvantage of making a premature decision on an intermediate parameter (e.g. TDoA) in the first step, thus discarding useful information. A better approach would use the principle of least commitment: preserve and propagate all the intermediate information to the end and make an informed decision at the very last step [120], hence solving the positioning problem directly in a single step. This approach essentially attempts to find the point in space which yields maximum energy, which due to the high dimensional optimisation involved, has had limited use to date.

- Although many positioning devices and services are currently available, it is still necessary and challenging to develop an integrated and seamless positioning platform to provide a uniform solution for different network configurations. If successful, the integration of different positioning systems within various networks would enable a continuous delivery of location context information spanning vast distances. Much work remains to be done in this area of system inter-operability.

Appendix A

Derivation of LMS Gradient Estimate

The method of steepest decent is used to find the vector $\mathbf{w}^{(k)} \triangleq [w_1^{(k)}, \ldots, w_L^{(k)}]$ at time k that minimises the quadratic function $E\left[\left|\mathbf{e}^{(k)}(t)\right|^2\right]$. The error function $\mathbf{e}^{(k)}(t)$ is defined as the difference between the received signal $\mathbf{y}(t)$ and its estimate at the kth step:

$$\mathbf{e}^{(k)}(t) = \mathbf{y}(t) - \sum_{\ell=1}^{L} w_\ell^{(k)} \mathbf{s}_\ell(t; \hat{\boldsymbol{\omega}}_\ell^{(k)}) \tag{A.1}$$

where $\mathbf{s}_\ell(t; \hat{\boldsymbol{\omega}}_\ell^{(k)})$ is the estimated contribution to the received data from the ℓth path at the kth step and $\hat{\boldsymbol{\omega}}_\ell^{(k)} \triangleq [\hat{\beta}_\ell^{(k)}, \hat{\tau}_\ell^{(k)}, \hat{\theta}_\ell^{(k)}]$ is the vector of channel parameter estimates of the ℓth path at the kth step.

Instead of solving for (A.1) directly, a coefficient update equation of the form [78]

$$\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} + \Delta\mathbf{w}^{(k)} \tag{A.2}$$

is used, where, $\Delta\mathbf{w}^{(k)}$ is a correction that is applied to the coefficients $\mathbf{w}^{(k)}$ to form a new set of coefficients $\mathbf{w}^{(k+1)}$. The key is then to control $\Delta\mathbf{w}^{(k)}$ such that the sequence of corrections should decrease the mean square error $E\left[\left|\mathbf{e}^{(k)}(t)\right|^2\right]$. In the method of steepest descent, each correction involves taking a step in the direction of maximum descent down the quadratic error surface. Mathematically, this direction is given by the gradient, which is the vector of partial derivatives of $E\left[\left|\mathbf{e}^{(k)}(t)\right|^2\right]$ w.r.t. the coefficients

$\mathbf{w}^{(k)}$:

$$\nabla E\Big[\big|\mathbf{e}^{(k)}(t)\big|^2\Big] = \begin{bmatrix} \frac{\partial E\Big[\big|\mathbf{e}^{(k)}(t)\big|^2\Big]}{\partial w_1} \\ \vdots \\ \frac{\partial E\Big[\big|\mathbf{e}^{(k)}(t)\big|^2\Big]}{\partial w_L} \end{bmatrix} \tag{A.3}$$

Finally, the LMS algorithm simplifies the steepest decent method by using one-point sample mean, and the expectation requirement can be relaxed. Assuming $\mathbf{w}^{(k)}$ is complex, then the gradient vector is the derivative of $\big|\mathbf{e}^{(k)}(t)\big|^2$ w.r.t. $\mathbf{w}^{*(k)}$. Writing in matrix notations, the partial derivative for the ℓth component is:

