

UNIVERSITY OF OSLO
Department of Informatics

**High Resolution
Beamforming of
SIMRAD EM3000
Bathymetric
Multibeam Sonar
Data**

Are Rønhovde

Cand Scient thesis

October 1999



For Emily, whenever I may find her

First of all I would like to thank Suhail Younis for telling me about this thesis. If it had not been for Suhail I would probably not have written any of the following. Next, thank you Stian Henriksen, Børre Bjerkholt and Erich Suter for both lunches and discussions about writing. I would also like to thank Jens T. B. Thielemann, both for discussions related to the thesis, and non-thesis discussions. I also appreciated the time I spent at the barrack, while working with the thesis I met several interesting people there.

Much appreciation goes to Luren Yang and Torfinn Taxt, my guides into the unknown. We had our disagreements, but in the end it seems that it was the struggle that made it worth the trip. I have learned a lot.

I would also like to thank my family, especially my parents Evy and Jostein and my brother Frode, for their understanding and guidance during the time I have worked with the thesis.

Finally, thank you Linda! Without you I would never have ended up at the University of Oslo in the first place.

Are Rønhovde
October 1999

Contents

Summary	vii
1 Introduction	1
2 Wave Theory	7
2.1 Wave Propagation	7
2.2 Spatial Resolution	9
2.3 Transition between Nearfield and Farfield	10
2.4 Sampling	12
2.5 Array and Signal Model	12
3 Bathymetric Sonars	15
3.1 The Sidescanning Sonar	16
3.2 The Multibeam Sonar	17
4 Estimation of Covariances	21
4.1 The Covariance Matrix	21
4.2 Noise and Artifacts	23
4.3 Time Averaging	24
4.4 Spatial Smoothing	24
4.5 Forward Backward Smoothing	25
4.6 Noise Subtraction	26
5 Beamforming	29
5.1 Conventional Beamformer	31
5.2 High Resolution Beamformers	32
6 Data and Evaluation Methods	43
6.1 Data Sets	43
6.2 Implementation Issues	44
6.3 Evaluation Methods	46
6.3.1 Quantitative Evaluation	47
6.3.2 Qualitative Comparison	54
7 Experimental Results	57
7.1 Quantitative Observations	57
7.1.1 Minimum-Variance	58
7.1.2 Eigen-Vector	60
7.1.3 MUSIC	62

7.1.4	Minimum-Norm	66
7.1.5	Root-MUSIC	69
7.1.6	Alternative root-MUSIC	71
7.1.7	ESPRIT	74
7.2	General Observations	76
7.3	Comparing the Best Results	78
7.4	Qualitative Observations	80
7.4.1	Spectra	80
7.4.2	After bottom detection	87
7.4.3	Summary of the Observations	99
8	Discussion	101
8.1	Stationarity Problems: Causes and Cures	102
8.1.1	Violating the Stationarity Criterion	102
8.1.2	Along-track Resolution	102
8.2	Sea-bed Imaging	103
8.3	Bottom Detection	103
8.4	Where do we go?	104
A	Notation and Abbreviations	107
B	Paper written for Oceans '99	109

List of Figures

2.1	Sensors in the near and farfield.	11
2.2	A receive array in its coordinate system.	13
3.1	A bathymetric sonar at work.	16
3.2	The beam plane of a sidescanning sonar.	17
3.3	Geometry of the arrays and their beamform for a multibeam sonar.	18
4.1	Array split into several subarrays.	25
4.2	The effect of forward-backward averaging.	26
5.1	Effect of angular increments when trying to resolve two closely spaced sources.	30
6.1	Definition of echo incidence areas for quantitative evaluation.	45
6.2	Spectrum in polar coordinates, before and after bottom detection.	45
6.3	Explanation of different FWHM approaches.	47
6.4	Two different power spectra for the same signal.	48
6.5	Different averages of FWHM and width-location overlap	49
6.6	Flowchart for quantitative evaluation.	51
7.1	<i>PWR</i> and <i>WLO</i> results for Minimum-Variance.	59
7.2	<i>M-flops</i> for Minimum-Variance and Eigen-Vector.	62
7.3	<i>PWR</i> and <i>WLO</i> results for MUSIC.	63
7.4	<i>PWR</i> and <i>WLO</i> results for Minimum-Norm.	66
7.5	<i>WLO</i> and <i>WLO(M)</i> results for root-MUSIC.	70
7.6	<i>WLO</i> and <i>WLO(M)</i> results for the alternative root-MUSIC.	72
7.7	<i>WLO</i> results for ESPRIT.	74
7.8	Spectra after a log transform.	81
7.9	Spectra after a log transform.	82
7.10	Spectra after a log transform.	83
7.11	Spectra after a log transform.	84
7.12	Spectra after a log transform.	85
7.13	Spectra after a log transform.	86
7.14	Spectra after a simple bottom detection algorithm.	87
7.15	Spectra after a simple bottom detection algorithm.	88
7.16	Spectra after a simple bottom detection algorithm.	90
7.17	Spectra after a simple bottom detection algorithm.	91
7.18	Spectra after a simple bottom detection algorithm.	92
7.19	Spectra after a simple bottom detection algorithm.	93

7.20 A small area covering the time gap.	95
7.21 Effect of trying to estimate M in a smarter way.	97
7.22 Effect of trying to estimate M in a smarter way.	98

List of Tables

5.1	Beamforming methods	30
6.1	Parameter combinations used for beamforming	54
7.1	The best results for the Minimum-Variance algorithm.	60
7.2	The best results for the Eigen-Vector algorithm.	61
7.3	The best results for the MUSIC algorithm.	65
7.4	The best results for the Minimum-Norm algorithm.	68
7.5	The best results for the root-MUSIC algorithm.	71
7.6	The best results for the alternative root-MUSIC algorithm.	73
7.7	The best results for the ESPRIT algorithm.	75
7.8	The best results achieved for each method.	79
A.1	Abbreviations and the most important quantities.	108

Summary

Originally, beamforming meant forming the transmit beam for some purpose. Today, we use the word beamforming both for the transmit – and the receive beam. The beam is simply the mainlobe of the array's radiation – or response pattern. Changing this pattern, either by weighting the array elements or by steering its direction, is what beamforming is all about.

In most signal processing applications, there is a tradeoff between different needs. The tradeoff in bathymetric sonars is between cross-track coverage, accuracy and resolution. The goal of this thesis, is to improve the resolution of a bathymetric sonar, working only with the output of the array – after the tradeoff is done. To achieve this, the Fourier transform method for beamforming and some high-resolution, also called adaptive, methods have been studied.

In this thesis, the performance of several high-resolution beamformers are studied: **Minimum-Variance, Eigen-Vector, MUSIC, Minimum-Norm, root-MUSIC and ESPRIT**. The conventional Fourier transform, used for beamforming in many array systems today, suffers from low resolution. It has low ability of resolving closely spaced sources. For bathymetric sonars, this may lead to incorrect mapping of the sea-floor.

Different ways of estimating the spatial covariance matrix are investigated, and a method aimed to reduce structured noise in this matrix is developed.

Some quantitative measures are developed and used as part of the evaluation of the high-resolution beamformers. The measures show that estimation of the spatial covariance matrix has a large impact on the performance of the beamformers, and that most methods perform better when the structured noise in this matrix is reduced.

An alternative approach for a high-resolution beamformer (root-MUSIC) is also developed, and is found to perform better than the one given in literature.

It is found that among the high-resolution beamformers studied here, MUSIC and ESPRIT perform best with regard to the quantitative measures used for evaluation. These results are supported by visual examination of different spectra for the beamformers.

Finally, some interesting research areas based on the observations in the thesis are proposed, and a new way of obtaining bathymetric measures is suggested.

Chapter 1

Introduction

Throughout history, mapping of the sea floor has been important for all coastal regions. At first only for safe navigation, it is now important for navigation as well as for positioning of offshore installations such as oil platforms or oil and gas pipes. The mapping is done through bathymetry, the measurement of depths of water in oceans, seas, and lakes. Historically, depths were measured by using a lead-line, later on single beam echo-sounders were used and presently multibeam echo-sounders are used. Echo-sounders are also known as sonars - SOund NAvigation Ranging.

Sonars use different beamforming methods to obtain the bathymetric measures. Originally, beamforming meant forming the transmit beam for some purpose. Today, we use the word beamforming both for the transmit - and the receive beam. The beam is simply the mainlobe of the array's radiation - or response pattern. Changing this pattern, either by weighting the array elements or by steering its direction, is what beamforming is all about [7, 8, 28].

The main problem in bathymetric mapping is to achieve a good resolution while maintaining the positional accuracy of the bottom, preferably obtaining the bathymetric measures in real time. In addition, it is important to keep the costs as low as possible, which means mapping a widest possible cross track [18]. As in most signal processing applications, there is a tradeoff between different needs. The tradeoff in bathymetric sonars is between cross-track coverage, accuracy and resolution. Currently, the most important bathymetric sonars are of two types: sidescan sonars and multibeam sonars.

Sidescan sonars can map very broad sectors of the sea floor. This property together with their low price make them economic in use, but they are not as accurate in determining the origin of the echoes as multibeam sonars [3, 9]. For rough sea floors a sidescan sonar will fail to give accurate bathymetric positions [3, 9]. The failure is due to the low number of sensors in the sidescan sonar. A typical sidescan sonar only measures the incoming waves at two positions. This restricts it from computing the origin of more than one echo at each time instant.

Multibeam sonars often have less spatial resolution capability¹, map smaller (but still wide) sectors, but have a higher positional accuracy than sidescan sonars [3, 9]. Multibeam sonars are preferred when the sea floor is rough, such as when navigating along an underwater canyon, a pipe or a cable [6]. Under such conditions only the multibeam sonar is capable of computing reliable positions of all bottom features yielding echoes at each time instant.

Since multibeam sonars have less resolution capability than sidescan sonars, it is important to improve the resolution of the former sonars. Note that we are actually talking about two different definitions of resolution. The measurements of a sidescan sonar will always be corrupted in the event of multiple scatters [9], while where bottom conditions are simple the attainable resolution is only limited by the pulse length. A short pulse insonifies² a small area of the bottom, giving a high spatial resolution [3]. The resolution of the sidescan sonar is thus defined as the ability to determine a single plane wave's direction of propagation.

The measurements of a multibeam sonar will only be corrupted in the event where two scatters are too closely spaced to be resolved. The multibeam sonar's ability to resolve closely spaced sources is what we mean by its resolution. A high resolution is essential, since one can then resolve closely spaced objects or bottom features whose echoes impinge on the sonar simultaneously.

The most commonly used method for multibeam beamforming, is the Fourier transform method. This method can be efficiently implemented as the Fast Fourier transform, but is inadequate in resolving closely spaced synchronous echoes [19]. The goal of this thesis, is to improve the resolution of a bathymetric sonar, working only with the output of the array - after the tradeoff between cross-track coverage, accuracy and resolution is done. That is, *we want to improve the resolution without changing the physical properties of the sonar.*

To improve the bathymetric results of the multibeam sonars, several high-resolution beamforming methods have been proposed [4, 11, 18, 19, 26, 27, 28, 29]. Among these are both spectral based and parametric methods as denoted by Krim and Viberg [11]. This distinction will also be used in the thesis, since it immediately tells us what different approaches have been used to compute the bathymetric positions (directions of arrival).

Spectral based methods are based on a spectrum computed over all possible angles (the parameters of interest for bathymetric sonars) of incoming waves, and pick the angles in the spectrum with largest magnitude as the Directions Of Arrival (*DOA*) of the propagating waves [11]. The spectral based methods tested here are: Minimum Variance [11, 7], Eigen Vector [7], MUSIC [11, 13, 18, 22, 24, 27] and Minimum Norm [11, 13, 24, 27, 31]. Minimum-Variance was chosen since it, as the oldest adaptive method, serves as a gateway to modern high-resolution methods. From it, subspace based methods such as Eigen-Vector, MUSIC and Minimum-Norm evolved. Eigen-Vector is little used. But, when the number of signals is chosen too low, it only

¹Applying a beamwidth based resolution.

²This word, resembling illuminate, was introduced in literature by de Moustier [3].

degrades to the Minimum-Variance algorithm, whereas MUSIC will degrade in a much wider sense (Sec. 5.2). MUSIC will “whiten” some signals, and regard them as noise. Eigen-Vector will also regard some signals as noise, but they will not be “whitened” and should still be seen after beamforming. Minimum-Norm was chosen because it should exhibit a lower bias and better resolution than MUSIC [11, 13].

These methods have the potential of higher resolution than the Fourier Transform method, but may still not exhibit more than one peak in the spectrum although there are two distinct, but closely spaced, sources [22].

Parametric methods require a simultaneous search for all parameters of interest (ie the directions of arrival), at the expense of an increased computational complexity. However, for Uniform Linear Arrays the increased computational complexity can be avoided without loss of performance [11].

The parametric methods studied in this thesis are Root-MUSIC [4, 11, 13, 22, 27] and ESPRIT [11, 13, 23, 26]. Root-MUSIC is used to compare the difference in performance between spectral and parametric methods. In small samples root-MUSIC has empirically been found to perform better than the spectral form [11]. Since root-MUSIC is superior to root-Minimum-Norm [10, 27], only root-MUSIC is studied here. ESPRIT was chosen to see if this computationally efficient algorithm performs equally well as root-MUSIC. Maximum Likelihood techniques are not studied since they require accurate initial estimates. Such estimates are often achieved by first applying spectral based methods. Thus, the computational load would be significant.

For the parametric methods, Krim stated that: “*coherent signals impose no conceptual difficulties for parametric methods*” c.f. [11]. Rao and Hari [22] are not quite as bombastic, but find that some parametric methods have the potential of resolving closely spaced sources where spectral based methods fail. “*Spectral methods always have less resolution compared to Root forms*”, c.f. [22]. It is unclear whether Rao also thinks of ESPRIT as a Root form, but I interpret him as if he does. The difference in resolution is due to signals being assumed to have distinct z -plane roots in the root approaches, while in the spectral based methods a stronger assumption of distinct peaks in the spatial spectrum is assumed [22]. It is easy to distinguish roots that are close to each other, but in the spectral approaches one might just experience a single “blurred” peak.

Now, let us do a quick review of what has been found for some of the different methods. We will consider both general results for the high-resolution methods, and results indicating how to estimate the spatial covariance matrix to get the best results. All results have been based only on simulated data.

Minimum Variance’s accuracy improves with the signal to noise ratio (SNR), and its resolution characteristics can not be known other than by simulations when the number of snapshots (samples in time) is not much larger than the amount of sensors [7]. The resolution of Eigen Vector, MUSIC and Minimum Norm and their positional accuracy are limited by the amount of available data and the background noise characteristics [7].

Rao and Hari [23, 24] found that forward/backward smoothing was *always* preferable to forward-only smoothing³. It was also found that MUSIC should be smoothed with as few subarrays as possible, while Minimum-Norm and ESPRIT performed better for different numbers of subarrays, depending on whether the SNR was low or high and how many signals were present. That is, the L element sensor array should be split into different subarray sizes for different signal characteristics. One signal case: $(L + 1)/4$ subarrays for low SNR and $(L + 1)/3$ for high SNR. Two signal case: $(L + 1)/8$ subarrays for low SNR and $(L + 1)/5$ for high SNR.

Srinivas and Reddy [27] found that *“root-MUSIC performs better than root-min norm when the number of sensors exceeds two. ... the MSE with root-MUSIC is less than that of root-min norm. ... MUSIC is less sensitive than the min norm method when the number of sensors exceeds 2. ... MUSIC is more robust to errors. Finally, the performance of both the methods improves when the source spacing increases”*.

Li and Lu [13] summarizes their results like this:

- *“At low SNR, MUSIC has large bias which is comparable to the root mean-squared error (RMSE) while Min-Norm and ESPRIT have negligible bias compared to RMSE”*.
- *“When the number of sensors is small, MUSIC has the largest bias which is again comparable to RMSE, and bias of MUSIC decreases fastest with increase of number of sensors”*.
- *“The bias is not monotonically decreasing for MUSIC and Min-Norm with increase of number of sensors”*.
- *“Compared to RMSE, bias of MUSIC is very large at small number of snapshots while bias of Min-Norm and ESPRIT are always small enough to be neglected”*.

Earlier research on high-resolution beamforming has been based on simulations with several snapshots [4, 10, 27, 31]. It is important to see if spectral and parametric methods are applicable also in a real world bathymetric sonar scenario. This is the central topic of this thesis. In real world scenarios, generally, only one snapshot is available for estimating the directions of propagating signals.

The rest of the thesis is organized as follows :

Chapter 2 will help the reader understand the remaining chapters.

- Basic wave theory is presented.
- The model which the beamforming methods are based on is given.

Chapter 3 Knowledge of sonars, especially multibeam sonars, motivates for further improvements of the current methods used for bathymetric mapping. The sidescanning sonar and the multibeam sonar will be further explained.

³These smoothing approaches are discussed in chapter 4.

Chapter 4 treats different ways of dealing with noise and correlated signals.

- Noise is always present, thus it is important to gain some insight about noise to avoid drawing the wrong conclusions when processing any signal.
- Correlated signals may also corrupt our insight if the right precautions are not taken.

Chapter 5 tells the reader about the similarities and differences among the treated methods. It explains both conventional and high-resolution methods.

Chapter 6 presents data sets, implementation issues and evaluation methods. It explains how the methods are evaluated in a meaningful way, even though there is no knowledge of the true values we wish to compare against.

Chapter 7 presents the experimental results of the different methods.

- The best way of estimating the covariance matrix is found for each beamforming method.
- Then the beamformers are compared for these special choices of covariance matrices.
- The beamformers are also compared visually, for the same covariance matrices.
- According to these evaluations, MUSIC and ESPRIT perform best.

Chapter 8 The discussion suggests some future research areas. Further, a new method for obtaining bathymetric measures is proposed.

Chapter 2

Wave Theory

This chapter gives the basic theory needed to understand the remaining chapters. An introduction to propagation of waves is given, followed by an explanation of the signal processing restrictions this propagation leads to.

2.1 Wave Propagation

Any wave propagating through a medium must obey the physical laws associated with that particular medium, and a set of boundary conditions. Acoustic waves in inviscid fluids are longitudinal. The particles move back and forth in the direction of propagation of the wave. The acoustic waves can be regarded as a series of compressions and rarefactions, caused by pressure changes in the fluid as it is compressed or expanded.

The law of conservation of mass relates density ρ and particle velocity \mathbf{v} .

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho \mathbf{v}) \quad (2.1)$$

Here ∇ is the Laplace operator.

$$\nabla = \frac{\partial}{\partial x} + \frac{\partial}{\partial y} + \frac{\partial}{\partial z} \quad (2.2)$$

Equation (2.1) states that the net influx of mass in a small volume equals the rate which the mass increases.

Sound propagating through a fluid causes only small adiabatic compressions and expansions of the fluid about its equilibrium density ρ_0 . Thus, the density can be expressed by a small perturbation ρ' about the equilibrium $\rho = \rho_0 + \rho'$. For the pressure we have $p = p_0 + p'$. The adiabatic equation of state, which says that the pressure is a function of ρ , $p(\rho)$, can be expressed through a Taylor expansion

$$p = p_0 + (\rho - \rho_0) \left(\frac{\partial p}{\partial \rho} \right)_{\rho_0} + \frac{1}{2} (\rho - \rho_0)^2 \left(\frac{\partial^2 p}{\partial \rho^2} \right)_{\rho_0} + \dots \quad (2.3)$$

We only keep the lowest order term, since the fluctuations ρ' are small, $\rho' \ll \rho_0$. By defining $s = (\frac{\rho - \rho_0}{\rho_0})$, (2.3) can be written as

$$p' = \rho_0 s \left(\frac{\partial p}{\partial \rho} \right)_{\rho_0} \quad (2.4)$$

This allows us to express the nonlinear (2.1) as a linearized equation.

$$\frac{\partial s}{\partial t} = -\nabla \cdot \mathbf{v} \quad (2.5)$$

Further, the particle velocity \mathbf{v} , caused by the pressure and density perturbations, is much smaller than the speed of sound.

By using Newton's second law, we can express the acceleration \mathbf{a} , caused by a small force $d\mathbf{f}$ acting on a small volume dV with small mass dm in the fluid as $d\mathbf{f} = \mathbf{a}dm$. If we consider a fluid, the mass dm equals ρdV , while $d\mathbf{f} = -\nabla p dV$. Thus, $d\mathbf{f} = \mathbf{a}dm$ can be expressed as

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = -\frac{1}{\rho} \nabla p(\rho) \quad , \quad (2.6)$$

where the left part is an expression for the acceleration \mathbf{a} . This can be further simplified by using $\rho' \ll \rho_0$, and in addition require $|\mathbf{v} \cdot \nabla \mathbf{v}| \ll \left| \frac{\partial \mathbf{v}}{\partial t} \right|$. We can then replace ρ with ρ_0 , and drop $\mathbf{v} \cdot \nabla \mathbf{v}$ in (2.6). Using the fact that p_0 is a constant, we can also replace ∇p with $\nabla p'$ in (2.6).

$$\rho_0 \frac{\partial \mathbf{v}}{\partial t} = -\nabla p' \quad (2.7)$$

Now, combine the divergence of (2.7) with the time derivative of (2.5) to obtain

$$\nabla^2 p' = \rho_0 \frac{\partial^2 s}{\partial t^2} \quad (2.8)$$

Equation (2.4) enables us to eliminate s

$$\nabla^2 p' = \frac{1}{c^2} \frac{\partial^2 p'}{\partial t^2} \quad , \quad (2.9)$$

where c^2 is defined as $\left(\frac{\partial p}{\partial \rho} \right)_{\rho_0}$.

Equation (2.9) is the spatial wave equation, for the three spatial dimensions and time. It is the basic equation for propagating waves.

Let $s(\mathbf{x}, t)$, with spatial coordinates $\mathbf{x} = [x \ y \ z]^T$ and time t , represent the pressure of sound at some point in space and time instead of p' . Then, a more familiar form of (2.9) is

$$\frac{\partial^2 s}{\partial x^2} + \frac{\partial^2 s}{\partial y^2} + \frac{\partial^2 s}{\partial z^2} = \frac{1}{c^2} \frac{\partial^2 s}{\partial t^2} \quad (2.10)$$

Equation (2.10) also governs how signals pass through a medium from a radiating source to some sensing device. Thus, combining information from the wave equation at several points in space, we may gain insight about source-position given knowledge about the waveform – and vice versa.

Assume the solution of the wave equation to be separable in all coordinates, ie

$$s(x, y, z, t) = f(x)g(y)h(z)p(t)$$

and that $s(\mathbf{x}, t)$ has a complex exponential form

$$s(x, y, z, t) = A \exp\{j(\omega t - k_x x - k_y y - k_z z)\} \quad (2.11)$$

with A some complex constant and k_x, k_y, k_z and ω real constants ($\omega \geq 0$).