$$\begin{aligned}
& \frac{\partial}{\partial w_\ell^*}\left\{\sum_{m=1}^{M}\sum_{n=1}^{N}\mathbf{E}(m,n)\mathbf{E}^*(m,n)\right\} \\
= \;& \frac{\partial}{\partial w_\ell^*}\left\{\sum_{m=1}^{M}\sum_{n=1}^{N}\left[\mathbf{Y}(m,n)-\sum_{\ell=1}^{L}\mathbf{S}_\ell(m,n)w_\ell\right]\left[\mathbf{Y}^*(m,n)-\sum_{\ell=1}^{L}\mathbf{S}_\ell^*(m,n)w_\ell^*\right]\right\} \\
& \text{we drop the indices } m \text{ and } n \text{ for now to reduce clutter} \\
= \;& \frac{\partial}{\partial w_\ell^*}\left\{\left[\mathbf{Y}\mathbf{Y}^*-\mathbf{Y}\sum_{\ell=1}^{L}\mathbf{S}_\ell^*w_\ell^*-\mathbf{Y}^*\sum_{\ell=1}^{L}\mathbf{S}_\ell w_\ell+\sum_{\ell=1}^{L}\mathbf{S}_\ell w_\ell\sum_{\ell=1}^{L}\mathbf{S}_\ell^*w_\ell^*\right]\right\} \\
= \;& \frac{\partial}{\partial w_\ell^*}\left\{\left[\underbrace{-\mathbf{Y}\sum_{\ell=1}^{L}\mathbf{S}_\ell^*w_\ell^*}_{\bigstar}+\underbrace{\sum_{\ell=1}^{L}\mathbf{S}_\ell w_\ell\sum_{\ell=1}^{L}\mathbf{S}_\ell^*w_\ell^*}_{\blacklozenge}\right]\right\} \\
& \text{Partial derivatives of} \bigstar \text{and} \blacklozenge : \\
\bigstar' = \;& \frac{\partial}{\partial w_\ell^*}\left\{-\mathbf{Y}\Big[\mathbf{S}_1^*w_1^*+\ldots+\mathbf{S}_\ell^*w_\ell^*+\ldots+\mathbf{S}_L^*w_L^*\Big]\right\} = -\mathbf{Y}\mathbf{S}_\ell^* \\
\blacklozenge' = \;& \frac{\partial}{\partial w_\ell^*}\left\{\sum_{\ell=1}^{L}\mathbf{S}_\ell w_\ell\Big[\mathbf{S}_1^*w_1^*+\ldots+\mathbf{S}_\ell^*w_\ell^*+\ldots+\mathbf{S}_L^*w_L^*\Big]\right\} = \sum_{\ell=1}^{L}\mathbf{S}_\ell w_\ell\mathbf{S}_\ell^* \\
& \text{Combining} \bigstar' \text{and} \blacklozenge' \text{and applying summation gives:} \\
= \;& \sum_{m=1}^{M}\sum_{n=1}^{N}\left[-\mathbf{Y}\mathbf{S}_\ell^*+\sum_{\ell=1}^{L}\mathbf{S}_\ell w_\ell\mathbf{S}_\ell^*\right] \\
= \;& \sum_{m=1}^{M}\sum_{n=1}^{N}\left[-\mathbf{S}_\ell^*\Big(\mathbf{Y}-\sum_{\ell=1}^{L}\mathbf{S}_\ell w_\ell\Big)\right] \\
= \;& \sum_{m=1}^{M}\sum_{n=1}^{N}\Big[-\mathbf{S}_\ell^*(m,n)\mathbf{E}(m,n)\Big] \quad \text{for } \ell=1,\ldots,L
\end{aligned} \tag{A.4}$$

Appendix B

Vectorisation of Fitness Evaluations

Without parallel implementations, a computationally efficient method of evaluating the fitness values for all λ offspring is by vectorising the computation of the **for** loop in Algorithm 4.1. The computational bottleneck here is the summation of the estimates of all paths in (4.3):

$$\hat{\mathbf{y}} = \sum_{\ell=1}^{L} \hat{\beta}_\ell \mathbf{a}(\hat{\theta}_\ell) g(t - \hat{\tau}_\ell) \tag{B.1}$$

which is evaluated λ times for each ES generation.

Define $\mathbf{G}_{\text{ES}}$ as the $\lambda L \times N$ matrix formed by stacking the delayed pulse shape samples according to:

$$\mathbf{G}_{\text{ES}} \triangleq \left[\mathbf{g}\left(\tau_1^{(1)}\right), \ldots, \mathbf{g}\left(\tau_1^{(\lambda)}\right) \middle| \mathbf{g}\left(\tau_2^{(1)}\right), \ldots, \mathbf{g}\left(\tau_2^{(\lambda)}\right) \middle| \ldots \middle| \mathbf{g}\left(\tau_L^{(1)}\right), \ldots, \mathbf{g}\left(\tau_L^{(\lambda)}\right)\right]^T \tag{B.2}$$

where $\tau_\ell^{(j)}$ represents the jth candidate solution for the ToA of the ℓth path.

Similarly, define $\mathbf{B}_{\text{ES}}$ as the $\lambda L \times N$ matrix of all candidate amplitudes according to:

$$\mathbf{B}_{\text{ES}} \triangleq \left[\boldsymbol{\beta}_1^{(1)}, \ldots, \boldsymbol{\beta}_1^{(\lambda)} \middle| \boldsymbol{\beta}_2^{(1)}, \ldots, \boldsymbol{\beta}_2^{(\lambda)} \middle| \ldots \middle| \boldsymbol{\beta}_L^{(1)}, \ldots, \boldsymbol{\beta}_L^{(\lambda)}\right]^T \tag{B.3}$$

with each $1 \times N$ vector $\boldsymbol{\beta}_\ell^{(j)} \triangleq (\beta_\ell^{(j)} \mathbf{1}_N)^T$; where $\beta_\ell^{(j)}$ represents the jth candidate solution for the amplitude of the ℓth path.

Finally, by stacking all the candidate array responses into a $\lambda ML \times \lambda L$ matrix according to:

$$\mathbf{A}_{\mathrm{ES}} \triangleq \begin{bmatrix} \mathbf{a}\left(\theta_1^{(1)}\right) & & & & & & & & \\ & \ddots & & & & & & & \\ & & \mathbf{a}\left(\theta_1^{(\lambda)}\right) & & & & & & \\ & & & \mathbf{a}\left(\theta_2^{(1)}\right) & & & & & \\ & & & & \ddots & & & & \\ & & & & & \mathbf{a}\left(\theta_2^{(\lambda)}\right) & & & \\ & & & & & & \ddots & & \\ & & & & & & & \mathbf{a}\left(\theta_L^{(1)}\right) & \\ & & & & & & & & \ddots \\ & & & & & & & & & \mathbf{a}\left(\theta_L^{(\lambda)}\right) \end{bmatrix} \tag{B.4}$$

where $\theta_\ell^{(j)}$ represents the jth candidate solution for the DoA of the ℓth path. The sparse representation of $\mathbf{A}_{\mathrm{ES}}$ should be used in practice to reduce storage requirement.

Then, the evaluations of $\hat{\mathbf{y}}$ in (B.1) for all λ offspring in matrix notation are contained within the product:

$$\hat{\mathbf{Y}}_{\mathrm{ES}} = \mathbf{A}_{\mathrm{ES}}\left(\mathbf{B}_{\mathrm{ES}} \diamond \mathbf{G}_{\mathrm{ES}}\right) \tag{B.5}$$

which can be extracted with ease using built-in Matlab functionalities.

Bibliography

[1] M. Venkatachalam, K. Etemad, W. Ballantyne, and B. Chen, "Location services in WiMax networks," *IEEE Communications Magazine*, vol. 47, no. 10, pp. 92–98, Oct. 2009.

[2] J. Bull, "Wireless geolocation," *IEEE Vehicular Technology Magazine*, vol. 4, no. 4, pp. 45–53, Dec. 2009.

[3] TechNavio, "TechNavio insight report: Global location based services (LBS) market 2008-2012," Apr. 2009. [Online]. Available: http://www.technavio.com/content/global-location-based-services-lbs-market-2008-2012

[4] A. Sayed, A. Tarighat, and N. Khajehnouri, "Network-based wireless location: challenges faced in developing techniques for accurate wireless location information," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 24–40, Jul. 2005.

[5] M. Ibnkahla, *Signal Processing for Mobile Communications Handbook*. Boca Raton, FL, USA: CRC Press, 1998.

[6] M. Vossiek, L. Wiebking, P. Gulden, J. Wieghardt, C. Hoffmann, and P. Heide, "Wireless local positioning," *IEEE Microwave Magazine*, vol. 4, no. 4, pp. 77–86, Dec. 2003.

[7] S. S. Soliman and C. E. Wheatley, "Geolocation technologies and applications for third generation wireless," *Wiley InterScience Wireless Communications and Mobile Computing Magazine*, vol. 2, no. 3, pp. 229–251, May 2002.

[8] M. Porretta, P. Nepa, G. Manara, and F. Giannetti, "Location, location, location," *IEEE Vehicular Technology Magazine*, vol. 3, no. 2, pp. 20–29, Jun. 2008.

[9] G. Sun, J. Chen, W. Guo, and K. Liu, "Signal processing techniques in network-aided positioning: a survey of state-of-the-art positioning designs," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 12–23, Jul. 2005.

[10] S. Gezici and H. Poor, "Position estimation via ultra-wide-band signals," *Proceedings of the IEEE*, vol. 97, no. 2, pp. 386–403, Feb. 2009.