The wave equation now becomes

$$k_x^2 s(x, y, z, t) + k_y^2 s(x, y, z, t) + k_z^2 s(x, y, z, t) = \frac{\omega^2 s(x, y, z, t)}{c^2}$$

Cancelling $s(x, y, z, t)$ gives us one constraint on the real constants of the complex form.

$$k_x^2 + k_y^2 + k_z^2 = \frac{\omega^2}{c^2}$$

With this constraint, signals of the form as in (2.11) satisfy the wave equation (2.10). By super-positioning complex exponentials any signal satisfy the wave equation [7].

$\mathbf{k} = [k_x \ k_y \ k_z]^T$ is the *wavenumber vector*. Its magnitude is the wavenumber, $k = |\mathbf{k}| = 2\pi/\lambda$, where $\lambda = 2\pi c/\omega$ is the wave length. The wavenumber tells us how many cycles in radians the propagating wave goes through per meter in the direction of propagation. The wave propagates one wavelength λ during one temporal period.

Sometimes, waves propagate through inhomogeneous media, such is the case in underwater acoustics. In the ocean, the sound velocity depends on pressure, and temperature, which changes with depth. This means that the waves will not propagate in a straight line, rather they will bend and follow the fastest way of propagation through the medium, following Snell's law. This phenomenon is known as ray bending.

2.2 Spatial Resolution

Spatial resolution is *the* question at stake in this thesis. It is one of the most important issues when trying to improve the bathymetric sonars. The resolution defines an array's ability to determine a plane wave's direction of propagation [7], but also the ability to separate two plane waves impinging synchronously on the array from slightly different directions [7, 11, 19]. These are fundamentally different measures. The first definition tells us how exact the direction of a wave is estimated, the other

gives the ability to distinguish between two sources. Both definitions are meaningful, but for different kinds of sonars. The resolution of the sidescan (Sec. 3.1) sonar is defined as the ability to determine a plane wave's direction of propagation. The multibeam (Sec. 3.2) sonar's ability to resolve closely spaced sources is what we mean by its resolution. The latter is the kind of resolution we want to measure in this thesis, even though the former is also measured.

Unfortunately, resolution is not necessarily easy to measure. The ability to determine a plane wave's direction of propagation is fairly straight forward to measure, it equals measuring the width of the array pattern's mainlobe. But, different measures of the array pattern apply at different times. We have both wavenumber resolution and angular resolution [7]. The ability to distinguish between two sources is not quite as easy to assess. Not only does this ability depend on the two wave numbers, or directions, the relative phases of the propagating waves are also important. This means that the resolution depends upon the signal characteristics, and these may vary from one sample in time to the other.

“Resolution generally refers to the ability to distinguish closely spaced signal sources. One typically refers to some spectral-like measure, which would exhibit peaks at the locations of the sources. Whenever there are two peaks near two actual emitters, the latter are said to be resolved. However, for parametric techniques, the intuitive notion of resolution is non-trivial to define in precise terms. The subspace-based approach relies on certain geometrical properties of the assumed data model, resulting in a resolution capability which (in theory) is not limited by the array aperture, provided that the data collection time and/or SNR are sufficiently large and assuming the data model accurately reflects the experimental scenario.” c.f. [11]. Therefore, real world data are needed to figure out how good the resolution of different beamformers actually are (Chap. 5). An acceptable measure of this resolution has to be developed to enable a qualified meaning of different beamformers' application to sonar bathymetry. Such a measure is developed in chapter 6.

2.3 Transition between Nearfield and Farfield

In the following all sources are assumed to radiate energy in a spherical manner.

In sonars we want to find the source of the radiating energy, for bathymetric sonars the source is some point on the sea floor. We need some sensing device. A collection of sensors is usually denoted an array, and may be for instance a cable (passive sonars), a 2D grid or even a 3D grid. A specific structure called the Uniform Linear Array (ULA) is a collection of sensors, laid out along one dimension, where all sensors have the same intersensor spacing. The distance between each of the L sensors is d .

For spherical waves, planar wave fronts can be assumed when the sensing device is far enough from the source (Fig. 2.1). “Far enough” needs to be defined. We want the waves impinging on the device of length $\mathcal{D} = Ld$ to add constructively.

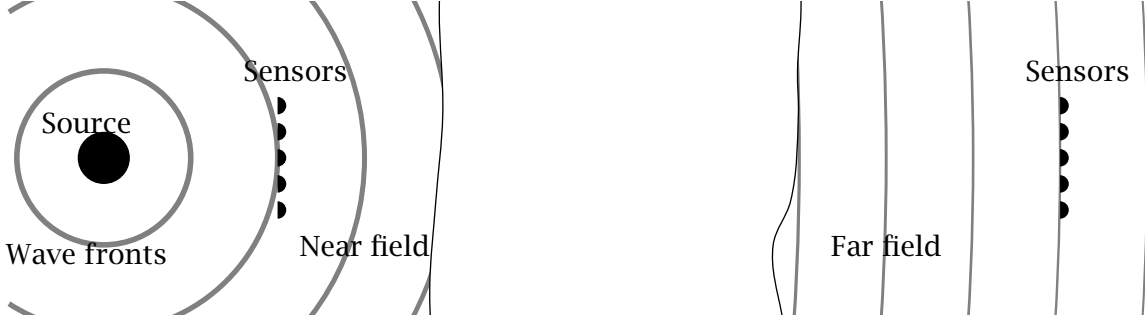


Figure 2.1: Sensors in the near and farfield.

This means that the wavefronts have to be fairly planar. The more planar, the more constructively they add. Define the maximum acceptable error to be $\lambda/8$, at the end of the device [30]. For a wave propagating from distance R , the formula for an error Δ at the point y on the array ($y \in [-\mathcal{D}/2, \mathcal{D}/2]$) becomes:

$$\begin{aligned}
 \Delta &= \sqrt{R^2 + y^2} - R \\
 &= R\sqrt{1 + \left(\frac{y}{R}\right)^2} - R \\
 &\cong \frac{y^2}{2R}
 \end{aligned} \tag{2.12}$$

By increasing R , we can reduce the error across the array. The largest error will occur at the end of the array, $y = \mathcal{D}/2$. We want $\Delta \leq \lambda/8$, thus

$$\begin{aligned}
 \frac{y^2}{2R} &\leq \lambda/8 \quad , \text{ let } y = \mathcal{D}/2 \\
 \frac{\mathcal{D}^2}{8R} &\leq \lambda/8 \\
 R &\geq \mathcal{D}^2/\lambda
 \end{aligned} \tag{2.13}$$

If the wave propagates from a point closer to the array than \mathcal{D}^2/λ , the error at the end of the array is larger than $\lambda/8$ and the waves will not add up constructively. Thus, we get a crossover from the nearfield to the farfield at a distance equal to \mathcal{D}^2/λ . All waves with origin further away than this distance will add constructively in the device. Since $\mathcal{D}^2/\lambda = L^2\lambda/4$, this means that for the *EM3000*¹, with $\lambda \approx 0.005\text{m}$ and $L = 80$ elements, the farfield begins at a range of approximately 8 meters. By using fewer sensors, the range to the farfield can efficiently be made smaller, since it is defined by the square of the number of sensors.

¹EM3000 is a 300kHz bathymetric multibeam Simrad sonar, used for the experiments in this thesis.

2.4 Sampling

In temporal sampling, the Nyquist theorem specifies [16]: For a signal with energy in the frequency area $|\Omega| < \Omega_0$, the sampling frequency Ω_s has to be more than twice the upper band frequency Ω_0 . Let T_s be the sampling period, and T_0 the period of the analog signal

$$\begin{aligned}\Omega_s &= \frac{2\pi}{T_s} > 2\Omega_0 \\ T_s &< T_0/2\end{aligned}$$

Unless we sample the signal more often than at half its period, some of the information needed to reconstruct the analog signal is lost – this is known as aliasing.

In the same manner as in temporal sampling, aliasing also occurs in spatial sampling. Therefore, we have to ensure that the distance d between neighboring sensors is small enough to avoid spatial aliasing. Consider a signal with spatial frequencies below k_0 , $|k| \leq k_0$. For the spatial sampling frequency, k_s , we must have

$$\begin{aligned}k_s &= \frac{2\pi}{d} \geq 2k_0 \\ d &\leq \frac{2\pi}{2k_0} = \frac{\pi}{2\pi/\lambda_0} = \frac{\lambda_0}{2}\end{aligned}\tag{2.14}$$

to ensure that spatial aliasing is avoided. The signal can be reconstructed without loss of information if we sample at positions spaced less than half the wave length apart [7].

Spatial aliasing may lead to collection of signal energy in other directions than the one of interest. This means that if a signal is propagating from a direction that we unwillingly and perhaps unknowingly collect energy in, the energy collected here will appear as if it is collected in the specified direction. This leads us to observe that there are signals present in the *direction of look*², even though no signals are actually present in that direction.

2.5 Array and Signal Model

An array is a collection of sensors used to sample signals in space. Each sensor, l , has a directional sensitivity determined by the physical form (aperture) of the sensor element. In the case of a *linear aperture*³ extended along the x -axis from $-D/2$ til $D/2$ and with response equal to 1 over the entire aperture, the aperture function or directional sensitivity of the sensor is of the form:

$$g_l(\vec{k}) = \frac{\sin(k_x D/2)}{k_x D/2}\tag{2.15}$$

²see (2.17) and (2.18)

³a sensor/element that gathers energy over a finite one dimensional area

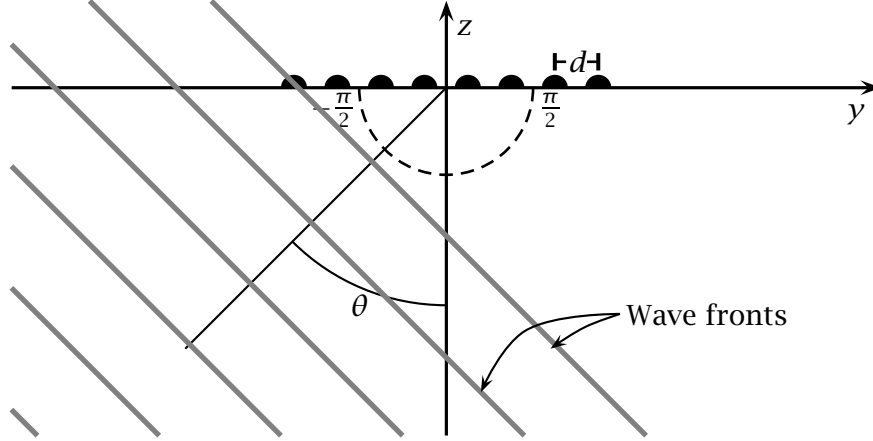


Figure 2.2: A receive array in its coordinate system, transmit array centered along the x -axis.

Such a directional sensitivity ensures that the response of the sensor is at its largest for signals emanating from the y - z plane normal to and running through the sensor. Thus, the collected energy is mainly from the y - z plane (Fig. 2.2).

In this thesis, ϕ is defined as the angle between the x -axis and the z -axis. Further, θ is the angle between the z and y -axis, with $\theta = 0$ when lying on the z -axis (Fig. 2.2).

The output, $y_l(t)$, of sensor l placed on the y -axis, but oriented alongside the x -axis, can now be modeled as:

$$y_l(t) = g_l(\phi) e^{-jk y_l \sin \theta} s(t) = a_l(\phi, \theta) s(t) \quad , \quad (2.16)$$

where $s(t)$ is the acoustic signal, $g_l(\phi)$ is the directional sensitivity (assuming that the sensitivity is equal for all θ) and $e^{-jk y_l \sin \theta}$ is the delay of $s(t)$ from the array's origin to y_l . Thereby, the amplitude part of $a_l(\phi, \theta)$ depends on $g_l(\phi)$, while the delay depends on θ . The output is thus a weighting and delay of the emitted signal $s(t)$, where the weight is a function of the angle ϕ and the delay of θ .

In the following, we will assume that $g_l(\phi)$ in (2.15) ensures that *all* sensors on the y -axis mainly collect energy from the y - z plane. Further, all elements are placed on the y -axis and have the same directional sensitivity $g_1(\phi) = \dots = g_L(\phi) = g(\phi)$, $l = 1 \dots L$. Suppose the inter element distance is d , we then have a Uniform Linear Array (ULA). For a single narrow band signal s_m , emanating from direction θ_m , we now have:

$$\mathbf{a}_{ULA}(\theta_m) = g(\phi) [1 e^{-jkd \sin \theta_m} \dots e^{-j(L-1)kd \sin \theta_m}]^T \quad (2.17)$$

The vector \mathbf{a} is called the steering vector of received signal number m , denoted s_m , where θ_m is the direction of signal s_m and k is the wavenumber. The vector \mathbf{a} defines the signal's delay from one sensor to the other over the entire array - it steers the direction of the signal with regard to the array. The exponents of \mathbf{a} express the phase difference of the signal between the different elements. It depends on the

direction of propagation and the frequency of the signal. Assume that all signals arrive, roughly, from the direction of interest ϕ , $g(\phi)$ models g in the y - z plane of interest. Further, if each sensor's directional sensitivity is independent of θ , we get:

$$\mathbf{a}_{ULA}(\theta_m) = g[1 e^{-jkd \sin \theta_m} \dots e^{-j(L-1)kd \sin \theta_m}]^T \quad (2.18)$$

Dropping the ULA subscript for convenience, the output vector $\mathbf{y}_m(t)$ of the Uniform Linear Array for a single signal $s_m(t)$ is:

$$\mathbf{y}_m(t) = \mathbf{a}(\theta_m)s_m(t) \quad (2.19)$$

Equation (2.19) states that a signal coming from the farfield has linear phase shifts in the different sensor elements. These phase shifts are exactly what (2.18) expresses. In the following, g will be dropped for simplicity.

Further, if M signals impinge on the array at one time instant, the output of the ULA will be:

$$\mathbf{y}(t) = \sum_{m=1}^M \mathbf{a}(\theta_m)s_m(t) \quad (2.20)$$

This can also be written as:

$$\mathbf{y}(t) = \mathbf{A}(\theta)\mathbf{s}(t) \quad (2.21)$$

where $\mathbf{A}(\theta) = [\mathbf{a}(\theta_1), \dots, \mathbf{a}(\theta_M)]$ is a steering matrix and $\mathbf{s}(t) = [s_1(t) \dots s_M(t)]^T$ is a vector of waveforms/signals.

At a given moment of time t , the signals sampled at the array are contaminated by noise and erroneous measurements (also a kind of noise). The samples over the array are collected in a vector $\mathbf{y}(t)$, containing both the impinging signals and the noise. The vector \mathbf{y} now holds all available information about the signals impinging on the array, and the *Directions Of Arrival (DOA)* for the impinging signals can be computed by using the phase information of the different elements in the vector.

Adding noise to equation (2.21) yields our model of the system:

$$\mathbf{y}(t) = \mathbf{A}(\theta) \mathbf{s}(t) + \mathbf{n}(t) \quad (2.22)$$

where $\mathbf{A}(\theta)$ is denoted the steering matrix and $\mathbf{n}(t)$ is additive noise present at time t .

If the array consists of sensors spaced in accordance with (2.14), the array is said to be unambiguous. This ensures that incoherent signals have unique steering vectors, the columns in $\mathbf{A}(\theta)$ are linearly independent. This fact is exploited in section 5.2, but is first discussed somewhat more detailed in chapter 4.

Chapter 3

Bathymetric Sonars

Bathymetric sonars are devices for mapping of the sea floor. The mapping is done by estimating the range from the vessel to several points on the bottom. By computing the range r , from the vessel to each of these points, together with the angle θ , between a line from each point to the vessel and the line perpendicular to the sea surface (Fig. 3.1), we get a polar coordinate positioning of each point.

In reality we do not have the exact distances, angles and points. The bathymetric sonar works by estimating the distances through the knowledge that sound propagates a given distance at a given time. The sonar sends out a pulse of sound, a *ping*, and observes the echoes. Angles are estimated for each received echo through beamforming algorithms, and angle together with distance (time) gives us the polar coordinates we seek. Our points on the sea floor are found. This is somewhat simplified. In practice, a layered sea model is used. The sound velocity in each layer is estimated, and the ray path is determined.

By following the above procedure, maps of large areas on the sea floor can be made. This is done by putting the information from many such succeeding pings together. If a vessel sends several pings while going forth (x direction), these pings will when put together make out a stripe of depth-measured area (z values) where the vessel is running. Several such stripes lying side by side will then constitute an entire bathymetric map. To get more correct bathymetric maps, it is possible to let the stripes overlap each other, or run a second survey with stripes crossing the first surveys' stripes.

Today, two kinds of bathymetric sonars are the most important: sidescanning sonars and multibeam sonars. These two sonars work in slightly different ways, the sidescanning sonar sends and listens to the side (hence the name), while the multibeam sonar sends downwards and listens in several directions. In fact, today multibeam sonars are often tilted slightly to the side, enabling them to map broader sectors than what is achieved by no tilting. The sidescanning sonar is only capable of receiving one signal at a time, whereas the multibeam sonar can receive many signals at each time instant.

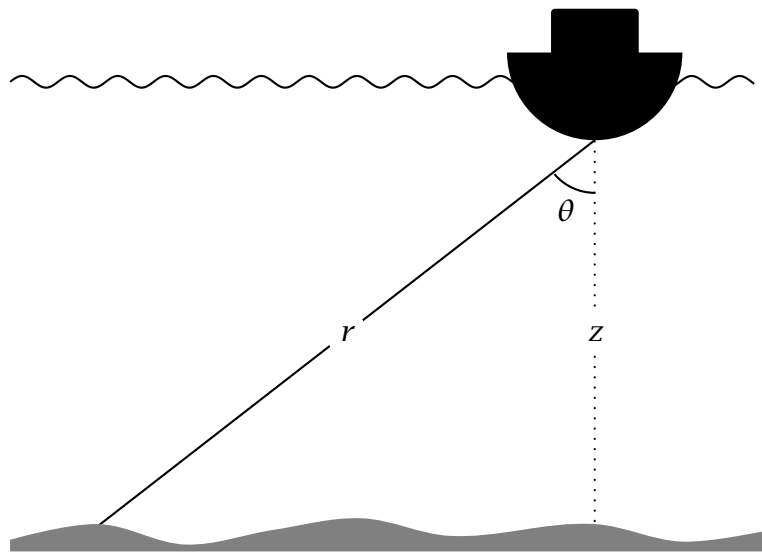


Figure 3.1: A bathymetric sonar at work. Ship viewed from behind.

3.1 The Sidescanning Sonar

A sidescanning sonar is a sonar that sends a pulse to its side, and analyses the receiving echoes (Fig. 3.2). The array configuration in the sidescanning sonar actually consists of two identical parallel arrays, mounted side by side, parallel to the moving direction. By such a geometry, the array transmits sound which is more powerful in a “plane” orthogonal to the direction of movement than in the other directions. This way it will only receive strong echoes directly from the side. The directional sensitivity, see (2.15), of the receive sensors will only strengthen this effect.

Modern sidescanning sonars estimate the direction of the echoes by interferometry [3]. Interferometry is a technique that works by finding the difference in phase of an incoming signal sampled at two or more places in space. The phase difference gives information about the direction of the signal - and hence its origin, assuming one knows the signal’s frequency and propagation velocity.

Since a typical sidescanning sonar only receives signals at two places in the beam plane it can not resolve more than one signal at a time [3, 11, 12]. This means that if more than one echo is present at a time, these signals will “counterbalance” and the result is that one does not know from which direction either signal came. Thereby, the bathymetric mapping can be poor when the bottom is rough [3, 9]. This leads to the fact that the sidescanning sonar is mainly used where the bottom to be mapped is fairly smoothly curved. Under such conditions it can achieve high resolution, depending on the pulse length, resulting in a very good mapping. Mapped sectors 7 times broader than the tow altitude can be achieved in these cases [3]. The resolution of the sidescan sonar is defined as the ability to determine a plane wave’s direction of propagation.

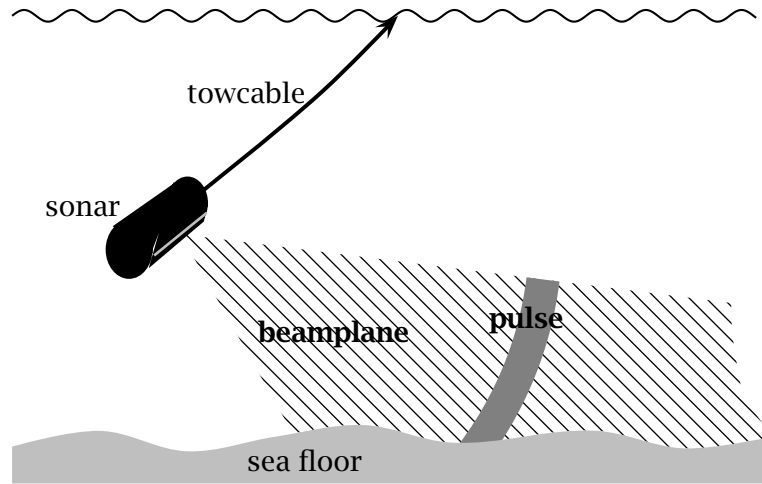


Figure 3.2: The beam plane of a sidescanning sonar.

Another problem with the sidescanning sonar is that by sending the pulse-signal to the side, part of the beam may hit the sea surface and reflect back from it. This might lead to ambiguity in the phase measurements [3]. Such is not the case when the sonar is towed at large depths.

Sidescanning sonars are also used for sea bed imaging, which again can be used for classification. This can either be done by: (1) Measuring the backscatter strength of the sea floor and its angular variations, which are known to be strongly dependent on the sea-bed type [14]. Or: (2) Analysis of the spectral features of the backscattered time signals, since the backscatter signal spectrum should reproduce the spatial sea floor spectrum and therefore be representative of the sea-bed [14]. Lurton et al. [14] discuss these methods with respect to multibeam sonars, but the methods have previously been shown to be a useful tool for classification by sidescan sonars [14].

Both methods tell us something about the bottom's consistence (sea-bed type), and combined with bathymetric maps we gain knowledge about what sort of sea-bed type we have at different places and depths.

3.2 The Multibeam Sonar

The multibeam sonar also transmits signals in a plane orthogonal to the direction of movement. It uses, as the sidescanning sonar, a transmit array mounted parallel to the moving direction. To receive the signal, the multibeam sonar uses a receive array orthogonal to the transmit array. The two arrays are mounted in a T or L configuration (Fig. 3.3).