[11] S. Gezici, Z. Tian, G. Giannakis, H. Kobayashi, A. Molisch, H. Poor, and Z. Sahinoglu, "Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 70–84, Jul. 2005.

[12] F. Gustafsson and F. Gunnarsson, "Mobile positioning using wireless networks: possibilities and fundamental limitations based on available wireless network measurements," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 41–53, Jul. 2005.

[13] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 37, no. 6, pp. 1067–1080, Nov. 2007.

[14] Y. Gu, A. Lo, and I. Niemegeers, "A survey of indoor positioning systems for wireless personal networks," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 13–32, 2009.

[15] K. Krizman, T. Biedka, and T. Rappaport, "Wireless position location: fundamentals, implementation strategies, and sources of error," in *47th IEEE Vehicular Technology Conference*, vol. 2, May 1997, pp. 919–923.

[16] K. Pahlavan, X. Li, and J. Makela, "Indoor geolocation science and technology," *IEEE Communications Magazine*, vol. 40, no. 2, pp. 112–118, Feb. 2002.

[17] R. Bajaj, S. Ranaweera, and D. Agrawal, "GPS: location-tracking technology," *Computer*, vol. 35, no. 4, pp. 92–94, Apr. 2002.

[18] 3GPP, "The mobile broadband standard - LTE," Aug. 2010. [Online]. Available: http://www.3gpp.org/LTE

[19] IEEE Standards Association, "IEEE 802.11n-2009 standard," Oct. 2009. [Online]. Available: http://standards.ieee.org/getieee802/download/802.11n-2009.pdf

[20] ——, "IEEE 802.16-2009 standard," May. 2009. [Online]. Available: http://standards.ieee.org/getieee802/download/802.16-2009.pdf

[21] T. Sarkar, Z. Ji, K. Kim, A. Medouri, and M. Salazar-Palma, "A survey of various propagation models for mobile communication," *IEEE Antennas and Propagation Magazine*, vol. 45, no. 3, pp. 51–82, Jun. 2003.

[22] A. Lemma, A.-J. van der Veen, and E. Deprettere, "Analysis of joint angle-frequency estimation using ESPRIT," *IEEE Transactions on Signal Processing*, vol. 51, no. 5, pp. 1264–1283, May 2003.

[23] M. C. Vanderveen, C. B. Papadias, and A. Paulraj, "Joint angle and delay estimation (JADE) for multipath signals arriving at an antenna array," *IEEE Communications Letters*, vol. 1, no. 1, pp. 12–14, Jan. 1997.

[24] M. C. Vanderveen, "Estimation of parametric channel models in wireless communication networks," Ph.D. dissertation, Stanford university, 1997.

[25] G. G. Raleigh and T. Boros, "Joint space-time parameter estimation for wireless communication channels," *IEEE Transactions on Signal Processing*, vol. 46, no. 5, pp. 1333–1343, May 1998.

[26] R. Roy and T. Kailath, "ESPRIT - estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 7, pp. 984–995, Jul. 1989.

[27] S. M. Kay, *Fundamentals of statistical signal processing : estimation theory.* Englewood Cliffs: PTR Prentice Hall, 1993.

[28] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society - Series B*, vol. 39, no. 1, pp. 1–38, Dec. 1977.

[29] M. Feder and E. Weinstein, "Parameter estimation of superimposed signals using the EM algorithm," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 4, pp. 477–489, Apr. 1988.

[30] J. A. Fessler and A. O. Hero, "Space-alternating generalized expectation-maximization algorithm," *IEEE Transactions on Signal Processing*, vol. 42, no. 10, pp. 2664–2677, Oct. 1994.

[31] B. H. Fleury, M. Tschudin, R. Heddergott, D. Dahlhaus, and K. I. Pedersen, "Channel parameter estimation in mobile radio environments using the SAGE algorithm," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 3, pp. 434–450, Mar. 1999.

[32] N. Chotikakamthorn, N. Chotikakamthorn, and J. A. Chambers, "IQML algorithm for multiple signal parameter estimation," *IEE Proceedings of Radar, Sonar and Navigation*, vol. 144, no. 5, pp. 237–244, Oct. 1997.

[33] H. Krim and M. Viberg, "Two decades of array signal processing research: The parametric approach," *IEEE Signal Processing Magazine*, vol. 13, no. 4, pp. 67–94, 1996.