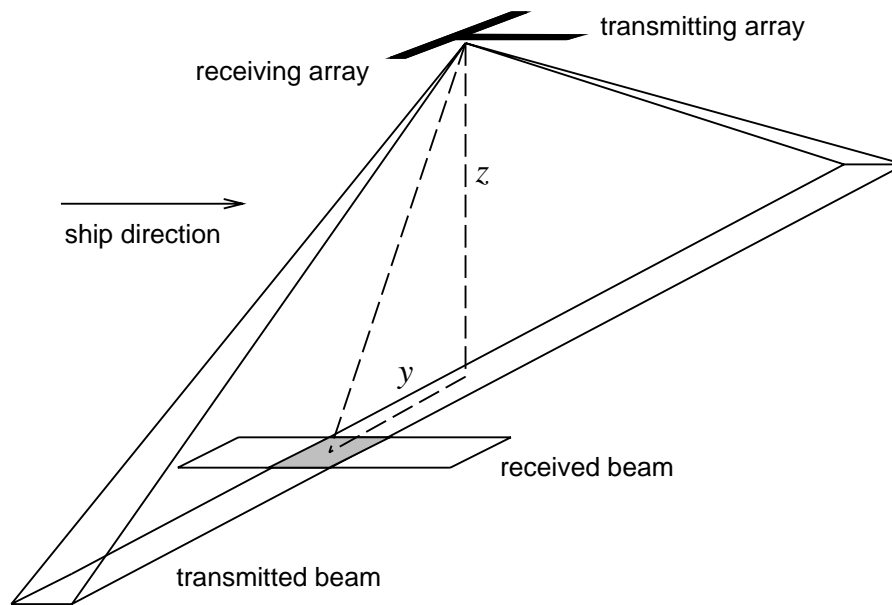


Figure 3.3: Geometry of the arrays and their beamform for a multibeam sonar.

By picking up signals in an array mounted orthogonal to the ship's direction the multibeam sonar can, as opposed to the sidescanning sonar, compute the Directions of Arrival (*DOA*) of *several* signals at the same time - one signal less than the number of sensors [11, 12]. The multibeam sonar will only be corrupted in the event where two scatters are too closely spaced to be resolved. This means that multibeam sonars are more suitable for complex bottom surfaces than the sidescanning sonar [6]. In such environments the multibeam sonar yields higher accuracy of the bottom estimates than the sidescanning sonar [3].

In addition, multibeam sonars can be used for sea-bed imaging [1, 14]. The same methods for sea-bed type identification apply for the multibeam sonars [14] as for the sidescan sonars. In fact the multibeam sonars may reduce some of the ambiguities of the observed backscattering strength in wide non-flat sea floor areas, as opposed to the classical sidescan sonars [1].

The multibeam sonar's ability to resolve closely spaced sources is what we mean by its resolution. A high resolution is essential, since one can then resolve closely spaced objects or bottom features. Since the multibeam sonar determines several bottom points broadside while moving ahead, it can map broad sectors of the bottom by just one passage [6, 19, 29].

Many different multibeam sonar systems exist: Sea Beam (series), Hydrosweep, Bottom Chart, Echos XD, EM (series), Benigraph, MBSS and SARAS [3, 6, 9, 21, 29]. Their physical sizes vary with which frequencies are used, which again vary with what the sonar is to be used for. Absorption of sound in the ocean increases proportional to the square of the acoustic frequency. Thus, absorption will condition the range capabilities of any sonar [3]. Sonars for large depths require lower frequency, otherwise

too much signal power is lost when the large amounts of water are penetrated. With intersensor spacing, d , approximately equal to half the wavelength, see (2.14), the spacing becomes large due to the low frequency. To retain a satisfactory resolution, a narrow mainlobe is needed both for the transmit and receive array. Hence, these arrays become large. Shallow water sonars can use higher frequencies, which results in smaller, more convenient, arrays.

Multibeam sonars may also be pointed to the side, to give a wider swath or mapping sector. This way, some of the advantages of sidescanning sonars are adapted, while the more robust bathymetric mapping is kept since multibeam sonars can handle multiple scatters.

Current multibeam echo sounders use a conventional method, usually implemented as the Fast Fourier transform, to form receiving beams. The beamwidth is normally 1.5 to 5 degrees athwart ship. This spatial resolution is unsatisfactory, especially for beams of oblique incidence angles. Whereas the aperture of the array is of size Ld for normal incidence signals, its effective aperture reduces to $Ld \cos \theta$ for oblique angles θ . Thus, since the resolution (beamwidth) of the conventional method is defined through the reciprocal of the aperture, the resolution degrades as θ increases. For oblique beams, the cross-track length of the receive beam footprints are large, and bottom echo detection based on the beam amplitude values is inaccurate. The bottom detection accuracy and the mapping resolution can be improved by using subarray interferometry [32], since for an oblique beam, the pulse length is much shorter than the cross-track length of the footprint. However, the method assumes a single scatterer within each beam footprint. It works well when the bottom curve is smooth within a footprint, but fails for complex bottom curves with multiple scatters.

Chapter 4

Estimation of Covariances

The key to understanding most, if not all, beamforming algorithms is the spatial covariance matrix \mathbf{R} . Before introducing this quantity, we need some idea of how the sampling in the array is done. Beamforming algorithms are based upon the assumption that the propagation characteristics of the wave field do not change with time, at least not over the time the beamforming is done.

4.1 The Covariance Matrix

In Chap. 2, the output of the array was outlined. The assumption was that a given signal had a planar wave front and a narrow band frequency. If each sensor, y_l , samples the signal for a period of time T , the output of the short-time Fourier transform for a sensor will be

$$Y_l(t, \omega)e^{j\omega t} = \int_0^T \tilde{w}(\tau)y_l(t + \tau)e^{-j\omega\tau} d\tau \quad (4.1)$$

This is an approximation of sensor y_l 's spectrum output at time t and frequency ω , and is often called a snapshot. The connection to photography is obvious, except from \tilde{w} which is a window that may change with time. Actually, a camera also has a window that changes with time - think of the aperture. This is not suddenly open, the mechanical device uses some time to open and close - and can be viewed as a window in time.

As for cameras, the wave field should not change during the sampling period, or else the short-time Fourier transform becomes a "blurred" approximation.

We want to cancel the phase shifts in (2.19), due to the propagation direction across the array. This equals delay and sum in the time domain. Thus,

$$\mathbf{a}_{ULA}(\theta_m) = [1 \ e^{-jkd \sin \theta_m} \ \dots \ e^{-j(L-1)kd \sin \theta_m}]^T$$

models the signal's propagation characteristics in the frequency domain. Multiplying $Y_l(t, \omega)$ with $e^{jkd l \sin \theta_m}$ cancels the phase shift in $a_l(\theta_m) = e^{-jkd l \sin \theta_m}$. If this is done for each sensor, the phase of the signal is ideally equal at all sensors.

The Fourier transform output of the beamformer is now

$$\begin{aligned} Z(t, \omega) e^{-j\omega t} &= \sum_{l=0}^{L-1} \tilde{w}_l Y_l(t, \omega) e^{jkd l \sin \theta_m} \\ &= \mathbf{a}^H(\theta) \mathbf{Y}(t, \omega) \end{aligned} \quad (4.2)$$

where \mathbf{a}^H is the transposed and conjugated of \mathbf{a} . This in turn, will lead to the signal $s_m(t)$ in the array being added (or stacked) constructively. The power of the beamformer for a given signal direction θ is then

$$P(\theta) = \int_{-\infty}^{\infty} \mathbf{a}^H(\theta) \mathbf{Y} \mathbf{Y}^H \mathbf{a}(\theta) d\omega \quad (4.3)$$

For narrow band signals the integrand is skipped. We are now ready to define the covariance matrix

$$\mathbf{R} = \mathbf{Y} \mathbf{Y}^H \quad , \text{ ideally } \mathcal{E} \{ \mathbf{Y} \mathbf{Y}^H \} \quad (4.4)$$

This matrix is conjugate symmetric, it equals its own conjugate transpose, and is known as a Hermitian matrix. In addition, it would have a Toeplitz structure if it was truly the expectation of $\mathbf{Y} \mathbf{Y}^H$. That is, all elements along any diagonal would equal each other [7].

For notational convenience, $\mathbf{Y}(t, \omega) = \mathbf{y}(t)$ in the following. \mathbf{Y} , or \mathbf{y} , is the raw data available for this thesis. Since ω is a narrow band frequency, it is skipped.

The covariance matrix consists of two parts, one part constitutes the signal covariance matrix, while the other is the noise covariance matrix. By using (2.22), this can be written

$$\begin{aligned} \mathbf{R} &= \mathcal{E} \{ \mathbf{y}(t) \mathbf{y}^H(t) \} \\ &= \mathbf{A} \mathcal{E} \{ \mathbf{s}(t) \mathbf{s}^H(t) \} \mathbf{A}^H + 2\mathbf{A} \mathcal{E} \{ \mathbf{s}(t) \} \mathcal{E} \{ \mathbf{n}^H(t) \} + \mathcal{E} \{ \mathbf{n}(t) \mathbf{n}^H(t) \} \\ &= \mathbf{A} \mathbf{P} \mathbf{A}^H + \sigma^2 \mathbf{I} \quad \text{since } \mathcal{E} \{ \mathbf{n} \} = 0 \\ &= \mathbf{V}_s \mathbf{\Lambda}_s \mathbf{V}_s^H + \sigma^2 \mathbf{V}_n \mathbf{V}_n^H \quad , \end{aligned} \quad (4.5)$$

where it is assumed that the noise is additive, spatial white and Gaussian with zero mean and variance σ^2 . This means that in the ideal situation, we can find a source covariance matrix \mathbf{P} and its matrix of eigenvectors \mathbf{V}_s , with the corresponding eigenvalues $\mathbf{\Lambda}_s$. This may seem somewhat magical, but will be explained a bit later. Unfortunately, since the eigenvectors of Hermitian matrices are orthogonal, these eigenvectors do not equal the steering vectors in \mathbf{A} (unless only one signal is present). Signals seldom arrive from "orthogonal directions".

All is not lost, however, \mathbf{V}_s forms a basis for the steering vectors, with rank equal to the number of signals M . We can relate \mathbf{V}_s and \mathbf{A} by

$$\mathbf{V}_s \mathbf{T} = \mathbf{A} \quad (4.6)$$

where \mathbf{T} is an invertible square matrix. Since the steering vectors are linearly independent, this basis can only represent the steering vectors [7]. Each steering vector can be expressed as a linear combination of these eigenvectors.

From linear algebra, a matrix $\mathbf{C} = \mathbf{A}\mathbf{A}^H$ is positive semidefinite. In the case of no noise, this means that the eigenvalues in $\mathbf{\Lambda}_s$ are greater or equal to 0. In the white-noise case, the noise portion contributes σ^2 to the eigenvalues. Thus, the eigenvalues belonging to \mathbf{V}_s must exceed or equal σ^2 .

Two cases occur: (1) In the case of incoherent or partial coherent signals, we can find the signal+noise eigenvectors by picking those that have eigenvalues larger than σ^2 [7]. Thus, \mathbf{R} can be split into its final parts as was done in (4.5). (2) When signals are *perfectly* coherent, one or more of the signal eigenvalues equal σ^2 [7].

For instance, when two signals are perfectly coherent, the signal term in (4.5) resemble a single outer product – not composed of an ideal steering vector. We might actually have an eigenvector that equals the difference between the two signals, and since steering vectors are linearly independent, this linear combination of signals can not be a signal. The largest eigenvector is related to both steering vectors, but neither steering vector can be obtained from it without knowing the other – \mathbf{T} does not exist [7].

Remember that, ideally, \mathbf{V}_s and thus $\mathbf{A}\mathbf{P}\mathbf{A}^H$, had rank equal to the number of signals M . This means that $\mathbf{A}\mathbf{P}\mathbf{A}^H$ also has a null space $\mathbf{V}_n^H \mathbf{a} = 0$ of dimension $L-M$, where \mathbf{V}_n are the noise-only eigenvectors with eigenvalues σ^2 . It is this null space that is exploited in the high-resolution methods in section 5.2.

The covariance matrix \mathbf{R} is of size $L \times L$, and since one of the eigenvectors must belong to the noise subspace – at most $L-1$ eigenvectors can be used to span the signal+noise¹ subspace \mathbf{V}_s . This means that an array of L sensors can not distinguish more than $L-1$ signals. From now on, we will talk of \mathbf{V}_s as the *signal subspace*.

If we wish to weight the sensors, either to suppress the data recorded at the ends of the array – or simply to reduce the sidelobes, this can be done by multiplying \mathbf{R} with a diagonal matrix \mathbf{W} . Each diagonal element, w_l , of \mathbf{W} will then weight its corresponding y_l , since $\mathbf{W}\mathbf{y}\mathbf{y}^H \mathbf{W}^H = \mathbf{W}\mathbf{R}\mathbf{W}^H$.

4.2 Noise and Artifacts

The array of a multibeam bathymetric sonar will always have thermal noise in its electronics, an artifact. A noise component will also be typical background radiation, consisting of random waves propagating toward the array from all directions with equal probability. This constitutes an isotropic noise field. Given a large amount of such noise sources, superposition of them will according to the Central Limit Theorem be roughly Gaussian.

¹the noise portion of this subspace contributes σ^2 to the eigenvalues, the corresponding eigenvectors are not affected in the white-noise situation

If the isotropic noise is temporally filtered to be monochromatic with temporal frequency ω_0 , the correlation function has the form of a spatial sinc function [7, Sec. 2.6]. Such is the case for the EM3000, and spatial sampling at locations separated by $\lambda/2$ will then yield uncorrelated noise components.

Yet another problem than noise are coherent signals, see discussion following (4.6). To achieve better estimates when coherent signals are present, we have to apply some operations on \mathbf{R} . These operations will be explained in the next few sections.

4.3 Time Averaging

Ideally $\mathbf{R} = \mathcal{E} \{\mathbf{y}\mathbf{y}^H\}$, and the signal covariance matrix has full rank. The expectation can only be found when the signals are stationary for an infinite amount of time. Needless to say, in a bathymetric sonar scenario, the signals are not stationary for an infinite amount of time. In practice \mathbf{R} is estimated as

$$\hat{\mathbf{R}}_f^N = \frac{1}{N} \sum_{t=1}^N \mathbf{y}(t)\mathbf{y}^H(t) \quad (4.7)$$

an average over time. This is the same procedure that Rao and Hari call the Forward-only approach in [24], hence the subscript. The size of N depends upon how long time one can assume the signal to be stationary. In narrow band multibeam sonars, the sampling frequency is determined by the bandwidth, a signal pulse is often sampled only once or twice by the L elements. The EM3000 has pulse length $150 \mu\text{s}$, and samples at 14293 Hz , thereby the pulse is sampled 2.1 times. The received signal is therefore not assumed stationary in time, and N is assumed to equal 1. This strategy is also used by Pantzartzis [18], where a multibeam sonar with pulse length 0.2ms and sample frequency 10kHz is simulated. Since $N = 1$ for our sonar model, $\hat{\mathbf{R}}$ is greatly affected by noise. Still, it is the best we can do unless more stationary samples are made available by more extensive oversampling in time. In other situations, signals are stationary for several snapshots, eg [11], and large amount of data are available for estimating $\hat{\mathbf{R}}$. In these cases, a smaller variance of the estimated quantities is achieved by time averaging. Still, time averaging will not help when it comes to coherent signals, other methods must be used for this. Stationary signals that are truly coherent will stay coherent, no matter how long the stationary period is - even though the variance gets smaller by time averaging.

The notation $\hat{\mathbf{R}}_f$ will be used throughout the thesis, while N is dropped since it equals 1 in the following. We keep $\hat{\mathbf{R}}_f$ so that the notation is consistent with that from [24].

4.4 Spatial Smoothing

Spatial smoothing with subarrays, also known as Forward-only Smoothing (FS approach) [24], may help in difficult scenarios where coherent signals are present.

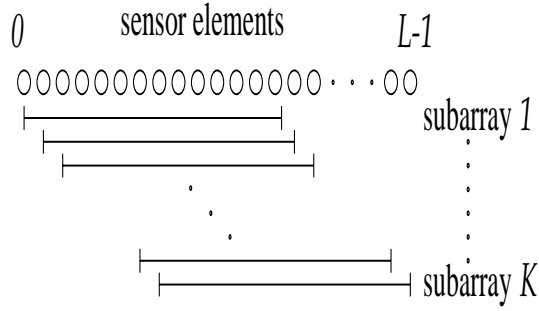


Figure 4.1: Array split into several subarrays. The subarrays are then summarized to obtain a spatial smoothing.

Coherent signals impose difficulties on the beamforming, since the rank of \mathbf{V}_s (discussed in Sec. 4.1) becomes deficient for such signals. The idea is to split the array in several equal and overlapping subarrays (Fig. 4.1), which are assumed to have identical steering vectors, except for their phase factor [11]. The subarrays are of size L_K , where $L_K = L - K + 1$ and K is the number of subarrays. The resulting covariance matrix is

$$\hat{\mathbf{R}}_{fs}^K = \frac{1}{K} \sum_{k=1}^K \hat{\mathbf{R}}_k^f \quad (4.8)$$

where each $\hat{\mathbf{R}}_k^f$ is of size $L_K \times L_K$ and is part of $\hat{\mathbf{R}}_f$, centered along its diagonal. Here, the superscript K in $\hat{\mathbf{R}}_{fs}^K$ tells us how many subarrays are used. From now on, L will be used for both L and L_K .

“The spatial smoothing induces a random phase modulation which in turn tends to decorrelate the signals that caused the rank deficiency”, c.f. [11]. This citation will become more clear after having read the next section. In short, the M signals are simulated as impinging on the array at K different times, but from the same angles. Actually, the phases are not random – they depend on the directions of the amount of subarrays and the intersensor spacing. In the next section, there is also a phase modulation – except, only one new phase is introduced for each of the signals.

Regarding spatial smoothing, some theory for optimal number of subarrays for different methods is developed by Rao and Hari in [23, 24]. This will be deferred until section 5.2, where the high-resolution methods are presented.

4.5 Forward Backward Smoothing

Forward-backward smoothing (FBS approach) [23, 24] works by left and right multiplying $\hat{\mathbf{R}}_{fs}^*$ with the matrix \mathbf{J} , where \mathbf{J} is an exchange matrix consisting of an identity

matrix of size $L \times L$ flipped left/right. The new estimate is

$$\hat{\mathbf{R}}_{fbs}^K = \frac{\hat{\mathbf{R}}_{fs}^K + \hat{\mathbf{R}}_{bs}^K}{2}, \quad \text{where } \hat{\mathbf{R}}_{bs}^K = \mathbf{J}\hat{\mathbf{R}}_{fs}^*\mathbf{J} \quad (4.9)$$

$\hat{\mathbf{R}}_{fs}^*$ is actually the conjugate of $\hat{\mathbf{R}}_{fs}^K$, but K had to be skipped in order to make room for the conjugate symbol $*$. For $K = 1$ this approach is known as forward/backward averaging. The $K = 1$ strategy only works for two coherent signals, but will then restore the rank of the signal covariance matrix [11].

The effect of multiplying $\hat{\mathbf{R}}_{fs}^*$ with \mathbf{J} is more easily understood if we only look at its effect on $\mathbf{a}^*(\theta_m)$. We have

$$\mathbf{J}\mathbf{a}^*(\theta_m) = e^{j(L-1)kd \sin \theta_m} \mathbf{a}(\theta_m) \quad (4.10)$$

which means that a signal coming from direction θ_m is simulated as impinging twice (Fig. 4.2), but at slightly different times and with only half the power (we divide (4.9) by 2). Say we have two coherent signals, impinging from direction θ_1 and θ_2 . Generally, $\mathbf{a}(\theta_1)$ will not be coherent with $\mathbf{J}\mathbf{a}^*(\theta_2)$ and $\mathbf{a}(\theta_2)$ not with $\mathbf{J}\mathbf{a}^*(\theta_1)$. The coherent signals are now “decorrelated”, and since there are now four signals impinging on the array, instead of two, no power is lost.

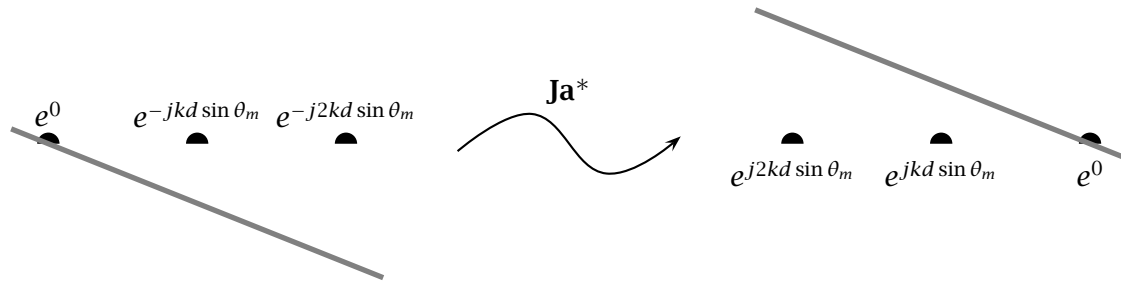


Figure 4.2: Visualizing the effect on the sensor delays when multiplying \mathbf{a}^* with the exchange matrix \mathbf{J} .

In [24, Theorem 3], it is found that “For a given number of subarrays K , $\hat{\mathbf{R}}_{fbs}^K$ is better conditioned than $\hat{\mathbf{R}}_{fs}^K$... The smaller condition number along with a larger λ_M , which plays an important role in low-SNR and high-resolution cases, indicates that the FBS approach is preferable to the FS approach.” Further, [24, Property 4] states that for MUSIC (one of the high-resolution methods): “For the case of uncorrelated sources, the FS and FBS estimates have the same properties.”

4.6 Noise Subtraction

An additional strategy for improving $\hat{\mathbf{R}}$ is used. The idea is to estimate the structured noise present in the array. This noise is estimated from the samples taken

before the first bottom echo returns, and is later subtracted from $\hat{\mathbf{R}}$ at each snapshot. This can be done if the noise is additive. If no structured noise is present, the effect of this procedure will be none. But, if the array in some way produces a systematically incorrect output, this output will be corrected. It can be viewed as calibrating the array. In any case, subtracting this noise should ideally get rid of the $2\mathbf{A}\mathcal{E}\{\mathbf{s}(t)\}\mathcal{E}\{\mathbf{n}^H(t)\}$ term in (4.5) if $\mathcal{E}\{\mathbf{n}(t)\} \neq 0$.

The structured noise is estimated as

$$\hat{\mathbf{N}} = \frac{1}{N_n} \sum_{t=1}^{N_n} \mathbf{y}(t)\mathbf{y}^H(t) \quad , \quad (4.11)$$

where N_n is the number of snapshots used to estimate the noise. When applied, this noise removal is done before the operations given by (4.8) and (4.9).

Note that since finding the eigenvectors of $\hat{\mathbf{R}}$ is an iterative procedure one might expect the procedure to converge faster when having removed the noise covariance.