[34] S. Haykin, J. P. Reilly, V. Kezys, and E. Vertatschitsch, "Some aspects of array signal processing," *IEE Proceedings of Radar and Signal Processing*, vol. 139, no. 1, pp. 1–26, Feb. 1992.

[35] A. Hero, H. Messer, J. Goldberg, D. J. Thomson, M. G. Amin, G. Giannakis, A. Swami, J. K. Tugnait, A. Nehorai, A. L. Swindlehurst, J. F. Cardoso, T. Lang, and J. Krolik, "Highlights of statistical signal and array processing," *IEEE Signal Processing Magazine*, vol. 15, no. 5, pp. 21–64, Sep. 1998.

[36] M. Tschudin, C. Brunner, T. Kurpjuhn, M. Haardt, and J. A. Nossek, "Comparison between unitary ESPRIT and SAGE for 3-d channel sounding," in *49th IEEE Vehicular Technology Conference*, vol. 2, Jul. 1999, pp. 1324–1329.

[37] M. D. Zoltowski, M. Haardt, and C. P. Mathews, "Closed-form 2-d angle estimation with rectangular arrays in element space or beamspace via unitary ESPRIT," *IEEE Transactions on Signal Processing*, vol. 44, no. 2, pp. 316–328, Feb. 1996.

[38] M. Haardt and J. A. Nossek, "Unitary ESPRIT: How to obtain increased estimation accuracy with a reduced computational burden," *IEEE Transactions on Signal Processing*, vol. 43, no. 5, pp. 1232–1242, May 1995.

[39] A.-J. van der Veen, M. C. Vanderveen, and A. Paulraj, "Joint angle and delay estimation using shift-invariance techniques," *IEEE Transactions on Signal Processing*, vol. 46, no. 2, pp. 405–418, Feb. 1998.

[40] B. A. Constantine and I. I. Panayiotis, *Introduction to Smart Antennas*. San Rafael, CA, USA: Morgan and Claypool Publishers, 2007.

[41] J.-S. Jiang, "Measurement, modeling, and performance, of indoor mimo channels," Ph.D. dissertation, Georgia Institute of Technology, Jul. 2004.

[42] Mathworks, "MATLAB - the language of technical computing," Aug. 2010. [Online]. Available: http://www.mathworks.co.uk/products/matlab/

[43] T.-J. Shan, M. Wax, and T. Kailath, "On spatial smoothing for direction-of-arrival estimation of coherent signals," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 33, no. 4, pp. 806–811, Aug. 1985.

[44] Y.-M. Chen, "On spatial smoothing for two-dimensional direction-of-arrival estimation of coherent signals," *IEEE Transactions on Signal Processing*, vol. 45, no. 7, pp. 1689–1696, Jul. 1997.

[45] D. A. Linebarger, R. D. DeGroat, and E. M. Dowling, "Efficient direction-finding methods employing forward/backward averaging," *IEEE Transactions on Signal Processing*, vol. 42, no. 8, pp. 2136–2145, Aug. 1994.

[46] R. Bachl, "The forward-backward averaging technique applied to TLS-ESPRIT processing," *IEEE Transactions on Signal Processing*, vol. 43, no. 11, pp. 2691–2699, Nov. 1995.

[47] S. U. Pillai and B. H. Kwon, "Forward/backward spatial smoothing techniques for coherent signal identification," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 1, pp. 8–15, Jan. 1989.

[48] B. Ottersten and T. Kailath, "Direction-of-arrival estimation for wide-band signals using the ESPRIT algorithm," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 2, pp. 317–327, Feb. 1990.

[49] A. Richter, "Estimation of radio channel parameters: Models and algorithms," Ph.D. dissertation, TU Ilmenau, 2005.

[50] M. Haardt and J. A. Nossek, "Simultaneous schur decomposition of several nonsymmetric matrices to achieve automatic pairing in multidimensional harmonic retrieval problems," *IEEE Transactions on Signal Processing*, vol. 46, no. 1, pp. 161–169, Jan. 1998.

[51] M. C. Vanderveen, A.-J. van der Veen, and A. Paulraj, "Estimation of multipath parameters in wireless communications," *IEEE Transactions on Signal Processing*, vol. 46, no. 3, pp. 682–690, Mar. 1998.

[52] J. Picheral and U. Spagnolini, "Parametric estimation of space-time channels with spatially correlated noise by JADE-ESPRIT," in *7th International Symposium on Signal Processing and Its Applications*, vol. 2, Jul. 2003, pp. 415–418.