Chapter 5

Beamforming

Beamforming can be used to (1) Focus an array at a particular point in space, as in medical ultrasound. (2) Steer an array to a certain direction (or angle), and thereby pick up signals arriving mostly from that direction (radar, sonar). (3) Search for propagation directions - as we do in bathymetric sonars. In addition, we get an enhancement of the Signal to Noise Ratio (*SNR*) [3, 7, 17]. Noise is randomly distributed, while the signals have a systematic phase shift from sensor to sensor. If one cancels these phase shifts, the signals can be added constructively, while the noise will stay at approximately the same level.

The propagation direction of a signal, or other signal parameters, have to be derived from the spatial samples, of the reflected signals, collected by the array. Such derivations are often called beamforming.

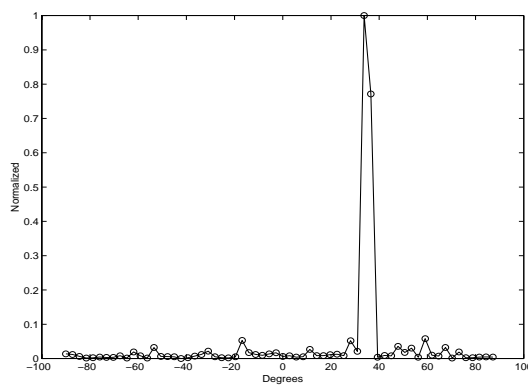
Originally, beamforming meant forming the transmit beam for some purpose. Today, we use the word beamforming both for the transmit - and the receive beam. The beam is simply the mainlobe of the array's radiation - or response pattern. Changing this pattern, either by weighting the array elements or by steering its direction, is what beamforming is all about [7, 8, 28].

As in most signal processing applications, there is a tradeoff between different needs. The tradeoff in bathymetric sonars is between cross-track coverage, accuracy and resolution [19]. The goal of this thesis, is to improve the resolution of a bathymetric sonar, working only with the output of the array - after the tradeoff is done. To achieve this, the Fourier transform method for beamforming and some high-resolution, also called adaptive, methods have been studied.

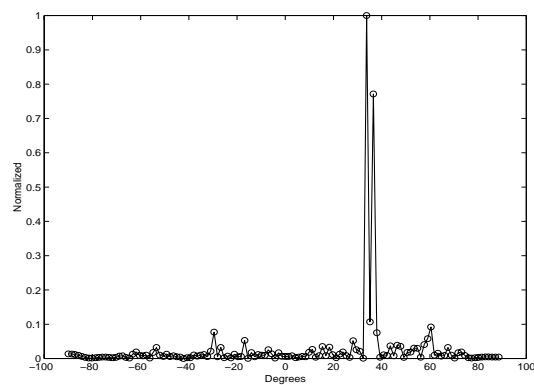
The beamforming methods can be divided into different groups, (1) conventional methods and (2) adaptive (or high-resolution) methods. The high-resolution methods can be further split (Tab. 5.1), into spectral based and parametric methods. These methods, again, are often referred to as eigenanalysis methods [7, 22] (or subspace methods [11, 22]), while in more recent papers, Rao [23] denote some subspace methods as state space methods. In the following I will only talk of conventional or high-resolution methods, just mentioning if they belong to the spectral or paramet-

Type	Conventional	High Resolution
Spectral Based	Sum and Delay	Minimum Variance
	Fourier Transform	MUSIC Eigen Vector Minimum Norm
Parametric		root-MUSIC ESPRIT

Table 5.1: Beamforming methods



(a) Spatial spectrum when using an angular increment of 2.8 degrees.



(b) Spatial spectrum when using an angular increment of 1.4 degrees.

Figure 5.1: Effect of angular increments when trying to resolve two closely spaced sources. Waves propagating from DOA 34 and 36 degrees.

ric methods.

Spectral based methods are those that need to scan all possible directions of impinging signals. For each set of samples taken at time t , one computes a spectrum over all directions. The largest peaks in the spectrum are taken to be the $DOAs$. These methods can not yield a better resolution than the size of the angular increments we use for computing the spectrum (Fig. 5.1), which means that higher resolution needs more computing. In addition, they need post processing of the spectrum. The post processing is crucial for obtaining correct bottom estimates.

Parametric methods require a simultaneous search for all parameters of interest (here the $DOAs$), at the expense of an increased computational complexity [11]. The parametric methods have the potential of resolving closely spaced sources where spectral based methods fail [22]. “Spectral methods always have less resolution compared to Root forms” c.f. [22].

5.1 Conventional Beamformer

Two important issues in developing multibeam systems are the mapping resolution and the bottom detection accuracy. Current multibeam echo sounders use a conventional method, usually the Fast Fourier transform, to form receiving beams.

The conventional method is based on stacking (or summing) the elements in the array. For a signal of normal incidence, this will result in a improvement of the *SNR*. Ideally, it improves with a multiplicity equal to the amount of sensors L . The reason is that the signal will be equal in all sensors, and adds constructively, while the noise is different from element to element, and will therefore not add up in the same way. If the signal comes from another direction, we can simply delay the output of the elements in accordance with the signal's direction before we sum the elements. Delaying the output of the elements in the array is exactly what was done in (4.2) and (4.3).

Now, let us do this more formally. We want to maximize the power output of the array, in all directions, by multiplying the signal vector $\mathbf{y}(t)$ by a weight vector \mathbf{w} . The signal model (2.22) was stated as

$$\mathbf{y}(t) = \mathbf{A}(\theta) \mathbf{s}(t) + \mathbf{n}(t)$$

To keep things simple, assume that only one signal is present - such that $\mathbf{A}(\theta) = \mathbf{a}(\theta)$. Maximizing the power is done by

$$\begin{aligned} \max_{\mathbf{w}} \mathcal{E} \left\{ \mathbf{w}^H \mathbf{y}(t) \mathbf{y}^H(t) \mathbf{w} \right\} &= \max_{\mathbf{w}} \mathbf{w}^H \mathcal{E} \left\{ \mathbf{y}(t) \mathbf{y}^H(t) \right\} \mathbf{w} \\ &= \max_{\mathbf{w}} \left\{ \mathcal{E} |s(t)|^2 |\mathbf{w}^H \mathbf{a}(\theta)|^2 + \sigma^2 |\mathbf{w}|^2 \right\} \end{aligned} \quad (5.1)$$

if we assume the noise is spatially white. A non-trivial solution with norm equal to 1 is then

$$\mathbf{w}_{FT} = \frac{\mathbf{a}(\theta)}{\sqrt{\mathbf{a}^H(\theta) \mathbf{a}(\theta)}} \quad (5.2)$$

Putting (5.2) back into (5.1) yields the spatial spectrum

$$\begin{aligned} P_{FT}(\theta) &= \mathbf{w}_{FT}^H \mathcal{E} \left\{ \mathbf{y}(t) \mathbf{y}^H(t) \right\} \mathbf{w}_{FT} \\ &\approx \mathbf{w}^H \hat{\mathbf{R}} \mathbf{w} \\ &= \frac{\mathbf{a}(\theta)}{\sqrt{\mathbf{a}^H(\theta) \mathbf{a}(\theta)}} \hat{\mathbf{R}} \frac{\mathbf{a}(\theta)}{\sqrt{\mathbf{a}^H(\theta) \mathbf{a}(\theta)}} \\ &= \frac{\mathbf{a}^H(\theta) \hat{\mathbf{R}} \mathbf{a}(\theta)}{\mathbf{a}^H(\theta) \mathbf{a}(\theta)} \end{aligned} \quad (5.3)$$

of the conventional beamformer. The signal's direction is found by scanning (5.3) for all possible θ 's, the one (or ones) giving the highest power is the *DOA*.

The conventional beamformer suffers from rather limited resolution capabilities. Its resolution is bounded by the size of the spatial sampling array, and is defined to be the reciprocal of the main lobe width. As evident from the formula, this spectrum has to be scanned for all possible θ values.

Much could be said about the conventional beamformer, but this is not the goal of this thesis. To learn more about use of windows, sidelobe level, mainlobe width, Fourier transform and Fast Fourier transform see any good textbook, such as [5, 7, 8].

Realizing that the traditional or conventional Fourier transform method for beamforming is inadequate in resolving closely spaced synchronous echoes [19], one has to seek for other, better, methods. The spatial resolution is unsatisfactory, especially for beams of oblique incidence angles. For oblique beams, the cross-track length of the receive beam footprints is large, and bottom echo detection based on the beam amplitude values is inaccurate. Thus, it is interesting to examine how modern high-resolution beamformers work for multibeam echo sounders. A high resolution is essential, since one can then resolve closely spaced objects or bottom features. The ability of resolving closely spaced objects is of essence, since one can then get a more thorough and correct mapping of the sea floor. Promising methods have been proposed, and based on early research [4, 6, 10, 11, 12, 13, 18, 19, 22, 23, 24, 26, 27, 28, 29, 31] some methods were selected for further study.

5.2 High Resolution Beamformers

Several high-resolution beamformers exist, a few of them are presented here.

Minimum-Variance serves as a gateway to the modern high-resolution beamformers; some of the subspace based high-resolution beamformers can be said to have evolved from Minimum-Variance. These include Eigen-Vector, MUSIC, Minimum-Norm and to some extent root-MUSIC. ESPRIT is also based on the subspace approach, although it uses another subspace and is based on other parts of linear algebra.

Since the conventional beamformer has limited resolution capabilities, several new beamforming methods have been developed. One of the most easily understandable is the Minimum-Variance or Capon's beamformer.

Minimum-Variance is the result of a constrained optimization problem. Such problems often apply a weight vector \mathbf{w} , on the observations \mathbf{y} , to minimize the mean-squared value of $\mathcal{E}\{|\mathbf{w}^H \mathbf{y}(t)|^2\}$ subject to a constraint $\mathbf{C}\mathbf{w} = \mathbf{c}$.

Minimum-Variance [7, 11] The optimization problem of Minimum-Variance is to minimize the power of noise and signals coming from other directions than θ , the current direction of look. This can be presented as

$$\min_{\mathbf{w}} \mathcal{E}\{|\mathbf{w}^H \mathbf{y}(t)|^2\} = \min_{\mathbf{w}} \mathbf{w}^H \hat{\mathbf{R}} \mathbf{w} \quad \text{subject to} \quad \mathbf{a}^H(\theta) \mathbf{w} = 1 \quad , \quad (5.4)$$

and the constraint says that we want to keep a fixed gain in the θ direction. By using Lagrange multipliers

$$\mathcal{L}(\mathbf{w}, \lambda) = \mathbf{w}^H \hat{\mathbf{R}} \mathbf{w} + \lambda (\mathbf{a}^H(\theta) \mathbf{w} - 1) \quad (5.5)$$

we now try to minimize the new problem posed by (5.5)

$$\nabla_{\mathbf{w}} \mathcal{L}(\mathbf{w}, \lambda) = 2\hat{\mathbf{R}}\mathbf{w} + \lambda\mathbf{a}(\theta) \quad (5.6)$$

The minimum of (5.5) is found when its gradient (5.6) equals 0. This yields

$$\mathbf{w} = \frac{-\lambda \hat{\mathbf{R}}^{-1} \mathbf{a}(\theta)}{2} \quad (5.7)$$

Remember the constraint $\mathbf{a}^H(\theta) \mathbf{w} = 1$. Thus, by replacing \mathbf{w} in the constraint with the expression from (5.7) we have

$$\frac{-\lambda \mathbf{a}^H(\theta) \hat{\mathbf{R}}^{-1} \mathbf{a}(\theta)}{2} = 1 \quad (5.8)$$

and consequently

$$\lambda = \frac{-2}{\mathbf{a}^H(\theta) \hat{\mathbf{R}}^{-1} \mathbf{a}(\theta)} \quad (5.9)$$

This, in combination with (5.7), yields in turn the weight vector of Minimum-Variance

$$\mathbf{w}_{MV} = \frac{\hat{\mathbf{R}}^{-1} \mathbf{a}(\theta)}{\mathbf{a}^H(\theta) \hat{\mathbf{R}}^{-1} \mathbf{a}(\theta)} \quad (5.10)$$

Finally, an expression for the Minimum-Variance beamformer is found

$$\begin{aligned} P_{MV}(\theta) &= \mathbf{w}_{MV}^H \hat{\mathbf{R}} \mathbf{w}_{MV} \\ &= \frac{1}{\mathbf{a}^H(\theta) \hat{\mathbf{R}}^{-1} \mathbf{a}(\theta)} \end{aligned} \quad (5.11)$$

Thus, Minimum-Variance does not maximize the power, as the conventional beamformer did, but rather tries to minimize it in all directions except for the one we look in. This is done by nulling out the power in other directions than the one we are interested in, keeping a fixed gain in the direction of look.

Since we here make use of the inverse of the correlation matrix $\hat{\mathbf{R}}$, this method is obviously more time consuming than the conventional beamformer. More worrying though, is the fact that $\hat{\mathbf{R}}^{-1}$ may not even exist. When few samples are used, we risk that $\hat{\mathbf{R}}$ is singular. Here the number of snapshots is lower than the number of sensors, $N < L$, and $\hat{\mathbf{R}}$ is generally not invertible [11]. This means that we have to use the pseudo-inverse, $\hat{\mathbf{R}}^\dagger$, instead of the real inverse of $\hat{\mathbf{R}}$ in this algorithm. This was also done in [11]. It is difficult to measure the degradation in performance due to the applied pseudo-inverse, since there is no real inverse to compare to.

Eigen Analysis Methods If one rewrites (5.11), further improvements can be made. A hermitian matrix \mathbf{A} of size $n \times n$ can be rewritten as $\mathbf{A} = \sum_{i=1}^n \lambda_i \mathbf{v}_i \mathbf{v}_i^H$, and its inverse is described by $\mathbf{A}^{-1} = \sum_{i=1}^n \frac{1}{\lambda_i} \mathbf{v}_i \mathbf{v}_i^H$. If there were infinitely many samples available for a stationary signal, the covariance matrix would be both Hermitian and Toeplitz. In our scenario fewer samples are available, $N < L$, thus $\hat{\mathbf{R}}$ may not have a Toeplitz structure. Still, $\hat{\mathbf{R}}$ is hermitian. Thus, (5.11) can be written

$$\begin{aligned}
P(\theta) &= \frac{1}{\mathbf{a}^H(\theta) \hat{\mathbf{R}}^{-1} \mathbf{a}(\theta)} \\
&= \frac{1}{\mathbf{a}^H(\theta) \left(\sum_{i=1}^n \hat{\lambda}_i^{-1} \hat{\mathbf{v}}_i \hat{\mathbf{v}}_i^H \right) \mathbf{a}(\theta)} \\
&= \frac{1}{\sum_{i=1}^n \hat{\lambda}_i^{-1} |\mathbf{a}^H(\theta) \hat{\mathbf{v}}_i|^2} \\
&= \frac{1}{\sum_{i=1}^M \hat{\lambda}_i^{-1} |\mathbf{a}^H(\theta) \hat{\mathbf{v}}_i|^2 + \sum_{i=M+1}^L \hat{\lambda}_i^{-1} |\mathbf{a}^H(\theta) \hat{\mathbf{v}}_i|^2} ,
\end{aligned} \tag{5.12}$$

where, by convention, the eigenvalues are in descending order [7, Sec. 7.3]. Investigating the denominator in this last expression, we see that the sum that is contributing the most to the spectrum in the signal directions is in fact the last one. This last sum is the expression for the noise subspace. The reason for this being the main contributor is that when we look in a signal direction, the expression $|\mathbf{a}^H(\theta) \hat{\mathbf{v}}_i|^2$ for $i > M$ is small, while for $i \leq M$ it is large. Hence, the spectrum (5.12) takes on larger values if we skip the signal subspace (the first sum in the denominator).

Now then, what we have is a new spectrum denoted by:

$$P(\theta) = \frac{1}{\sum_{i=M+1}^L \hat{\lambda}_i^{-1} |\mathbf{a}^H(\theta) \hat{\mathbf{v}}_i|^2}$$

Equivalently, by replacing the vector multiplications with matrix forms, and letting $\hat{\mathbf{\Lambda}}_n$ express a diagonal matrix containing $\hat{\lambda}_{M+1}$ to $\hat{\lambda}_L$, we obtain the notation of Krim and Viberg [11]:

$$P(\theta) = \frac{1}{\mathbf{a}^H(\theta) \hat{\mathbf{V}}_n \hat{\mathbf{\Lambda}}_n^{-1} \hat{\mathbf{V}}_n^H \mathbf{a}(\theta)} \tag{5.13}$$

Equation (5.13) gives us both the MUSIC and the Eigen-Vector spectrum, outlined below.

Eigen Vector The spectrum of the Eigen Vector method [7] is simply the expression (5.13),

$$P_{EV}(\theta) = \frac{1}{\mathbf{a}^H(\theta) \hat{\mathbf{V}}_n \hat{\mathbf{\Lambda}}_n^{-1} \hat{\mathbf{V}}_n^H \mathbf{a}(\theta)} , \tag{5.14}$$

and therefore makes use only of the eigenvalues, $\hat{\mathbf{\Lambda}}_n$, belonging to the noisy part of the correlation matrix $\hat{\mathbf{R}}$. The need to compute the eigenvalues and vectors slows the algorithm even further down than Minimum-Variance. Further, $\hat{\mathbf{R}}$ may be singular

since $N < L$. Consequently, some λ 's may equal 0. Therefore, the pseudo-inverse once more has to be used.

When the number of signals M is chosen too low, Eigen-Vector will degrade towards the Minimum-Variance algorithm. This is easily seen from (5.12) by letting M equal 0.

MUSIC [11, 18, 22, 24, 27] also relies on computing the eigenvalues and eigenvectors, as in the Eigen Vector method. But, in this algorithm, we avoid the problem of inverting the matrix $\hat{\Lambda}_n$. In the MUSIC algorithm, the eigenvalues are taken to be of size 1. The fact that the noise eigenvalues are now set to 1, can be viewed as whitening the noise subspace [7, p 384]. Thus $\hat{\Lambda}_n = \mathbf{I}$, hence - we only need the eigenvectors.

$$P_{MU}(\theta) = \frac{1}{\mathbf{a}^H(\theta)\hat{\mathbf{V}}_n\hat{\mathbf{V}}_n^H\mathbf{a}(\theta)} \quad (5.15)$$

Albeit we no longer need to invert $\hat{\Lambda}_n$, we still have to compute the eigenvectors.

The above mentioned whitening of the noise subspace is all well if M is chosen correctly. What if M is erroneously set to some number lower than its real value? MUSIC will then whiten some of the eigenvectors corresponding to signals (5.12), with the effect that a degraded spectrum is computed.

Krim and Viberg [11] write that for coherent signals a signal eigenvector diverges into the noise subspace. Krim uses \mathbf{u} for the eigenvectors, and denotes the noise subspace as \mathbf{U}_n . *“Therefore, in general $\mathbf{U}_n^H\mathbf{a}(\theta) \neq 0$ for any θ and the MUSIC ‘spectrum’ may fail to produce peaks at the DOA locations. In particular, the ability to resolve closely spaced sources is dramatically reduced for highly correlated signals”* c.f. [11].

Other problems with MUSIC are discussed in [26], where Roy states that: *“in low SNR scenarios and in situations where even small sensor array errors are present, the ability of the conventional MUSIC spectrum to resolve closely spaced sources (i.e., observe multiple peaks in the measure) is severely degraded. Nevertheless, it should be emphasized that in spite of these drawbacks, MUSIC has been shown to outperform previous techniques.”*

In [19] Pantzartzis et al. claim that: *“if narrow peaks in the power spectrum means a potential for high resolution, MUSIC has a even higher potential than the parametric methods, but depends on a good estimate of the number of incoming signals M at time t ”.*

Pantzartzis et al. [19] also find that for 100 simulated pings, the standard deviation of MUSIC is less than that of the Fast Fourier Transform. *“The smaller deviation of the bathymetric error from the MUSIC estimator implies that the bottom features are estimated more accurately from the sharp peaks of the MUSIC spectrum than from those of the FFT beamformer. Therefore, the high-resolution performance does not*

come at the expense of lower accuracy, but actually the accuracy is slightly improved.” c.f. [19].

In yet another article by Pantzartzis et al. [18], the MUSIC algorithm is shown to lead to reliable bathymetric estimates also for more chaotic surfaces. Unfortunately, no quantitative comparison is given – only figures showing the original surface and the surface estimated by the MUSIC algorithm. These estimates seem to be acceptable.

Rao and Hari [24], find that for MUSIC “*minimal smoothing appears to be desirable to obtain reliable DOA estimates*”. However, smoothing can not be completely avoided, since it is necessary in a completely coherent environment.

Minimum-Norm [2, 11, 12, 13, 27, 31] was developed by Kumaresan and Tufts [12]. The basic idea was that if a weight vector $\mathbf{w} = [w_0 \cdots w_{L-1}]^T$ has the property $\mathbf{a}^H(\theta_m)\mathbf{w} = 0$, for $m = 1, \dots, M$, one can find a polynomial $D(z)$ with zeros at $e^{-jkd \sin \theta_m}$ (the z 's)

$$D(z) = \sum_{k=0}^{L-1} w_k z^{-k} \quad (5.16)$$

and thus find the θ 's. Since the noise subspace ideally is orthogonal to the signals, it seems plausible that \mathbf{w} should span the entire noise subspace to get the best estimate of $D(z)$. This will ensure $\mathbf{a}^H(\theta)\mathbf{w} = 0$ when θ is a signal's direction. The effect of letting \mathbf{w} span the whole noise subspace of $\hat{\mathbf{R}}$, is that “*the estimates of $e^{-jkd \sin \theta_m}$ are more accurate even at relatively low SNR values*” [12]. Further, if w_0 is constrained to be equal to 1 and the Euclidean length of \mathbf{w} is required to be minimum (hence the name), “*the $L-M$ extraneous zeros of $D(z)$ tend to be uniformly distributed within the unit circle and have less tendency for “false sources” or spurious estimates*” [12].

In [2], Buckley has a fine expression of Minimum-Norm as a constrained optimization procedure

$$\min_{\mathbf{w}} \mathbf{w}^H \mathbf{w} \quad \text{subject to} \quad \begin{bmatrix} 1 & 0 & \cdots & 0 \\ \hat{\mathbf{V}}_s^H \end{bmatrix} \mathbf{w} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (5.17)$$

The left part of (5.17) ensures a minimum norm of \mathbf{w} , while the right part constrains w_0 to equal 1 and further that \mathbf{w} is orthogonal to the signal subspace. Again, \mathbf{w} is best chosen as a linear combination of the eigenvectors in the noise subspace.

Now, how do we find \mathbf{w} ? Let

$$\hat{\mathbf{V}}_n = \begin{bmatrix} \mathbf{c}^T \\ \hat{\mathbf{V}}_n \end{bmatrix} \quad (5.18)$$

that is, \mathbf{c}^T consists of the first elements of the noise subspace eigenvectors, while $\tilde{\mathbf{V}}_n$ equals $\hat{\mathbf{V}}_n$, only that the first row is deleted. Then \mathbf{w} can be found as

$$\mathbf{w} = \begin{bmatrix} 1 \\ \tilde{\mathbf{V}}_n \mathbf{c}^* / \mathbf{c}^H \mathbf{c} \end{bmatrix} \quad (5.19)$$

which minimizes the norm of \mathbf{w} [12, 13]. Other estimates of \mathbf{w} are also possible, and can be found in [12, 27, 31].

A spectrum in accordance with (5.16) can now be formed as

$$P_{MN}(\theta) = \frac{1}{\mathbf{a}^H(\theta) \mathbf{w} \mathbf{w}^H \mathbf{a}(\theta)} \quad (5.20)$$

or as in [11]

$$P_{MN}(\theta) = \frac{1}{\mathbf{a}^H(\theta) \hat{\mathbf{V}}_n \hat{\mathbf{V}}_n^H \mathbf{W} \hat{\mathbf{V}}_n \hat{\mathbf{V}}_n^H \mathbf{a}(\theta)} \quad (5.21)$$

where \mathbf{W} is a matrix consisting of zeros only, except from the upper left element which equals 1. Such spectra will take on large values when θ equals a signal direction, since $\mathbf{a}^H(\theta) \mathbf{w} \approx 0$ in these cases. Remember that \mathbf{w} ideally is orthogonal to all signals.

Since (5.19) and (5.21) use more eigenvectors to find the \mathbf{w} orthogonal to the signal subspace than those defined by [12, 27, 31], the weights given by (5.19) and (5.21) are intuitively more appealing. The two latter expressions use the entire noise subspace to define the weights. The others [12, 27, 31] use the signal subspace and the assumption that $\mathbf{V}_n \mathbf{V}_n^H = \mathbf{I}$, which might be erroneous.

Rao and Hari [24] have developed some theory for the optimal number of subarrays, K , that should be used for spatial smoothing with the Minimum-Norm method. The best choice of K depends on how many signals that are present. For one signal, K is best chosen as $(L + 1)/4$ for low SNR - while for high SNR , K should be $(L + 1)/3$. When two signals are present, these choices change to $(L + 1)/8$ and $(L + 1)/5$. The same choices of K are valid when the signals are coherent, which is the more interesting case. Rao does not explicitly say where he draws the line between high and low SNR . But, from the examples he gives it is clear that 10dB is regarded as a high SNR .

Root-MUSIC For the MUSIC algorithm there is a polynomial-rooting version available, known as root-MUSIC. This has been studied in [4, 10, 11, 22, 27].

In MUSIC, the spectrum (5.15) took on large values because $\mathbf{a}^H(\theta) \hat{\mathbf{V}}_n \approx 0$, as for Minimum-Norm. This means that also MUSIC should be possible to define as a polynomial of the eigenvectors that span the noise subspace ($\hat{\mathbf{V}}_n$). Let $z = e^{-jkd \sin \theta}$. Define the vector $\mathbf{p}(z) = [1 \ z \ \dots \ z^{L-1}]^T$, we can then define polynomials $p_l(z)$ of the l 'th eigenvector of $\hat{\mathbf{R}}$ like this

$$p_l(z) = \mathbf{v}_l^H \mathbf{p}(z) \quad l = M+1, \dots, L \quad (5.22)$$

Each $p_l(z)$ now has M of its zeros at $e^{-jkd \sin \theta_m}$, for $m = 1, \dots, M$. What we really want, is to use the information in all noise eigenvectors simultaneously – much like we did for Minimum-Norm. This is achieved by using $\mathbf{p}^H(z)$ instead of $\mathbf{a}^H(\theta)$ in the denominator of (5.15), such that

$$\begin{aligned} P_{RM}(z) &= \mathbf{p}^H(z) \hat{\mathbf{V}}_n \hat{\mathbf{V}}_n^H \mathbf{p}(z) \\ &= \sum_{k=-(L-1)}^{L-1} c_k z^k \end{aligned} \quad (5.23)$$

where c_k is the sum of the elements on the k 'th diagonal of $\hat{\mathbf{V}}_n \hat{\mathbf{V}}_n^H$ [27]. Here $k < 0$ gives the k 'th sub diagonals, the ones below the actual diagonal, while $k > 0$ yields the k 'th super diagonals, those above.

Since $z = e^{-jkd \sin \theta}$, this polynomial is evaluated on the unit circle. Consequently, the roots of the $2(L - 1)$ degree polynomial (5.23) lie as mirrored pairs around the unit circle. Ideally, only the M signals have the right characteristics ($kd \sin \theta$) to actually be placed on the unit circle. Therefore, in root-MUSIC, the M largest roots, $\hat{z}_1, \dots, \hat{z}_M$, that lie inside or on the unit circle are taken as the *DOA* estimates (one could also use the M smallest outside). The angles of the M signals are then computed as

$$\hat{\theta}_m = \arccos\left(\frac{1}{kd} \arg\{\hat{z}_m\}\right), \quad m = 1, 2, \dots, M \quad (5.24)$$

As for MUSIC, Rao and Hari [24] find that for root-MUSIC, “*minimal smoothing appears to be desirable to obtain reliable DOA estimates*”. However, smoothing can not be completely avoided, since it is necessary in a completely coherent environment.

Alternative root-MUSIC In all papers I have read, root-MUSIC's *DOAs* are estimated by computing a polynomial giving roots inside and outside the unit circle. Still, all the authors say that the only interesting part is the M roots inside and closest to the unit circle [4, 10, 11, 22, 27].

Why not exploit the fact that $\hat{\mathbf{R}}$ is Hermitian? Why not simply compute the part of the polynomial that has roots inside the unit circle? This would lead to a smaller polynomial to solve. It can be achieved by replacing (5.23) with

$$P_{RM}(z) = \sum_{k=-(L-1)}^0 c_k z^k \quad (5.25)$$

Here, only the sum (c_k) of the elements on the sub diagonals and main diagonal of $\hat{\mathbf{V}}_n \hat{\mathbf{V}}_n^H$ are used.

Besides the lower computational load, there is one more thing to gain from this strategy. In general, the problem of finding multiple roots of a polynomial is an ill-conditioned mathematical problem [15]. In root-MUSIC, computed via (5.23), we try to find $2(L - 1)$ roots. Since, ideally, the M roots belonging to the signals lie on the unit circle – these M roots are actually very close to be double (multiple) roots.

Consequently, the results may be quite inaccurate for the M signal roots we are interested in. By computing root-MUSIC via (5.25), no double roots would exist, the new mathematical problem would be well-conditioned. Hence, more accurate *DOA* estimates should be achieved.

ESPRIT The idea behind ESPRIT [11, 23, 26], is to split the array in two subarrays, separated by a known displacement vector $\mathbf{\Delta}$ of magnitude Δ . The *DOA* estimates are then angles of arrival with respect to the direction of the vector $\mathbf{\Delta}$. For the Uniform Linear Array (ULA) we can split the array into two subarrays separated by distance d . Both subarrays are placed along the y -axis. The subarrays will be denoted Y' and Y'' . The signals received at sensor i in each of the subarrays, can be expressed as:

$$y'_i(t) = \sum_{m=1}^M s_m(t) a_i(\theta_m) + n_{y'_i}(t), \quad (5.26)$$

$$y''_i(t) = \sum_{m=1}^M s_m(t) e^{j\omega_0 d \sin \theta_m / c} a_i(\theta_m) + n_{y''_i}(t), \quad (5.27)$$

Note that for the ULA $y'_1(t) = y_1(t)$, $y'_2(t) = y''_1(t) = y_2(t)$, \dots , $y'_L(t) = y''_{L-1}(t) = y_{L-1}(t)$ and $y''_L(t) = y_L(t)$. Here $\mathcal{L} = L - 1$. The only sensors that are not shared by the two subarrays are sensor $y'_1(t)$ and sensor $y''_L(t)$, that is $y_1(t)$ and $y_L(t)$. If we combine the sensor outputs in each subarray into vectors, we can write (5.26) as

$$\mathbf{y}'(t) = \mathbf{A}' \mathbf{s}(t) + \mathbf{n}_{y'}(t), \quad (5.28)$$

$$\mathbf{y}''(t) = \mathbf{A}' \mathbf{\Phi} \mathbf{s}(t) + \mathbf{n}_{y''}(t) \quad (5.29)$$

Here \mathbf{A}' is the upper $(L - 1) \times M$ part of the steering matrix \mathbf{A} (confer (2.21) page 14), such that \mathbf{A}' is actually taken from

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}' \\ \text{last row} \end{bmatrix} = \begin{bmatrix} \text{first row} \\ \mathbf{A}'' \end{bmatrix} \quad (5.30)$$

$$\mathbf{A}'' = \mathbf{A}' \mathbf{\Phi} \quad (5.31)$$

Further, $\mathbf{s}(t)$ is the $M \times 1$ vector of signals, and the matrix $\mathbf{\Phi}$ is a diagonal $M \times M$ matrix containing the phase delays between the subarrays for the M signals

$$\mathbf{\Phi} = \text{diag}\{e^{j\phi_1}, \dots, e^{j\phi_M}\} \quad (5.32)$$

$\phi_m = kd \sin \theta_m$. The problem of estimating the *DOAs* is then to find these phase delays, hence finding $\mathbf{\Phi}$. The unitary¹ matrix $\mathbf{\Phi}$ relates the measurements from subarray Y' to those from subarray Y'' . Let

$$\bar{\mathbf{y}}(t) = \begin{bmatrix} \mathbf{y}'(t) \\ \mathbf{y}''(t) \end{bmatrix} = \bar{\mathbf{A}} \mathbf{s}(t) + \mathbf{n}_y(t) \quad (5.33)$$

$$\bar{\mathbf{A}} = \begin{bmatrix} \mathbf{A}' \\ \mathbf{A}' \mathbf{\Phi} \end{bmatrix}, \quad \mathbf{n}_y(t) = \begin{bmatrix} \mathbf{n}_{y'}(t) \\ \mathbf{n}_{y''}(t) \end{bmatrix} \quad (5.34)$$

¹square matrix, with orthonormal columns

The structure of $\bar{\mathbf{A}}$ is then exploited to obtain estimates of the diagonal elements of Φ .

In a perfect world, where no noise exists, an amount of samples sufficient to find M linearly independent vectors would span the signal subspace. That is, the M dimensional subspace of the L dimensional complex space, \mathbb{C}^L , spanned by $\bar{\mathbf{A}}$. Since the sensor arrays' translational invariance has induced a rotational invariance of these subspaces, we can use the information in this rotational invariance to find Φ . By knowledge of the covariance, \mathbf{R} , of the measurements, we can find the signal subspace. The eigenvectors corresponding to the M largest eigenvalues of \mathbf{R} give us $\mathbf{V}_s = [\mathbf{v}_1 | \cdots | \mathbf{v}_M]$, where the signal subspace $\Re\{\mathbf{V}_s\} = \Re\{\bar{\mathbf{A}}\}$.

Now, since $\Re\{\mathbf{V}_s\} = \Re\{\bar{\mathbf{A}}\}$ and $M < L$, there exists a unique nonsingular \mathbf{T} fulfilling

$$\mathbf{V}_s = \bar{\mathbf{A}}\mathbf{T} \quad (5.35)$$

Because of the invariance of the array, \mathbf{V}_s can be decomposed into $\mathbf{V}_{Y'} \in \mathbb{C}^{L \times M}$ and $\mathbf{V}_{Y''} \in \mathbb{C}^{L \times M}$ in the following way

$$\mathbf{V}_s = \begin{bmatrix} \mathbf{V}_{Y'} \\ \mathbf{V}_{Y''} \end{bmatrix} = \begin{bmatrix} \mathbf{A}\mathbf{T} \\ \mathbf{A}\Phi\mathbf{T} \end{bmatrix} \quad (5.36)$$

and then

$$\Re\{\mathbf{V}_{Y'}\} = \Re\{\mathbf{V}_{Y''}\} = \Re\{\mathbf{A}\}$$

$\mathbf{V}_{Y'}$ and $\mathbf{V}_{Y''}$ share column space, thus the rank of

$$\mathbf{V}_{Y'Y''} \stackrel{\text{def}}{=} [\mathbf{V}_{Y'} | \mathbf{V}_{Y''}] \quad (5.37)$$

is M . This implies that there exists a unique rank M matrix $\mathbf{F} \in \mathbb{C}^{2M \times M}$ such that

$$\begin{aligned} \mathbf{0} &= [\mathbf{V}_{Y'} | \mathbf{V}_{Y''}]\mathbf{F} = \mathbf{V}_{Y'}\mathbf{F}_{Y'} + \mathbf{V}_{Y''}\mathbf{F}_{Y''} \\ &= \mathbf{A}\mathbf{T}\mathbf{F}_{Y'} + \mathbf{A}\Phi\mathbf{T}\mathbf{F}_{Y''} \end{aligned} \quad (5.38)$$

\mathbf{F} spans the null-space of $[\mathbf{V}_{Y'} | \mathbf{V}_{Y''}]$. We can now rearrange (5.38), by defining the matrix Ψ .

$$\Psi \stackrel{\text{def}}{=} -\mathbf{F}_{Y'}[\mathbf{F}_{Y''}]^{-1} \quad (5.39)$$

Then (5.38) becomes,

$$\mathbf{A}\mathbf{T}\Psi = \mathbf{A}\Phi\mathbf{T} \Rightarrow \mathbf{A}\mathbf{T}\Psi\mathbf{T}^{-1} = \mathbf{A}\Phi \quad (5.40)$$

With \mathbf{A} of full rank, this implies

$$\mathbf{T}\Psi\mathbf{T}^{-1} = \Phi \quad (5.41)$$

and thus, the eigenvalues of Ψ equal the diagonal elements of Φ (Ψ and Φ are related through a similarity transformation). The columns of \mathbf{T} are the eigenvectors of Ψ .

Now, expressing Φ in (5.36) as (5.41) we obtain the following relation between $\mathbf{V}_{Y'}$ and $\mathbf{V}_{Y''}$

$$\mathbf{V}_{Y''} = \mathbf{V}_{Y'} \Psi \quad (5.42)$$

We can now compute the eigenvalues of Ψ using either a least-square or a total-least-square criterion. The least squares problem is easily solved by applying the pseudo-inverse of $\mathbf{V}_{Y'}$

$$\Psi = \mathbf{V}_{Y'}^\dagger \mathbf{V}_{Y''} \quad (5.43)$$

The diagonal elements in Φ will equal the eigenvalues of Ψ , and by applying (5.24) to these, the *DOA* estimates are found.

Also for ESPRIT, Rao and Hari [23] have come up with an optimal choice of K , still dependent upon *SNR* and how many signals that are present. For ESPRIT, these choices of K are $L/3$ (high *SNR*) and $L/4$ (low *SNR*) in the one signal case, while in the two signal case K should equal $(L + 1)/5$ (high to moderate *SNR*) and $(L + 1)/8$ (low *SNR*). Again, these choices are valid also for the coherent signal cases.

Chapter 6

Data and Evaluation Methods

This chapter presents the data that were available for this thesis. Further, a short description of how the algorithms were implemented is given. Finally, different ways of evaluating the performance of the various high-resolution methods are outlined.

6.1 Data Sets

The data available for this thesis were recorded by the Simrad EM3000 sonar, obtained as raw output of the array after bandpass filtering acoustic signals arriving at each sensor y_l , for $l = 1, \dots, 80$. The received signals were separated into in-phase and quadrature components, demodulated and then sampled at 14293 Hz. For each ping, these in-phase and quadrature data were written to file, represented as two bytes signed integers.

The transmit signal of EM3000 is a rectangular pulse of a sinusoidal wave with pulse length $150 \mu\text{s}$ and center frequency 300158 Hz. The receive array is a Uniform Linear Array of 80 elements. The distance between two neighboring elements is 2.55 mm. As $\lambda = c/f$, with c fixed at 1500 m/s, the intersensor spacing d is slightly larger than $\lambda/2$. Therefore, sidelobes alias into the mainlobe [7]. However, the array has to be steered to look for signals further to the side than approximately $\pm 86^\circ$ before aliasing becomes a real problem. In this work, such high steering angles are not used. Consequently, the spatial sampling avoids aliasing and the array can be assumed unambiguous (Sec. 2.5). If aliasing had become a problem for lower steering angles, weighting of the sensing elements (Sec. 4.1) had to be applied to avoid the negative effects of the aliasing.

As mentioned in section 4.3, the sample frequency (14293 Hz) and pulse length (150 μs) of EM3000 excludes time averaging. Thus, the estimated covariance matrix $\hat{\mathbf{R}}$ is significantly affected by any noise present. This can, to some extent, be improved by the other methods given in chapter 4.

The low number of stationary samples affects the resolution, the mean-square error

and the bias of the methods in unpredictable ways [7, 13, 27]. Ideally, the resolution of adaptive beamformers should be measured asymptotically, because: “*Direct analytic derivation of an adaptive technique’s resolution has defied all attempts to date*”, c.f. [7, Sec. 7]. The low number of stationary samples obtained by the EM3000 effectively restricts us from doing this. Further, since $\hat{\mathbf{R}}$ is estimated from noisy samples, the steered response of any beamformer will differ from its ideal form. Thereby, the accuracy of the location of signals may be affected by the resolution – if resolution is defined through the steered response [7]. Research has been done in order to gain some insight about these effects. In [27], Srinivas and Reddy analyze the finite data performance of MUSIC and Minimum-Norm, and derive expressions for the mean square error of the *DOA*. While Li and Lu [13], look at the bias of subspace-based methods – and try to say something about their performance for different *SNRs*, number of sensors in the array and various amounts of snapshots. These topics were mentioned in chapter 1.

In this study, 1760 snapshots are used, half of them containing signals from “normal incidence” echoes and half of them obliquely arriving echoes. These snapshots are taken from 22 different pings, which again come from 6 different ping series. The “normal incidence” signals are defined as signals arriving from θ values between -25° and 25° (Fig. 6.1). This wide area is used to obtain enough snapshots containing “normal incidence” signals for the evaluation in section 7.1. The definition of the obliquely arriving signals is stronger, these signals arrive from θ values above 40° to either side.

6.2 Implementation Issues

The data were processed sequentially, one time sample (snapshot) at a time. For each snapshot, a “power” spectrum is computed (Fig. 6.4).

For one ping, many succeeding power spectra (one for each snapshot) constitute a spectrum for the entire bottom scene viewed by one ping. Such spectra are visualized by letting different “powers” or amplitudes map to different colors (Fig. 6.2(a)). This final spectrum composes a polar coordinate picture of the bottom. A problem with this picture, is that secondary bottom echos may be present – in addition to other noise. By post processing this picture in some intelligent way, we are hopefully able to distinguish between actual bottom returns and noise or extraneous echoes. This is called *bottom detection*. We can then do a simple scanconversion from polar coordinates to the true bottom map, which is given as x , y and z coordinates.

Bottom Detection One way of doing the bottom detection, is to run through every vertical line in the spectrum and find the snapshot with the largest amplitude (Fig. 6.2). This may be done together with thresholding the spectrum, in order to get rid of the least significant parts of it. All vertical lines do not necessarily contain bottom returns, this will depend on how many snapshots we take of a ping and the geometry of the bottom. This procedure is a simple bottom detection algorithm,

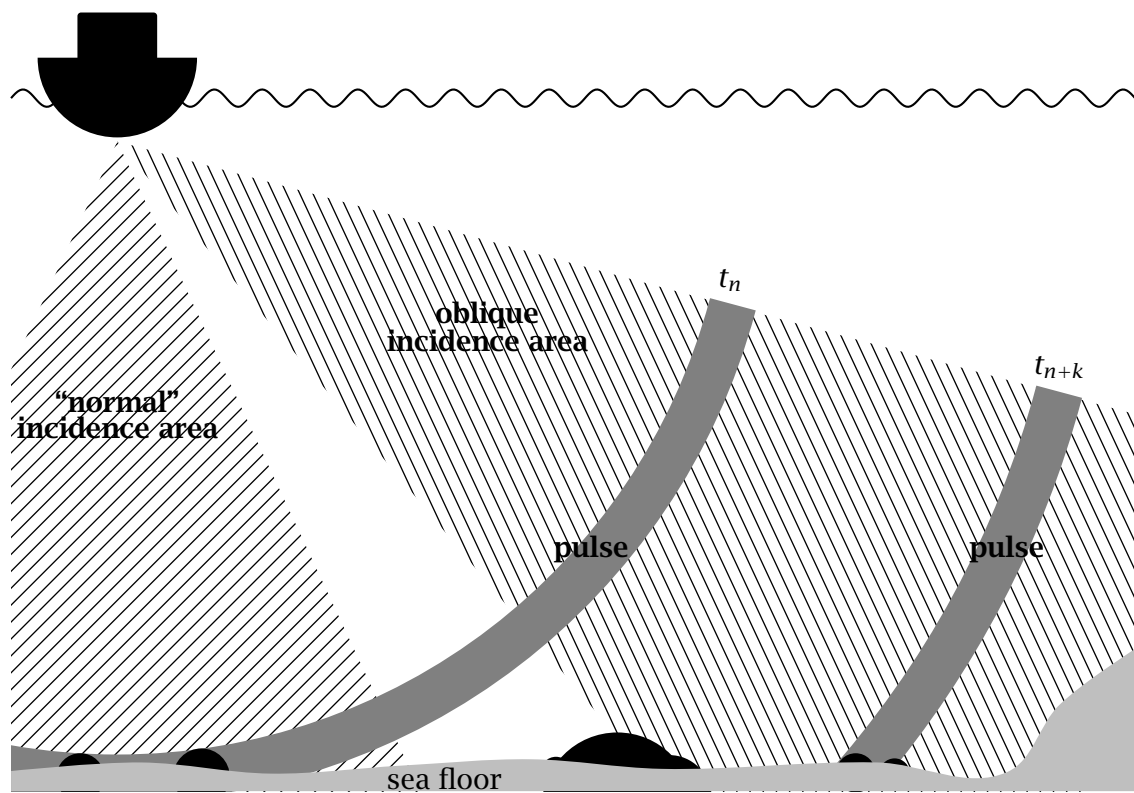
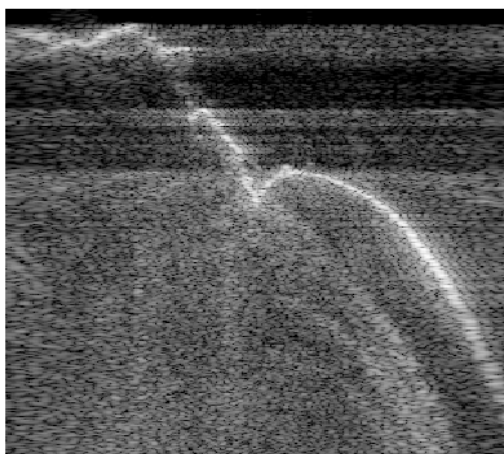


Figure 6.1: Definition of the echo incidence areas used for the quantitative evaluation. The sea floor coverage of the pulse is also shown (exaggerated pulse length). Ship viewed from behind.