[53] M. A. Hernandez, L. Genis, R. Calders, and G. Janssen, "Subspace based estimation of parameters and linear space-time multiuser detection for WCDMA systems," in *6th International Symposium on Spread Spectrum Techniques and Applications*, vol. 1, Sep. 2000, pp. 249–253.

[54] H. L. Van Trees, *Detection, Estimation, and Modulation Theory, Part IV: Optimum Array Processing.* NY, USA: John Wiley & Sons, 2002.

[55] J. Li, J. Conan, and S. Pierre, "Joint estimation of channel parameters for MIMO communication systems," in *2nd International Symposium on Wireless Communication Systems*, Sep. 2005, pp. 22–26.

[56] M. Wax and A. Leshem, "Joint estimation of time delays and directions of arrival of multiple reflections of a known signal," *IEEE Transactions on Signal Processing*, vol. 45, no. 10, pp. 2477–2484, Oct. 1997.

[57] T. K. Moon, "The expectation-maximization algorithm," *IEEE Signal Processing Magazine*, vol. 13, no. 6, pp. 47–60, Nov. 1996.

[58] N. Cadalli and O. Arikan, "Wideband maximum likelihood direction finding and signal parameter estimation by using the tree-structured EM algorithm," *IEEE Transactions on Signal Processing*, vol. 47, no. 1, pp. 201–206, Jan. 1999.

[59] C. M. Tan, M. A. Beach, and A. R. Nix, "Multipath parameters estimation with a reduced complexity unitary-SAGE algorithm," in *5th European Conference on Personal Mobile Communications*, vol. 14, no. 6, Apr. 2003, pp. 515–528.

[60] C. C. Chong, D. I. Laurenson, C. M. Tan, S. McLaughlin, M. A. Beach, and A. R. Nix, "Joint detection-estimation of directional channel parameters using the 2-d frequency domain SAGE algorithm with serial interference cancellation," *IEEE International Conference on Communications*, vol. 2, pp. 906–910, Aug. 2002.

[61] B. H. Fleury, D. Dahlhaus, R. Heddergott, and M. Tschudin, "Wideband angle of arrival estimation using the SAGE algorithm," *4th IEEE International Symposium on Spread Spectrum Techniques and Applications*, vol. 1, pp. 79–85, Sep. 1996.

[62] K. I. Pedersen, B. H. Fleury, and P. E. Mogensen, "High resolution of electromagnetic waves in time-varying radio channels," *8th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, vol. 2, pp. 650–654, Sep. 1997.

[63] B. H. Fleury, P. Jourdan, and A. Stucki, "High-resolution channel parameter estimation for MIMO applications using the SAGE algorithm," in *International Zurich Seminar on Broadband Communications*, vol. 30, Feb. 2002, pp. 1–9.

[64] A.-J. van der Veen, "Matlab implementation of SI-JADE," 1997. [Online]. Available: http://ens.ewi.tudelft.nl/pubs/JADE.shar

[65] P. Stoica and A. Nehorai, "MUSIC, maximum likelihood, and cramer-rao bound," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 5, pp. 720–741, May 1989.

[66] Y. Fu and Z. Tian, "Cramer-rao bounds for hybrid TOA/DOA-based location estimation in sensor networks," *IEEE Signal Processing Letters*, vol. 16, no. 8, pp. 655–658, Aug. 2009.

[67] G. H. Golub and C. F. Van Loan, *Matrix computations*, 3rd ed. Baltimore, MD, USA: Johns Hopkins University Press, 1996.

[68] P. Comon and G. H. Golub, "Tracking a few extreme singular values and vectors in signal processing," *Proceedings of the IEEE*, vol. 78, no. 8, pp. 1327–1343, Aug. 1990.

[69] W. Utschick, "Tracking of signal subspace projectors," *IEEE Transactions on Signal Processing*, vol. 50, no. 4, pp. 769–778, Apr. 2002.

[70] H. Zamiri-Jafarian and S. Pasupathy, "EM-based recursive estimation of channel parameters," *IEEE Transactions on Communications*, vol. 47, no. 9, pp. 1297–1302, Sep. 1999.

[71] P. J. Chung and J. F. Bohme, "Recursive EM algorithm for stochastic ML DOA estimation," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, 2002, pp. 3029–3032.

[72] ——, "Recursive EM and SAGE-inspired algorithms with application to DOA estimation," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 2664–2677, Aug. 2005.