(a) Polar coordinate spectrum of a quite complex bottom.



(b) Polar coordinate picture after bottom detection.

Figure 6.2: Spectrum in polar coordinates. Vertical axis time, horizontal axis angle.

which is based on the knowledge that not more than one signal can impinge on the array from a given direction. If two signals arrive from the same direction, the one with lowest amplitude (hopefully the second) is considered to be a secondary echo. This must be discarded to give correct bottom estimates. The result of such a bottom detection algorithm can be seen in figure 6.2(b). Other more sophisticated approaches must be used in more difficult scenarios.

Estimation of M At each snapshot, the high-resolution methods (except Minimum-Variance) depend on an estimate of the number of signals M currently received by the array. In this work, M is estimated through simple combinations of the first few eigenvalues of $\hat{\mathbf{R}}$, after any smoothing or noise subtraction operations are applied to it (Sec. 4.3 through 4.6). This means that M *may vary with what operations we apply to $\hat{\mathbf{R}}$* .

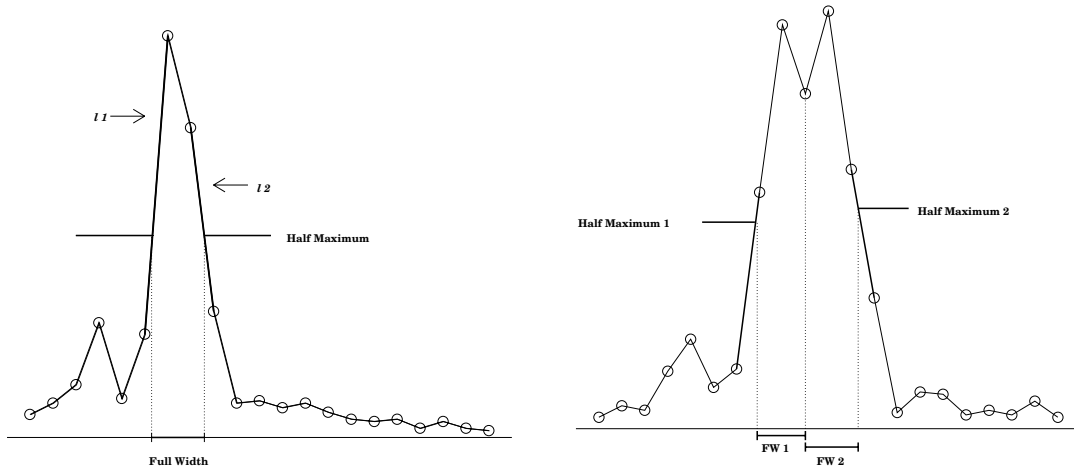
For the data sets used, the following computation of M seems to give reasonable results. For each of the first four eigenvalues that are larger than $0.95\mu(\boldsymbol{\lambda}) - 0.38std(\boldsymbol{\lambda})$, M is increased by one. Here $\boldsymbol{\lambda}$ is a vector of the six largest eigenvalues, μ is its mean value and std its standard deviation. This means that not more than four signals will be accepted at any discrete time step. This strategy was found by examining several snapshot's power spectra and eigenvalues, and should by no means be regarded as optimal or statistically founded - it is plain engineering and a fast procedure.

In time periods where no signals are present, this strategy will result in four accepted signals. The six largest eigenvalues roughly equal the mean. To avoid this one can restrict the number of signals when the standard deviation is less than some predefined value. When the standard deviation is very low, it is quite likely that the variation seen in the eigenvalues is simply due to the the one snapshot case ($N = 1$). It is highly unlikely that we have six signals, with the same SNR , impinging at the same time.

The spectra may be computed via any of the methods from chapter 5. Each of these methods are based on the covariance matrix $\hat{\mathbf{R}}$, which in turn may be computed by any combination of the methods in chapter 4. In addition, we might apply weighting of the sensors before computing $\hat{\mathbf{R}}$. Consequently, many algorithm combinations have to be run in order to find the best way of estimating $\hat{\mathbf{R}}$ for any high-resolution method.

6.3 Evaluation Methods

In this section, the methods for evaluating the performance of the high-resolution methods will be explained. The performance can to some extent be evaluated by quantitative measures, but we also compare the different methods by a visual examination of the spectra.



(a) The Full Width is computed at the Half Maximum height, even though there are no points corresponding to this level.

(b) FWHM when two peaks have no values lower than HM between them.

Figure 6.3: Explanation of different FWHM approaches.

6.3.1 Quantitative Evaluation

If we define the resolution to reflect the width of the steered response about the direction of propagation, the Full Width Half Maximum (FWHM) of the signal peaks is an acceptable measure of a method's resolution (Fig. 6.3(a)). Unfortunately, it says nothing about how accurate the method computes the directions of arrival (DOA).

If one considers the highest peaks in two different spectra, for the same snapshot, it is not enough to have a smaller FWHM (better resolution) to be regarded as the best method. The best method must also have its peak at the source's location.

Since we do not really know the *true* bottom in these experiments, we assume that the Fourier transform spectrum "tells" us what is reasonably close to the truth. Now, if a method's peak-width is smaller than that of the FT spectrum, and in addition has its peak-width within the area covered by the FT spectrum's peak-width; this method will be considered better than the Fourier transform. The area covered by the peak-width, for instance 33–37 degrees, will be called the width-location (WL). The best would be a method that was capable of resolving two sources in an area where the FT spectrum is only capable of finding one source, hence one peak.

The FWHM is computed for a given peak (Fig. 6.3). First one finds the Half Maximum (HM) of the peak, computed as half way between the minimum value and the magnitude of the given peak in the spatial spectrum at time t . When the HM has been found, one has to compute the Full Width (FW) of the peak at this level. Computing the width is done by defining a linear function, $l(\cdot)$, running through the first point (θ value) yielding a value lower than HM, and the point yielding a value higher than HM. The slope of the linear function is computed, and from this function a

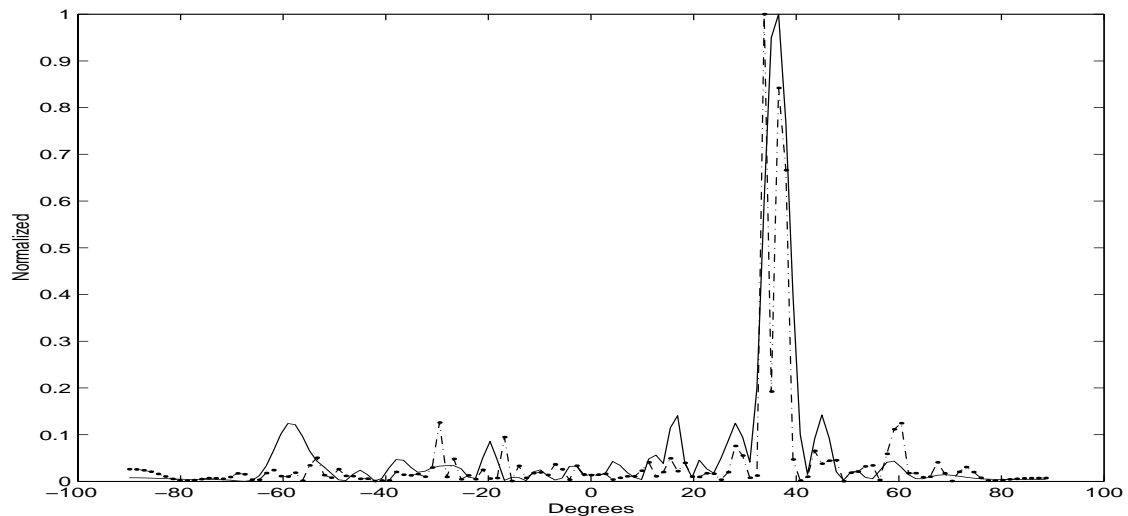


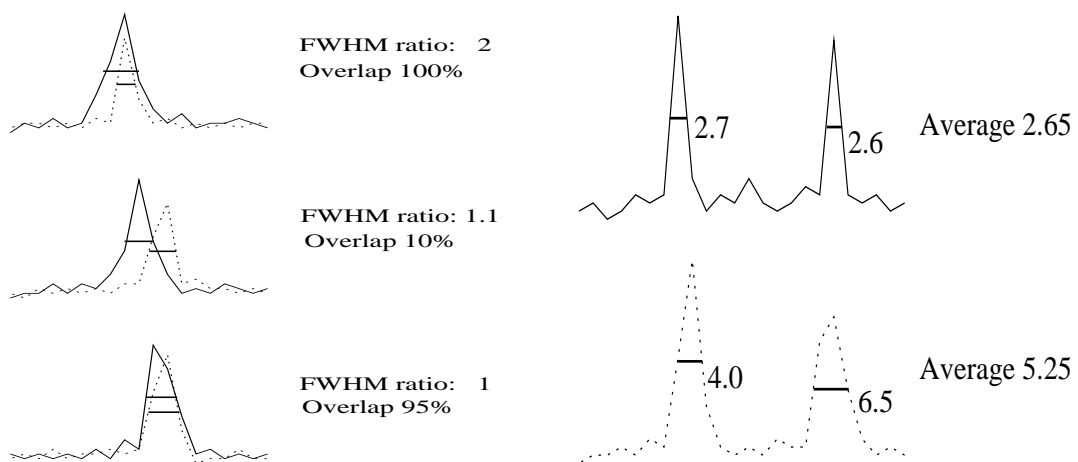
Figure 6.4: Power spectrum (dashdot) with two peaks inside one peak of another power spectrum (solid). The dash dotted power spectrum resolving two sources is computed for an ULA with 80 elements. Solid power spectrum resolving only one source - ULA with 30 elements. Sources propagating from 34 and 37 degrees.

point corresponding to the exact HM is found even though this point may not exist since only angular increments of length h in degrees defines the corresponding axis. This is done both to the left and right of the given peak location, as with l_1 and l_2 in figure 6.3(a). Given these two points, one knows the distance between them and hence the FW. A special case occurs when two peaks have no values lower than HM between them. Then, one of the FW points belonging to a peak is defined as where the spectrum is no longer declining, but begins to increase towards the next peak (Fig. 6.3(b)).

Now, “measures of goodness” are FWHMs along with the amount of an arbitrary method’s width-location overlapped by the FT spectrum’s width-location. 100% of a method’s width-location within the FT spectrum’s width-location is excellent, the two methods are then performing equally well. 200% would be even better. This occurs when a method has resolved two distinct peaks totally within the area enveloped by only one peak in the FT spectrum. In figure 6.4 one can see that one spectrum resolves two sources, while the other spectrum only finds one source. This event would lead to a percentage of width-location overlap equal to 184.4 %, which is quite good. The measure that quantifies the amount of close peaks enveloped by only one peak in the Fourier spectrum is explained on page 52.

Peak-Width Ratio

All the ratios, for all snapshots, of the FWHM of the FT spectrum’s highest peak to an arbitrary method’s highest peak’s FWHM have been measured. All the snapshot’s ratios are time-averaged and a final value is computed (Fig. 6.5(a)). If this value is



(a) Average FWHM ratio (PWR) for the highest peaks is 1.4. The averaged overlap (WLO) between the methods is 68.3%. FT spectrum solid. Three snapshots.

(b) The $PWR(M)$ for **one snapshot** is 0.5. The average of the M FWHM's of the FT (top), is divided by the average of the FWHM's of a high-resolution method (bottom) The ratios for several snapshots will be time-averaged, as in the left figure. The total amount of overlap, $WLO(M)$, is also computed.

Figure 6.5: Different averages of FWHM and width-location overlap. Approaches for highest peak, and for “all” peaks.

larger than one, the method we are studying has a better resolution than the FT. This value is denoted Peak-Width-Ratio, PWR . The PWR is the expression (6.1), only that M is set to one - no matter what M 's actual size is (Fig. 6.6).

Next, for each snapshot, an average of the FWHM of *all significant* peaks (assuming one knows how many sources, M , there are) of the FT spectrum has been compared to the FWHM average of "all" the peaks in the spectrum of another method (Fig. 6.6). Then, a time averaged ratio as PWR is computed from these ratios, it is the *mean* Peak-Width Ratio $PWR(M)$. This means that there are actually two different PWR measures. Fortunately, both of them can be defined by the same formula if one uses a parameter M - the number of signals. The measure between the FT and a high-resolution beamformer (HR) is given by

$$PWR(M) = \frac{1}{\mathcal{N}} \sum_{n=1}^{\mathcal{N}} \frac{\sum_{m=1}^M FWHM_{FT}(n, m)}{\sum_{m=1}^M FWHM_{HR}(n, m)} \quad (6.1)$$

where n is an index for a snapshot and m an index for a peak. For each snapshot, the FWHM of the M highest peaks obtained by the FT are compared to those obtained by a high-resolution beamformer (Fig. 6.5(b)). The ratio is then averaged for all \mathcal{N} snapshots used for measuring the performance.

If the PWR is larger than one, the high-resolution method has a better resolution than the FT - when only considering the highest peak at each snapshot. Given a $PWR(M)$ larger than one, roughly all peaks of the high-resolution method have a smaller FWHM than those of the FT.

Width-Location Overlap

A percentage of the overlap between the mainlobes of two methods will be computed as mentioned earlier. For each snapshot's highest peak, we compute how much of a method's peak's width-location is covered by the FT spectrum's peak's width-location (Fig. 6.5(a)). It is thus denoted the Width-Location Overlap, WLO . As for the PWR , the WLO is defined by letting M in (6.2) equal one - regardless of M 's size (Fig. 6.6). Finally, results for all snapshots are averaged.

A percentage of the total amount of overlap between the M peak's width-locations is also computed, by comparing the width-locations of the M signals for one snapshot in the FT spectrum with all of the M width-locations in a high-resolution method's spectrum. This is done for each snapshot, and then time-averaged. See figure 6.5(b). For this comparison, 100% is the best possible and means that all peaks in the spectra are at the same places in both methods (or perhaps several of the peaks in a method's spectrum within one of the FT spectrum's peaks). The measure is denoted

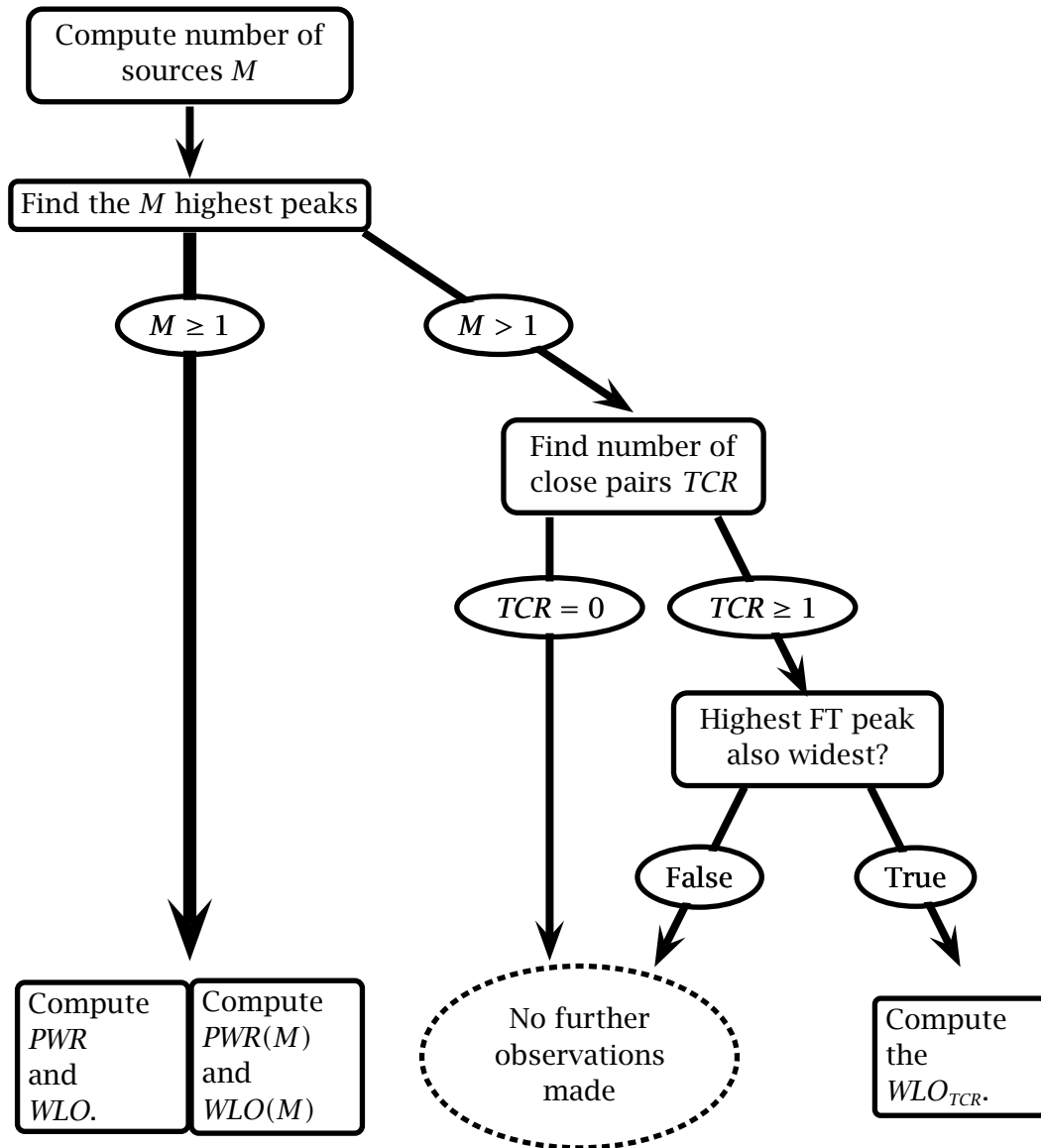


Figure 6.6: Flowchart for quantitative evaluation. This is performed at each snapshot, and later averaged over all snapshots.

the *mean* Width-Location Overlap, $WLO(M)$.

$$WLO(M) = \frac{1}{\mathcal{N}M} \sum_{n=1}^{\mathcal{N}} \sum_{j=1}^M \sum_{i=1}^M \frac{WL_{HR}(n, i) \cap WL_{FT}(n, j)}{FWHM_{HR}(n, i)} \quad (6.2)$$

It gives the percentage of the high-resolution method's peaks which are located within the FT peaks. This value is also averaged for all \mathcal{N} samples and all M peaks in each sample. The WLO , or $WLO(1)$, gives us the overlap between the highest peaks, while $WLO(M)$ gives us the averaged overlap between the M highest peaks - where M is the number of signals for the current snapshot (Fig. 6.6).

Closely spaced but resolved sources

The total number of closely spaced sources (dash dotted spectrum Fig. 6.4) a method has resolved is added up. This is achieved by, for every snapshot, counting how many closely spaced sources or close pairs there are (Fig 6.6). Sources are defined as closely spaced when the locations of the peaks are less than 2.5 degrees apart. It is simply assumed that when two closely spaced peaks are observed in the spectrum, these two peaks are closely spaced sources that are resolved. This measure is averaged over all snapshots, and will tell us something about a method's capability of resolving closely spaced sources. The measure is denoted Total of Close but Resolved sources, TCR .

Overlap for the close sources

A measure similar to WLO is also computed for the methods that resolve closely spaced sources. This is done by first examining the spectrum of the high-resolution methods, to see if close sources are present (Fig 6.6). Possible close sources are then compared to the peak of the Fourier spectrum that is *both* highest and widest. The idea is that wide peaks in the Fourier spectrum are the ones most likely of not having resolved closely spaced sources.

However, the widest peak may also be a random wide peak consisting of noise only. This thought is based on M being chosen properly. If M is correct, and a high-resolution method resolves closely spaced sources, one of the peaks in the Fourier spectrum will actually be noise - given that the Fourier method does not resolve the close sources. Therefore, the comparison is restrained to only the highest peak, if this is also the widest (Fig 6.6). Unfortunately, it was not thought of comparing them with the peak of the Fourier spectrum that is closest to these sources.

Thus, the WLO is computed by keeping j in (6.2) fixed such that it equals the highest and widest peak of the FT, while i is the index corresponding to the close peaks of the high-resolution beamformer. We do not divide by M in this measure.

This measure is somewhat dangerous, errors may well be present since several assumptions are made. Still, this way of measuring the *WLO* might be as good as comparing the width-location of the closely spaced sources to the peak in the Fourier spectrum that is closest to these sources. Because, from the conventional point of view the highest peaks of the Fourier spectra are the most important. In the Fourier approach M is not measured at all, usually a threshold is applied to the spectrum, and the bottom is found by some bottom detection strategy.

This measure of overlap will only be averaged over the snapshots where closely spaced sources are present. Thus, \mathcal{N} in (6.2) is replaced by the sum of all snapshots' *TCR*.

Computation Time

Finally, the execution time was computed for all methods - using an angular increment of a half degree, $h = 0.5$. The execution time for each method has been computed by using the floating point operation counter (*flops*) in MATLAB. The average amount of Mega-flops per snapshot was measured.

Computation time is an important issue for bathymetric sonars. We need fast algorithms. In addition to the differences between the methods, this measure will depend upon what operations we apply to $\hat{\mathbf{R}}$.

Comments to the measures

Before the evaluation in chapter 7 is started, a few comments about some of the above measures might be of interest.

The conventional Fourier transform suffers from a decreasing resolution for signals arriving from increasingly oblique angles. Therefore, the evaluation should be performed in such a way that one is able to say something about the performance of the methods for "normal" incidence, as well as obliquely incoming signals. This is achieved by inspecting the Fourier spectra before doing the quantitative evaluation. By this inspection, one is capable of deciding which snapshots that yield what sort of *DOA* (normal or oblique) for all pings. One can then evaluate different snapshots for different pings, such that measures are done for both normal incident and obliquely incoming signals.

The *PWR* measure is not applicable for root-MUSIC and ESPRIT. Since these methods estimate the *DOA* via zeroes in the z -plane (not quite true for ESPRIT), the width of the M peaks can be defined to be of any size. Still, the *PWR* measure can be used in a fruitful way.

The *PWR* of ESPRIT and root-MUSIC tells us something about the peak-width of the Fourier method. Having defined the peak-width of ESPRIT and root-MUSIC in the exact same way for all snapshots, one is able to tell how the width of the Fourier

Covariance Estimation Methods		
Forwd. /Backwd. Averaging	Noise Subtracted	Denoted
NO	NO	$\hat{\mathbf{R}}_{fs}^K$ (plain)
NO	YES	$\hat{\mathbf{R}}_{fs}^K - \hat{\mathbf{N}}$
YES	NO	$\hat{\mathbf{R}}_{fbs}^K$ (plain)
YES	YES	$\hat{\mathbf{R}}_{fbs}^K - \hat{\mathbf{N}}$

Table 6.1: These parameters were combined with 0, 2, 5, ..., 26 and 29 subarrays for beamforming.

spectra's peaks change for different types of signals. As stated above, measures are done for signals of normal incidence, as well as for obliquely incoming signals. Therefore, the *PWR* of either ESPRIT or root-MUSIC give us information about the Fourier method's difference in resolution capabilities for signals arriving from these directions.

Since the number of signals, M , is not found in an optimal way, the quantitative part of the evaluation is not necessarily an acceptable measure of how "good" bathymetric pictures one gets. Yet, it is a reasonable comparison of one and one snapshot. Where there are large peaks in the FT spectra, composed by signals or noise, one should also find peaks with the high-resolution methods. Because of this, it is not necessary to look at several independent data sets in the quantitative part, it is sufficient to look at several snapshots.

As mentioned in the beginning of section 6.3.1, it is not enough for a method to have the narrowest peaks to be regarded as the most promising high-resolution method. Thus, *the WLO of a method will be considered as more important than the PWR*. Also, a high-resolution method should be able to resolve closely spaced sources - therefore, *the TCR measure will be of some essence too*. But, we have to bear in mind that this measure is somewhat dangerous. It is based on the assumption that when two peaks are observed close to each other in the spatial spectrum, these two peaks are actually two closely spaced but resolved sources. This need not be the case! Nevertheless, *the WLO of the assumed close sources is computed*. We need fast methods in bathymetric sonars. Thus, *the amount of M-flops is also of importance*.

The above mentioned quantitative results have been computed for various data sets. All data sets have been processed with the same parameter combinations, as outlined in table 6.1.

6.3.2 Qualitative Comparison

As stated earlier, the quantitative part of the evaluation is not necessarily an acceptable measure of how good bathymetric spectra the high-resolution methods are able to compute. To complicate things further, methods yielding what appears to be

acceptable spectra may not necessarily be the same methods that give good results after bottom detection. Thus, in addition to evaluating the methods on a sample to sample basis, as in the quantitative part, we have to view the resulting pictures to get a qualified impression of which methods to prefer. The high-resolution methods that have a high *WLO* should give bathymetric spectra “equal” to the Fourier method, but the quantitative measures say nothing about the level of the background noise.

Chapter 7

Experimental Results

In this chapter, the results of the different high-resolution methods are given.

For each of the high-resolution methods (Sec. 5.2), the most prominent trends of each measure defined in section 6.3 are summed up. When all measures for a high-resolution method are done, a conclusion of how to estimate the covariance matrix, $\hat{\mathbf{R}}$, for this particular method is drawn. That is, *the parameter choices for estimation of $\hat{\mathbf{R}}$ that gave the best results on an overall basis are found for each method*. Finally, the results of the methods are compared for these specific parameter combinations in table 7.8.

Having done the quantitative comparison, we are ready to compare the methods visually in section 7.4.

7.1 Quantitative Observations

First the results of each method are given, and the best way of estimating $\hat{\mathbf{R}}$ is found for the current method. The *Width-Location Overlap (WLO)* (p. 52) of a method is considered to be more important than the *Peak-Width Ratio (PWR)* (p. 50). The *Total of Close but Resolved sources (TCR)* (p. 52) measure will also be of some essence, and the *WLO* of these assumed close sources (p. 52) has to be high. The computational load (measured in M-flops) is also important, fast algorithms are needed for bathymetric sonars.

In this study, 1760 snapshots are used, half of them containing signals from “normal incidence” echoes (-25° to 25°) and half of them obliquely arriving echoes (above 40° to either side). These snapshots are taken from 22 different pings, which again come from 6 different ping series. Thus, \mathcal{N} in the quantitative measures (Sec. 6.3.1) equals 880, both for normal and oblique incidence signals.

In the second section the best results for each method are compared. This will give us insight about what results to expect for each method, point out the differences

between the methods and also give us an idea of what the spectra will look like, both before and after bottom detection.

For most methods the PWR and $PWR(M)$ are highly correlated. Therefore, only the PWR will be discussed for these methods. This also applies to the WLO and the $WLO(M)$.

7.1.1 Minimum-Variance

Peak-Width Ratio To ensure the highest PWR for most subarray sizes, \hat{R} should be estimated by forward/backward smoothing in combination with noise removal (Fig. 7.1(a)/Tab. 7.1).

For less than 17 subarrays, quite similar results were obtained for the PWR whether plain forward-only smoothing, plain forward/backward smoothing or forward/backward smoothing in combination with noise removal was applied (Fig. 7.1(a)). Forward-only smoothing in combination with noise removal gave a lower PWR than the other estimation techniques (Fig. 7.1(a)).

When 17, or more, subarrays were applied, a higher PWR was obtained for forward/backward smoothing by removal of the estimated noise (Fig. 7.1(a)). Forward/backward smoothing was preferable to forward-only smoothing, whether the estimated noise was removed or not.

The same trends were apparent for both the normal incidence and obliquely incoming signals (Fig. 7.1(a)). The PWR was somewhat higher for signals arriving from oblique angles.

Similar results as those for the PWR were obtained for the $PWR(M)$, but with a larger difference between forward/backward smoothing with or without removal of the noise estimate.

Width-Location Overlap Plain forward/backward smoothing lead to a higher WLO than what was achieved by using plain forward-only smoothing (Fig. 7.1(b)). When more than 14 subarrays were used, the WLO achieved by plain forward/backward smoothing was mostly higher than what was obtained by the other methods (Fig. 7.1(b)). For 17 subarrays or more, the WLO was mostly above 60%. This is the suggested way to estimate \hat{R} in order to get the highest WLO for Minimum-Variance (Tab. 7.1).

Plain forward-only smoothing showed an increase in WLO as more subarrays were applied (Fig. 7.1(b)), although it was roughly 0 until about 8 subarrays were applied. The increase was most significant when more than 17 subarrays were applied, and resulted in a WLO of above 60% of the FT when 26 or 29 subarrays were used.

With few subarrays, neither method for estimating \hat{R} gave high WLO values – even though removing the estimated noise gave somewhat higher values than not remov-

ing it. This was clear both for forward/backward averaging and forward/backward smoothing (Fig. 7.1(b)).

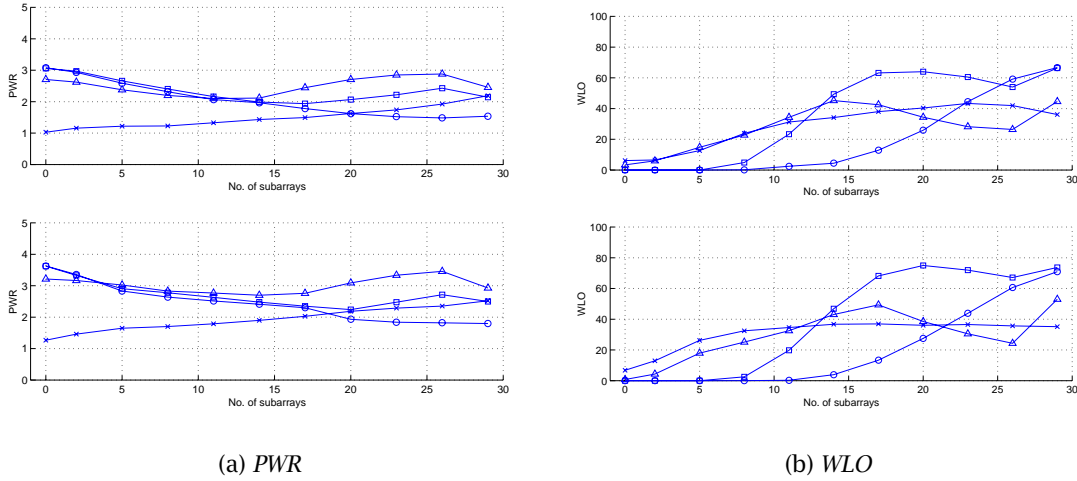


Figure 7.1: *PWR* and *WLO* results for Minimum-Variance. Upper part, normal incidence signals. Lower part, obliquely arriving signals. Plain forward-only smoothing \circ . Forward-only smoothing with noise removal \times . Plain forward/backward smoothing \square . Forward/backward smoothing with noise removal \triangle .

The *WLO* and *WLO(M)* gave similar results, except that the *WLO* was highest for signals arriving from oblique angles, while the *WLO(M)* was highest for signals of normal incidence. The difference was small though.

Resolution (TCR) Minimum-Variance was not very capable of resolving sources that were closely spaced. Minimum-Variance resolved between 0 and 0.5 closely spaced sources per snapshot, depending on how $\hat{\mathbf{R}}$ was estimated.

Forward/backward smoothing had to be used in order to resolve the highest amount of close sources. By applying plain forward/backward smoothing, there was a clear increase in ability of resolving closely spaced sources when using 17 subarrays or more (Tab. 7.1). Removing the estimated noise reduced the performance of forward/backward smoothing.

WLO of closely spaced sources To achieve a high *WLO* for the closely spaced sources, forward/backward smoothing had to be applied. To get *WLO* values higher than 100%, both for signals of normal incidence and oblique angles, 20 or more subarrays had to be used (Tab. 7.1). For this high number of subarrays, there was little difference in *WLO* whether the noise estimate was removed or not. For less than 17 subarrays, noise removal was preferable.

Minimum-Variance			
Measure	Resulted in	for $K \in$	when $\hat{\mathbf{R}}$ was estimated as
PWR	> 2.4	$[0, 29]$	$\hat{\mathbf{R}}_{fs}^K - \hat{\mathbf{N}}$
WLO	$> 60\%$	$[17, 29]$	$\hat{\mathbf{R}}_{fs}^K$ (plain)
TCR	> 0.21	$[17, 29]$	$\hat{\mathbf{R}}_{fs}^K$ (plain)
WLO_{TCR}	$> 106\%$	$[20, 29]$	$\hat{\mathbf{R}}_{fs}^K$ (plain)
M-flops	$19.8 \rightarrow 13.8$	$[20, 29]$	$\hat{\mathbf{R}}_{fs}^K$ (plain)

Table 7.1: The best results for the Minimum-Variance algorithm, based on the average between signals of normal incidence and oblique angles.

Computation Time The floating point operations measured by `flops` in MATLAB showed that the computational load was greatly reduced by applying subarrays in the estimation of $\hat{\mathbf{R}}$. The amount of Mega-flops decreased from about 50 until roughly 17 by increasing the amount of subarrays from 0 to 29 (Fig. 7.2(a)). The other methods for estimating $\hat{\mathbf{R}}$ were not as important, even though the computational burden was increased both by forward/backward smoothing and noise removal. The difference between plain forward-only smoothing (fastest) and forward/backward smoothing in combination with noise removal (slowest), was 14 M-flops for no subarrays and 4 M-flops for 29 subarrays.

Conclusion For Minimum-Variance, the PWR was quite high for all subarrays when applying forward/backward smoothing. When many subarrays were applied, it became higher by removing the estimated noise. The WLO declined when noise was removed. From Tab. 7.1 it is seen that more than 17 subarrays should be applied – not only for the WLO , but for the TCR as well. For the WLO of the closely spaced sources, it was advantageous to use many subarrays – 20 or more. Only with regard to the M-flops it seemed that forward-only smoothing was better than forward/backward smoothing. Since many subarrays should be applied, the difference is small.

Thus, for Minimum-Variance, we conclude that $\hat{\mathbf{R}}$ should be estimated by applying *plain forward/backward smoothing* while smoothing with between 20 and 29 subarrays. Since Minimum-Variance had a WLO of 66% for 17 subarrays, 70% for 20, 66% for 23 and 70% for 29 subarrays, it is safest to *use 20 subarrays* (Tab. 7.8).

7.1.2 Eigen-Vector

The results for the Eigen-Vector algorithm were much similar to those of Minimum-Variance. This was expected since the Eigen-Vector algorithm is the same as the Minimum-Variance algorithm, except that it only uses the L - M smallest eigenvalues of $\hat{\mathbf{R}}$ and their associated eigenvectors, whereas Minimum-Variance uses the entire $\hat{\mathbf{R}}$.

Peak-Width Ratio For the *PWR* measure, the only noticeable difference was that when $\hat{\mathbf{R}}$ was estimated without noise removal – the *PWR* was lower for Eigen-Vector than for Minimum-Variance when applying few subarrays. Similar results were observed for the *PWR(M)* measure.

Width-Location Overlap Also the *WLO* of the Eigen-Vector and Minimum-Variance algorithms was quite similar (Tab. 7.2). Only for few subarrays and noise removal a small difference occurred. The *WLO* increased from 0% for Minimum-Variance, until 5 to 10% for Eigen-Vector, when 0, 2 or 5 subarrays and no noise removal was applied. The same was observed for the *WLO(M)*.

Resolution (TCR) The ability to resolve closely spaced sources was quite similar for the Minimum-Variance and Eigen-Vector algorithms.

WLO of closely spaced sources The *WLO* for the close sources showed a different trend for the Eigen-Vector algorithm than for the Minimum-Variance algorithm.

For less than 17 subarrays the Eigen-Vector algorithm showed little difference between forward/backward smoothing with and without removal of the estimated noise, as opposed to Minimum-Variance – where noise removal improved the *WLO* of the closely spaced sources. The results when removing the estimated noise were fairly similar for Minimum-Variance and Eigen-Vector. When noise was not removed, and few subarrays were applied, the Eigen-Vector algorithm gave a few percentages higher *WLO* values for the close sources than the Minimum-Variance algorithm did. This was clear for both forward/backward smoothing and forward-only smoothing.

When forward/backward smoothing with more than 17 subarrays was applied, the Eigen-Vector algorithm (Tab. 7.2) performed quite similar to the Minimum-Variance algorithm.

Eigen-Vector			
Measure	Resulted in	for $K \in$	when $\hat{\mathbf{R}}$ was estimated as
<i>PWR</i>	> 2.4	[0, 29]	$\hat{\mathbf{R}}_{fs}^K - \hat{\mathbf{N}}$
<i>WLO</i>	> 56%	[14, 29]	$\hat{\mathbf{R}}_{fs}^K$ (plain)
TCR	> 0.20	[14, 29]	$\hat{\mathbf{R}}_{fs}^K$ (plain)
<i>WLO</i> _{TCR}	> 108%	[17, 29]	$\hat{\mathbf{R}}_{fs}^K$ (plain)
M-flops	50 → 33	[20, 29]	$\hat{\mathbf{R}}_{fs}^K$ (plain)

Table 7.2: The best results for the Eigen-Vector algorithm, based on the average between signals of normal incidence and oblique angles.

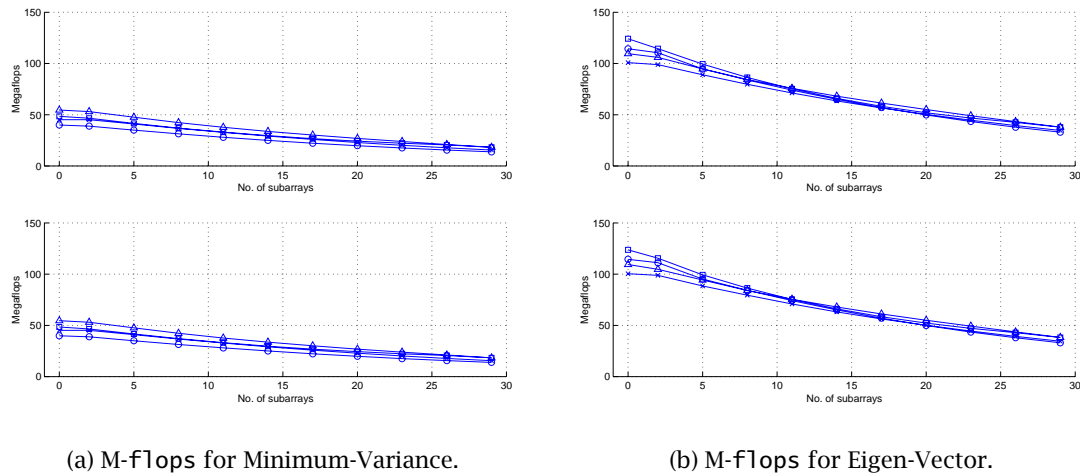


Figure 7.2: M-flops for Minimum-Variance and Eigen-Vector. Upper part, normal incidence signals. Lower part, obliquely arriving signals. Plain forward-only smoothing \circ . Forward-only smoothing with noise removal \times . Plain forward/backward smoothing \square . Forward/backward smoothing with noise removal \triangle .

Computation Time The need to compute the eigenvectors and values led to an increase in computational load for Eigen-Vector compared to Minimum-Variance (Fig. 7.2). Even though a decrease in M-flops was observed when increasing the number of subarrays, it is clear that Minimum-Variance is a faster method. Eigen-Vector used approximately 112 M-flops (Minimum-Variance 47) for no subarrays, this decreased to roughly 36 M-flops (Minimum-Variance 16) for 29 subarrays (Tab. 7.2).

For Eigen-Vector (Fig. 7.2(b)), another trend for the different approaches for estimating $\hat{\mathbf{R}}$ became clear than for Minimum-Variance (Fig. 7.2(a)). When Eigen-Vector used less than 11 subarrays, the two methods applying noise removal were faster than the same methods without noise removal. Forward/backward smoothing together with noise removal was not only faster than plain forward/backward smoothing, it was actually faster than plain forward-only smoothing as well. This observation is due to the estimation of the eigenvectors. The iterative procedure of computing the eigenvectors can be expected to converge faster if a good estimate of the noise is removed from $\hat{\mathbf{R}}$ before the eigenvectors are computed.

Conclusion Eigen-Vector achieved the best results for the same amount of subarrays and smoothing approach as Minimum-Variance (Tab. 7.8).

7.1.3 MUSIC

The MUSIC algorithm is basically the same as the Eigen-Vector algorithm, but MUSIC assumes all noise eigenvalues to equal each other. This difference lead to other results for MUSIC than Eigen-Vector.

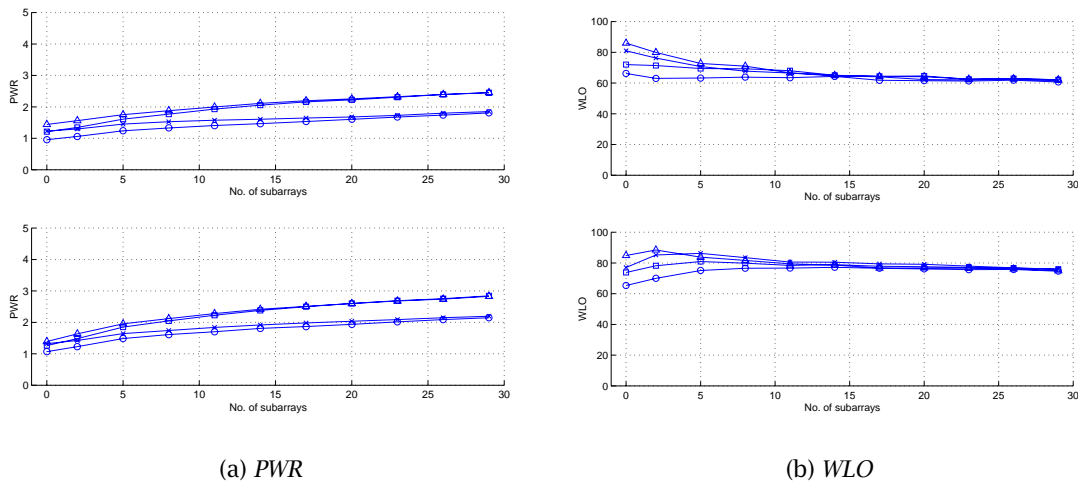


Figure 7.3: *PWR* and *WLO* results for MUSIC. Upper part, normal incidence signals. Lower part, obliquely arriving signals. Plain forward-only smoothing \circ . Forward-only smoothing with noise removal \times . Plain forward/backward smoothing \square . Forward/backward smoothing with noise removal \triangle .

Peak-Width Ratio MUSIC's *PWR* showed a steady increase along with increasing the number of subarrays (Fig. 7.3(a)). Plain forward/backward smoothing and forward/backward smoothing with noise removal gave a higher *PWR* than forward-only smoothing for all subarray sizes (Tab. 7.3). For 14 subarrays, or less, removing the estimated noise led to a higher *PWR* than no noise removal.

The difference between normal incidence and obliquely incoming signals was small, at least for few subarrays. When many subarrays were used, the obliquely incoming signals resulted in a *PWR* roughly 0.4 higher than for normal incidence signals.

The results for *PWR*(*M*) were somewhat lower for all subarray sizes than the *PWR* results, but showed the same trend.

The observed increase in *PWR* is quite opposite from what would be the case for the Fourier method, but can be explained by the fact that MUSIC's resolution increases with the amount of available data. For the MUSIC algorithm, use of subarrays seem to simulate an increase in available data.

Width-Location Overlap The subarray sizes that gave a high *PWR* for MUSIC were not the same that gave the highest *WLO* (Fig. 7.3(b)).

The highest *WLO*, for both normal and oblique incidence signals, was obtained for forward/backward smoothing with removal of the noise estimate (Tab. 7.3). The *WLO* was also higher for forward-only smoothing and noise removal than for plain forward/backward smoothing (Fig. 7.3(b)).

For the signals of normal incidence, forward-only smoothing and forward/backward

smoothing, both with noise removal, gave the highest *WLO* for no subarrays. The *WLO* dropped as the number of subarrays was increased (Fig. 7.3(b)).

Plain forward-only smoothing and forward/backward smoothing gave a lower *WLO* than the noise removal approaches. The difference in *WLO* as more subarrays were applied was small.

The *WLO* results for obliquely incoming signals were slightly different. Not only was the *WLO* higher for all subarray sizes, the trend for different amounts of subarrays also changed. For normal incidence signals, a drop off in *WLO* was observed from 0 until 5 subarrays, this changed to an increase of *WLO* for the obliquely arriving signals!

MUSIC's *WLO(M)* was not entirely similar to the *WLO*.

For the normal incidence signals, only small differences in *WLO(M)* could be seen for forward/backward smoothing together with noise removal as different subarray sizes were applied. Plain forward/backward smoothing now resulted in an increase of *WLO(M)* as more subarrays were applied. Forward-only smoothing and removal of the estimated noise resulted in a drop of the *WLO(M)* for around 5 subarrays. Plain forward-only smoothing had a drop in *WLO(M)* for 2 subarrays.

The *WLO(M)* trend for obliquely arriving signals was almost identical, although somewhat lower for all subarray sizes.

Resolution (TCR) MUSIC resolved more closely spaced sources by applying forward/backward smoothing than forward-only smoothing, both for the normal incidence and obliquely arriving signals. Noise removal in addition to forward/backward smoothing or forward-only smoothing, resulted in more resolved close sources than what was achieved by plain forward/backward smoothing or plain forward-only smoothing - even though the difference was small (Tab. 7.3).