[73] H. Bolcskei, D. Gesbert, C. Papadias, and A. van der Veen, *Space-time Wireless Systems: From Array Processing to MIMO Communications.* Cambridge University Press, 2006.

[74] W. Li, W. Yao, and P. Duffett-Smith, "Comparative study of joint TOA/DOA estimation techniques for mobile positioning applications," in *6th IEEE Consumer Communications and Networking Conference*, Jan. 2009, pp. 1–5.

[75] R. M. Buehrer and S. P. Nicoloso, "Comments on "partial parallel interference cancellation for CDMA"," *IEEE Transactions on Communications*, vol. 47, no. 5, pp. 658–661, May 1999.

[76] D. Divsalar, M. K. Simon, and D. Raphaeli, "Improved parallel interference cancellation for CDMA," *IEEE Transactions on Communications*, vol. 46, no. 2, pp. 258–268, Feb. 1998.

[77] G. Xue, J. Weng, L. Tho, and S. Tahar, "Adaptive multistage parallel interference cancellation for CDMA," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 10, pp. 1815–1827, Oct 1999.

[78] H. H. Monson, *Statistical Digital Signal Processing and Modeling.* NY, USA: John Wiley & Sons, 1996.

[79] P. S. R. Diniz, *Adpative Filtering - Algorithms and Practical Implementation.* Springer, 2008.

[80] D. B. Fogel, "An introduction to simulated evolutionary optimization," *IEEE Transactions on Neural Networks*, vol. 5, no. 1, pp. 3–14, Jan 1994.

[81] H. G. Beyer, *The Theory of Evolution Strategies.* NY, USA: Springer-Verlag, 2001.

[82] ——, "Toward a theory of evolution strategies: The (μ, λ)-theory," *Evolutionary Computation*, vol. 2, no. 4, pp. 381–407, 1994.

[83] T. Bäck and F. Hoffmeister, "Basic aspects of evolution strategies," *Statistics and Computing*, vol. 4, no. 2, pp. 51–63, 1994.

[84] A. P. Engelbrecht, *Computational Intelligence: An Introduction.* Wiley Publishing, 2007.

[85] T. Bäck, *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms.* Oxford, UK: Oxford University Press, 1996.

[86] H. G. Beyer and H. P. Schwefel, "Evolution strategies - a comprehensive introduction," *Natural Computing: an international journal*, vol. 1, no. 1, pp. 3–52, May 2002.

[87] H. P. Schwefel, *Evolution and Optimum Seeking: The Sixth Generation.* New York, NY, USA: John Wiley & Sons, 1993.

[88] T. Bäck, F. Hoffmeister, and H. P. Schwefel, "A survey of evolution strategies," in *4th International Conference on Genetic Algorithms*, 1991, pp. 2–9.

[89] M. Chang, K. Ohkura, and K. Ueda, "Some experimental observations of $(\mu/\mu, \lambda)$-evolution strategies," in *Congress on Evolutionary Computation*, vol. 1, May 2001, pp. 663–670.

[90] H. G. Beyer, "Toward a theory of evolution strategies: Self-adaptation," *Evolutionary Computation*, vol. 3, no. 3, pp. 311–347, 1995.

[91] Y. Jin and J. Branke, "Evolutionary optimization in uncertain environments-a survey," *IEEE Transactions on Evolutionary Computation*, vol. 9, no. 3, pp. 303–317, Jun. 2005.

[92] H. G. Beyer, "Evolutionary algorithms in noisy environments: Theoretical issues and guidelines for practice," in *Computer Methods in Applied Mechanics and Engineering*, 1998, pp. 239–267.

[93] ——, "Toward a theory of evolution strategies: On the benefits of sex - the $(\mu/\mu, \lambda)$ theory," *Evolutionary Computation*, vol. 3, no. 1, pp. 81–111, 1995.

[94] U. Sarac, F. K. Harmanci, and T. Akgul, "Experimental analysis of detection and localization of multiple emitters in multipath environments," *IEEE Antennas and Propagation Magazine*, vol. 50, no. 5, pp. 61–70, Oct. 2008.

[95] I. Wax, M. Ziskind, "Detection of the number of coherent signals by the MDL principle," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, no. 8, pp. 1190–1196, Aug. 1989.

[96] M. Wax, "Detection and localization of multiple sources via the stochastic signals model," *IEEE Transactions on Signal Processing*, vol. 39, no. 11, pp. 2450–2456, Nov. 1991.

[97] C. Y. Lee and E. K. Antonsson, "Dynamic partitional clustering using evolution strategies," in *26th IEEE Annual Conference of the Industrial Electronics Society*, vol. 4, Oct. 2000, pp. 2716–2721.

[98] C. Y. Lee, "Variable length genomes for evolutionary algorithms," in *Genetic and Evolutionary Computation Conference.* Morgan Kaufmann, 2000, p. 806.

[99] T. Shirazi, A. S. Seyedena, "Design of MLP using evolutionary strategy with variable length chromosomes," in *ACIS 9th International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, Aug. 2008, pp. 664–669.

[100] F. Tong, X. M. Xu, and D. S. Chen, "Blind channel equalization with order estimation based on evolutionary algorithm," in *International Conference on Wireless Communications, Networking and Mobile Computing*, Sept. 2006, pp. 1–3.

[101] D. V. Arnold and A. S. Castellarin, "A novel approach to adaptive isolation in evolution strategies," in *11th Annual conference on Genetic and evolutionary computation*, 2009, pp. 491–498.

[102] G. Rudolph, "Global optimization by means of distributed evolution strategies," in *1st Workshop on Parallel Problem Solving from Nature*, 1991, pp. 209–213.

[103] H. G. Beyer, M. Dobler, C. Hämmerle, and P. Masser, "On strategy parameter control by meta-ES," in *11th Annual conference on Genetic and evolutionary computation*, 2009, pp. 499–506.

[104] D. V. Arnold and A. MacLeod, "Hierarchically organised evolution strategies on the parabolic ridge," in *8th annual Conference on Genetic and evolutionary computation*, 2006, pp. 437–444.

[105] D. V. Arnold and H. G. Beyer, *Noisy Local Optimization with Evolution Strategies*. Norwell, MA, USA: Kluwer Academic Publishers, 2002.

[106] ——, "Local performance of the (1 + 1)-ES in a noisy environment," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 30–41, Feb. 2002.

[107] L. Kallel, B. Naudts, and A. Rogers, *Theoretical aspects of evolutionary computing*. London, UK: Springer-Verlag, 2001.

[108] A. I. Oyman, H. G. Beyer, and H. P. Schwefel, "Where elitists start limping evolution strategies at ridge functions," in *5th International Conference on Parallel Problem Solving from Nature*, 1998, pp. 34–46.

[109] G. Stimson, *Introduction to Airborne Radar, 2nd Edition*. Mendham, NJ, USA: SciTech Publishing, 1998.

[110] P. C. Chestnut, "Emitter location accuracy using TDOA and differential doppler," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 18, no. 2, pp. 214–218, Mar. 1982.

[111] Y. T. Chan and J. J. Towers, "Passive localization from doppler-shifted frequency measurements," *IEEE Transactions on Signal Processing*, vol. 40, no. 10, pp. 2594–2598, Oct. 1992.

[112] Y. T. Chan and F. L. Jardine, "Target localization and tracking from doppler-shift measurements," *IEEE Journal of Oceanic Engineering*, vol. 15, no. 3, pp. 251–257, Jul. 1990.

[113] R. J. Webster, "An exact trajectory solution from doppler shift measurements," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 18, no. 2, pp. 249–252, Mar. 1982.

[114] D. J. Torrieri, "Statistical theory of passive location systems," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 20, no. 2, pp. 183–198, Mar. 1984.

[115] Q. Wan, W. L. Yang, and Y. N. Peng, "Mobile localization using doppler symmetric constraint in case of non-line-of-sight," in *IEEE Global Telecommunications Conference*, 2004.

[116] A. Goldsmith, *Wireless Communications*. New York, NY, USA: Cambridge University Press, 2005.

[117] P. Lacomme, J. Hardange, J. Marchais, and E. Normant, *Air and Spaceborne Radar Systems - An Introduction*. Norwich, NY, USA: William Andrew Publishing, 1998.

[118] H. L. V. Trees, *Detection, Estimation, and Modulation Theory: Radar-Sonar Signal Processing and Gaussian Signals in Noise*. Melbourne, FL, USA: Krieger Publishing, 1992.

[119] H. G. Beyer and B. Sendhoff, "Evolution strategies for robust optimization," in *IEEE World Congress on Computational Intelligence*, 2006, pp. 1346–1353.

[120] Y. Rui and D. Florencio, "New direct approaches to robust sound source localization," in *International Conference on Multimedia and Expo*, vol. 1, Jul. 2003, pp. 737–740.