For the signals of normal incidence, applying forward/backward smoothing instead of forward-only smoothing gave approximately 0.2 to 0.3 more resolved sources per snapshot. The ability of resolving closely spaced sources was highest for between 5 and 20 subarrays, roughly 0.4 resolved sources per snapshot.

For obliquely incoming signals, the difference between forward-only smoothing and forward/backward smoothing was smaller. The amount of resolved closely spaced sources was highest for forward/backward smoothing with 2 subarrays and noise removal. This approach resulted in only 0.25 resolved sources per snapshot.

WLO of closely spaced sources The *WLO* of the closely spaced sources was highest when forward/backward smoothing (with or without noise removal) was applied. For 5 or less subarrays, it seemed advantageous to remove the estimated noise. When more than 5 subarrays were used, it appeared to be almost no difference between forward/backward smoothing with and without removing the estimated noise (Tab. 7.3).

MUSIC			
Measure	Resulted in	for $K \in$	when $\hat{\mathbf{R}}$ was estimated as
PWR	$1.4 \rightarrow 2.6$	$[0, 29]$	$\hat{\mathbf{R}}_{fbs}^K - \hat{\mathbf{N}}$
WLO	$> 76\%$	$[0, 8]$	$\hat{\mathbf{R}}_{fbs}^K - \hat{\mathbf{N}}$
TCR	≈ 0.3	$[2, 14]$	$\hat{\mathbf{R}}_{fbs}^K - \hat{\mathbf{N}}$
WLO_{TCR}	$> 105\%$	$[0, 2] \cup [14, 29]$	$\hat{\mathbf{R}}_{fbs}^K - \hat{\mathbf{N}}$
M-flops	$37 \rightarrow 24$	$[20, 29]$	$\hat{\mathbf{R}}_{fs}^K$ (plain)

Table 7.3: The best results for the MUSIC algorithm, based on the average between signals of normal incidence and oblique angles.

Computation Time The differences between MUSIC and Eigen-Vector led to a less computational burden for MUSIC. Again, it was seen that applying noise removal resulted in faster computations of the spectra than not removing the estimated noise.

Forward/backward smoothing needed 97 M-flops for no subarrays, whereas applying noise removal in addition led to a use of 82 M-flops. For the forward-only smoothing approach the numbers were 86 M-flops for no noise removal, and 74 M-flops when the estimated noise was subtracted. The difference between not removing the estimated noise and removing it became less evident when increasing the amount of subarrays. When using more than 14-17 subarrays, it was actually faster not to remove the estimated noise (but the difference was now only about 2 M-flops). By applying 29 subarrays, the use of M-flops decreased to roughly 26 - depending on what strategy for estimating $\hat{\mathbf{R}}$ that was used (Tab. 7.3).

Conclusion From the quantitative observations about MUSIC, the PWR was found to increase as more subarrays were applied. This was not the case for the WLO , which decreased when more than two subarrays were used for smoothing. In either case, forward/backward smoothing improved the results compared to forward-only smoothing - further, removing the noise gave better results than not removing it.

MUSIC resolved more close sources for forward/backward smoothing with noise removal with between 2 and 14 subarrays, the WLO for these sources was highest when 0, 2 or 14 to 29 subarrays were applied.

As I regard the WLO to be of greater importance than the PWR , few subarrays should be used for MUSIC in my opinion. To both get a high TCR and WLO , 2 subarrays should be applied. This might lead to a slightly lower WLO of the closely spaced sources, and surely more M-flops will be used. Still, $\hat{\mathbf{R}}$ should be estimated by forward/backward smoothing with two subarrays and noise removal (Tab. 7.8).

7.1.4 Minimum-Norm

Peak-Width Ratio Minimum-Norm accomplished a quite high PWR for all subarray sizes (Fig. 7.4(a)). A somewhat higher PWR was achieved when the estimated noise was not removed (Tab. 7.4).

For less than 20 subarrays, plain forward-only smoothing and plain forward/backward smoothing performed equally well. When more than 20 subarrays were used, the PWR of the forward/backward smoothing approach started to drop, while it stayed the same for forward-only smoothing.

When the estimated noise was removed, it was preferable to use forward/backward smoothing and not forward-only smoothing, both for the signals of normal and oblique incidence. Noise removal led to roughly 0.5 lower PWR for all subarray sizes than what was achieved for the two methods above.

The $PWR(M)$ measures were almost identical to the PWR measures.

Width-Location Overlap For the WLO , quite unexpected results appeared. The two methods for estimating $\hat{\mathbf{R}}$ that removed the estimated noise were *much* better than those not applying the noise removal (Fig. 7.4(b)).

Forward-only smoothing and forward/backward smoothing with noise removal gave a WLO of approximately 40% for the normal incidence signals and 50-60% for the signals arriving from oblique angles, for all subarrays (Fig. 7.4(b)). Plain forward-only smoothing and forward/backward smoothing resulted in a WLO of not more

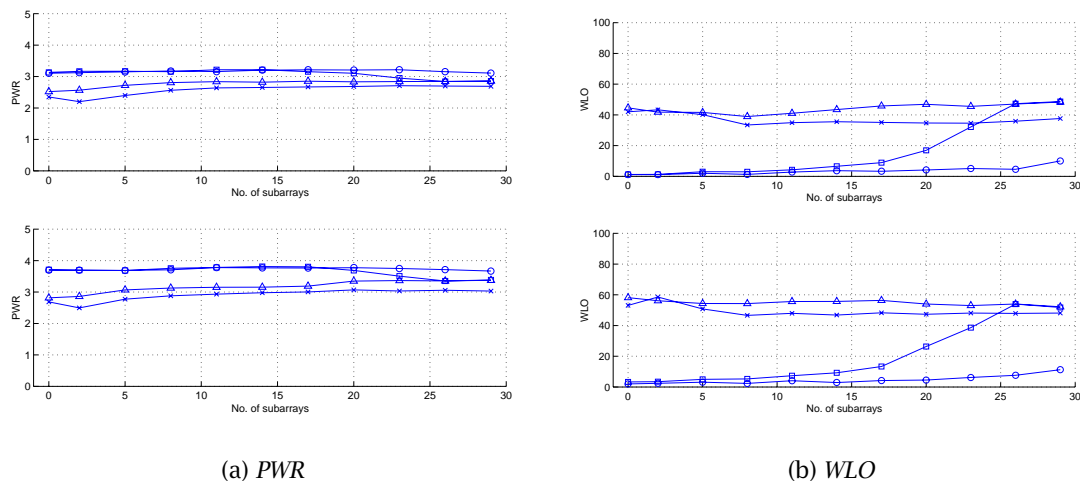


Figure 7.4: PWR and WLO results for Minimum-Norm. Upper part, normal incidence signals. Lower part, obliquely arriving signals. Plain forward-only smoothing \circ . Forward-only smoothing with noise removal \times . Plain forward/backward smoothing \square . Forward/backward smoothing with noise removal \triangle .

than 10-20% for 0 to 20 subarrays. Smoothing with more than 20 subarrays led to an increase in *WLO* for the plain forward/backward smoothing approach, whereas no significant increase was observed for the plain forward-only smoothing approach. For 26 and 29 subarrays, plain forward/backward smoothing performed similar to forward-only smoothing and forward/backward smoothing with noise removal.

Thus, removing the estimated noise can significantly improve the *WLO* for Minimum-Norm, at least when few subarrays are used (Tab. 7.4).

For Minimum-Norm, there was a noticeable difference between *WLO* and *WLO(M)*. The *WLO(M)* was higher than the *WLO* (Tab. 7.4). The *WLO(M)* of forward-only smoothing and forward/backward smoothing, both with noise removal, raised as the amount of subarrays was increased. Whereas the *WLO* was steady for forward-only smoothing and forward/backward smoothing with noise removal. The *WLO(M)* started with values roughly equal to the *WLO* (Fig. 7.4(b)), both for signals of normal and oblique incidence.

Resolution (TCR) Minimum-Norm's ability of resolving closely spaced sources was quite high, but depended upon how $\hat{\mathbf{R}}$ was estimated. It is clear that Minimum-Norm resolved more than twice as many close sources when the estimated noise was removed than it did without the noise removal (Tab. 7.4).

For noise removal, even though forward/backward smoothing always resolved more close sources than forward-only smoothing, forward-only smoothing and forward/backward smoothing showed the same trend (between 2 to 5 and 23 to 26 subarrays were preferable).

Plain forward/backward smoothing gave poor results when applying less than 23 subarrays. Plain forward-only smoothing always performed poorly, it resolved below 0.1 sources per snapshot.

The same trends appeared for both normal and oblique incidence signals, even though the results for obliquely arriving signals were lower for all subarray sizes.

WLO of closely spaced sources The *WLO* of the closely spaced sources was highest for forward/backward smoothing in combination with noise removal (Tab. 7.4). The *WLO* was only below 100% when less than 5 subarrays were applied for signals of oblique incidence. In any case the *WLO* increased as more subarrays were used.

Plain forward/backward smoothing only achieved the same *WLO* as obtained by noise removal when 26 or 29 subarrays were applied.

Plain forward-only smoothing was never capable of achieving a *WLO* of more than roughly 25%. For less than 17 subarrays, forward-only smoothing and noise removal gave better results than plain forward/backward smoothing.

Minimum-Norm			
Measure	Resulted in	for $K \in$	when $\hat{\mathbf{R}}$ was estimated as
PWR	> 3.4	$[0, 29]$	$\hat{\mathbf{R}}_{fs}^K$ (plain)
WLO	$50\% \pm 2$	$[0, 29]$	$\hat{\mathbf{R}}_{fbs}^K - \hat{\mathbf{N}}$
$WLO(M)$	$> 76\%$	$[17, 29]$	$\hat{\mathbf{R}}_{fbs}^K - \hat{\mathbf{N}}$
TCR	> 0.4	$[5, 26]$	$\hat{\mathbf{R}}_{fbs}^K - \hat{\mathbf{N}}$
WLO_{TCR}	$> 110\%$	$[14, 29]$	$\hat{\mathbf{R}}_{fbs}^K - \hat{\mathbf{N}}$
M-flops	$68.5 \rightarrow 46.5$	$[20, 29]$	$\hat{\mathbf{R}}_{fs}^K$ (plain)

Table 7.4: The best results for the Minimum-Norm algorithm, based on the average between signals of normal incidence and oblique angles.

Computation Time As for the other methods that compute the eigenvectors, noise removal was advantageous for Minimum-Norm when it came to the amount of M-flops. Forward-only smoothing was faster than forward/backward smoothing.

For no subarrays, plain forward/backward smoothing required 151 M-flops, removing the noise restricted the use of M-flops to 137. For 29 subarrays both methods used 51 M-flops.

Plain forward-only smoothing used 141 M-flops for no subarrays, this decreased to 128 when the noise estimate was removed. For 29 subarrays, not removing the noise was slightly less computationally demanding than removing it, 47 M-flops were used (Tab. 7.4). Removal of the noise needed 48 M-flops.

Conclusion Minimum-Norm always had a high PWR . The WLO laid steady around 50% for all subarray choices, but the $WLO(M)$ behaved differently.

If the number of sources, M , was chosen correctly, the $WLO(M)$ should be high. Therefore, to be able to figure out how many subarrays to use for Minimum-Norm, the $WLO(M)$ should be regarded instead of the WLO . This means that forward/backward smoothing with 17 or more subarrays should be used for Minimum-Norm. Also, the estimated noise should be removed to get the best results.

To make Minimum-Norm as fast as possible, 29 subarrays should be used. But, this leads to slightly fewer resolved sources, and a lower WLO for these sources.

Thus, to achieve the best results for the most important measures, *26 subarrays are suggested used for Minimum-Norm, together with noise removal and forward/backward smoothing.*

The reason for choosing 26 subarrays, is that with fewer subarrays Minimum-Norm was totally dependent upon noise removal. In cases where this noise is difficult to estimate, one can expect the WLO to behave as if the noise was not removed at all. Thus, it is safest to use 26 subarrays (Tab. 7.8).

7.1.5 Root-MUSIC

Peak-Width Ratio The *PWR* measure is not applicable for root-MUSIC. The reason is that the "peaks" of root-MUSIC are not real peaks, they are δ -pulses. However, the *PWR* can still be measured by defining every such δ -pulse to have a certain width. In this work, the width is defined to equal $h(0.5)$ – the angular increment size that is used for the spectral based methods.

For root-MUSIC, this leads to a *PWR* of 3.5 for the normal incidence signals, while for the obliquely arriving signals it equals 4.1. Thus, the Fourier method's resolution is better for normal incidence signals than for oblique. That is, the FWHM of the Fourier method equals approximately 1.75 degrees for signals of normal incidence, while for obliquely arriving signals the FWHM is roughly 2 degrees.

Width-Location Overlap Having defined the width of a peak for root-MUSIC, we are now able to find the *WLO*. If the width of the peaks in the root-MUSIC spectrum were defined as for δ -pulses, the *WLO* would simply tell us if the highest root-MUSIC peak was covered by the highest Fourier peak, or not. It would either be 0, or 100%. By defining the peak-width as the spatial area from $-h/2$ below the δ -pulse's θ value to $h/2$ above, one is capable of telling where the δ -pulse is placed compared to the peak of the Fourier method.

The *WLO* of root-MUSIC was not very high (Fig. 7.5(a)). At its highest, it was 28%. This occurred for signals of normal incidence, for obliquely arriving signals it did not exceed 20%.

The highest *WLO* was achieved by using forward/backward smoothing. Applying noise removal in addition led to a higher *WLO* when less than 14 subarrays were used, use of more subarrays gave similar results for forward/backward smoothing either noise removal was applied or not (Tab. 7.5).

For the normal incidence signals it was advantageous to use between 14 and 26 subarrays, while it for signals of oblique angles seemed better to apply between 8 and 20 subarrays.

The *WLO(M)* of root-MUSIC was much higher than the *WLO*, roughly twice as high (Fig. 7.5(b)). The difference in *WLO(M)* observed when applying noise removal or not was even larger for few subarrays than for the *WLO*. Forward/backward smoothing with noise removal always performed best (Tab. 7.5).

For signals of normal incidence, forward/backward smoothing and noise removal gave the highest *WLO(M)* for about 14 subarrays. Obliquely arriving signals had the highest *WLO(M)* for about 8 subarrays.

Resolution (TCR) Root-MUSIC's ability of resolving closely spaced sources was at its best for signals of normal incidence. The best results were achieved when noise was removed, and forward/backward smoothing applied (Tab. 7.5). Forward-only

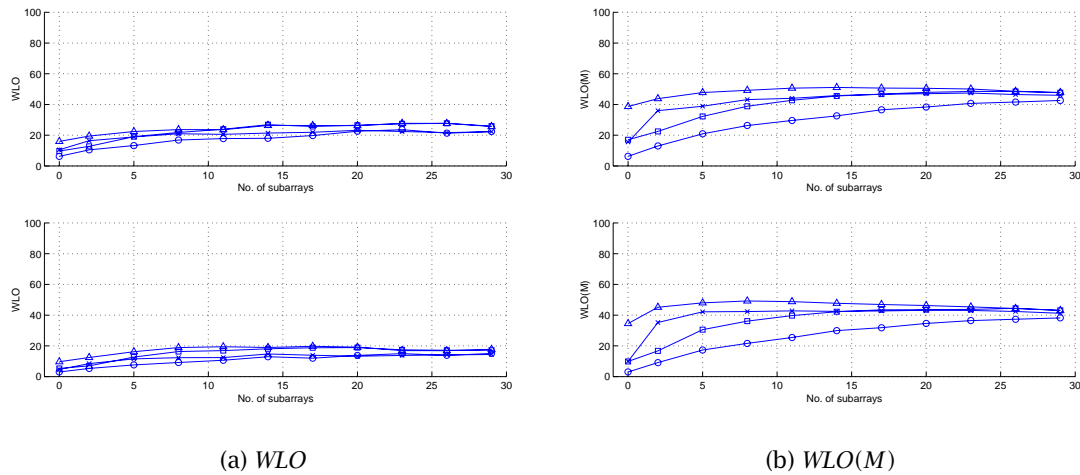


Figure 7.5: *WLO* and *WLO(M)* results for root-MUSIC. Upper part, normal incidence signals. Lower part, obliquely arriving signals. Plain forward-only smoothing \circ . Forward-only smoothing with noise removal \times . Plain forward/backward smoothing \square . Forward/backward smoothing with noise removal \triangle .

smoothing with noise removal showed the exact same trend, but resolved fewer sources per snapshot. When the estimated noise was not removed, both forward-only smoothing and forward/backward smoothing degraded.

For signals arriving from oblique angles, the trend for the different subarray sizes was somewhat different. Still, forward/backward smoothing with noise removal always performed best (Tab. 7.5).

WLO of closely spaced sources The *WLO* of the closely spaced sources was not above 100% for any way of estimating $\hat{\mathbf{R}}$. For few subarrays it was advantageous to apply noise removal. The differences between noise removal or no noise removal became smaller as more subarrays were applied (Tab. 7.5).

For the normal incidence signals, forward/backward smoothing together with noise removal performed better than plain forward/backward smoothing when less than 17 subarrays were applied. With more subarrays they performed roughly equal, and obtained *WLO* values close to 100%. Forward/backward smoothing and noise removal always performed better than forward-only smoothing. Forward-only smoothing with noise removal performed better than plain forward-only smoothing for less than 26 subarrays.

Obliquely arriving signals resulted in *WLO* values roughly half as high as for the normal incidence signals. For less than 20 subarrays, forward/backward smoothing with noise removal gave higher *WLO* values for the close sources than all other ways of estimating $\hat{\mathbf{R}}$. Plain forward/backward smoothing gave equally high *WLO* values as forward/backward smoothing with noise removal, when more than 20 subarrays

were applied. For less than 26 subarrays, forward-only smoothing with noise removal performed better than plain forward-only smoothing, as for signals of normal incidence.

root-MUSIC (5.23)			
Measure	Resulted in	for $K \in$	when $\hat{\mathbf{R}}$ was estimated as
WLO	$21\% \pm 2$	$[5, 29]$	$\hat{\mathbf{R}}_{fbs}^K - \hat{\mathbf{N}}$
$WLO(M)$	$48\% \pm 1.4$	$[5, 26]$	$\hat{\mathbf{R}}_{fbs}^K - \hat{\mathbf{N}}$
TCR	> 0.45	$[2, 14]$	$\hat{\mathbf{R}}_{fbs}^K - \hat{\mathbf{N}}$
WLO_{TCR}	$> 75\%$	$5 \cup [14, 23]$	$\hat{\mathbf{R}}_{fbs}^K - \hat{\mathbf{N}}$
M-flops	$134 \rightarrow 83$	$[20, 29]$	$\hat{\mathbf{R}}_{fs}^K$ (plain)

Table 7.5: The best results for the root-MUSIC algorithm, based on the average between signals of normal incidence and oblique angles.

Computation Time The polynomial rooting in root-MUSIC was very time demanding. When no subarrays were applied, more than 300 M-flops were needed to compute the spectrum no matter how $\hat{\mathbf{R}}$ was estimated. By increasing the amount of subarrays to 29, this was reduced to 88 M-flops for the two forward/backward smoothing approaches. The forward-only smoothing approaches needed 83 M-flops for no noise removal and 85 M-flops when the noise was removed (Tab. 7.5).

Conclusion The *PWR* of root-MUSIC is of no interest. Only the other measures are important. It is clear that *forward/backward smoothing and noise removal are preferable* also for root-MUSIC. The $WLO(M)$ was higher than the WLO , as it was for Minimum-Norm. This measure was highest for 11 subarrays, but increasing the amount of subarrays only led to a small drop off. However, increasing the amount of subarrays also led to fewer resolved close sources, but the WLO of these sources was higher for between 14 and 23 subarrays. Since root-MUSIC is very time consuming, one must use many subarrays. Actually, it uses so many M-flops that one should *use 29 subarrays* and accept a degradation of the performance (Tab. 7.8).

7.1.6 Alternative root-MUSIC

In this section, the results for root-MUSIC defined by (5.25) are given.

Peak-Width Ratio gives no meaning for root-MUSIC.

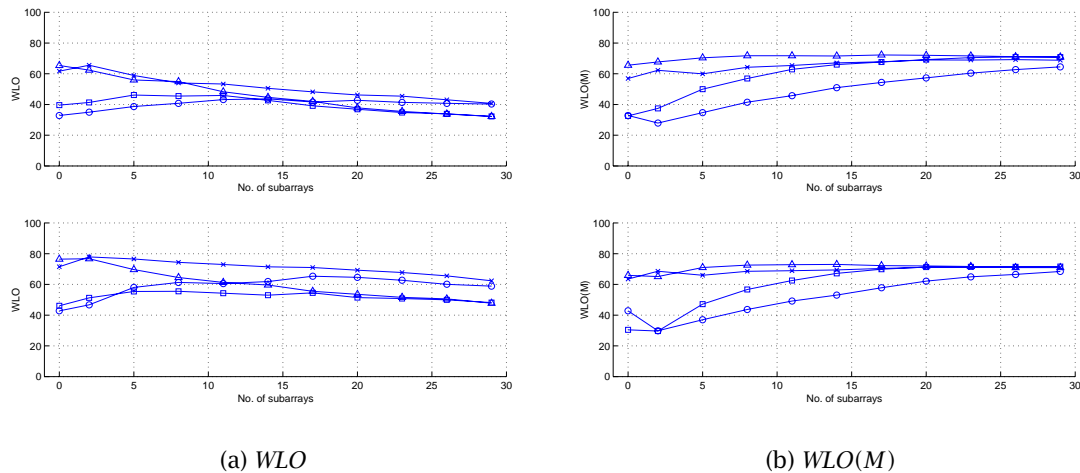


Figure 7.6: *WLO* and *WLO(M)* results for root-MUSIC (5.25). Upper part, normal incidence signals. Lower part, obliquely arriving signals. Plain forward-only smoothing \circ . Forward-only smoothing with noise removal \times . Plain forward/backward smoothing \square . Forward/backward smoothing with noise removal \triangle .

Width-Location Overlap While the *WLO* of root-MUSIC was not above 28% for the former root-MUSIC method, it was almost as high as 80% for the signals of oblique incidence when computed via (5.25).

The best *WLO* results were obtained by forward-only smoothing with noise removal (Tab. 7.6), only “beaten” by forward/backward smoothing with noise removal, when no subarrays were applied. Both forward-only smoothing and forward/backward smoothing gave higher *WLO* values for noise removal than the plain approaches, when few subarrays were applied. As more subarrays were applied, both methods applying noise removal degraded in performance (Fig. 7.6(a)). Plain forward-only smoothing and forward/backward smoothing performed better as more subarrays were applied, but started with much lower *WLO* values (Fig. 7.6(a)). For 17, or more, subarrays, plain forward/backward smoothing and forward/backward smoothing with noise removal performed equal. Forward-only smoothing always had an edge over plain forward-only smoothing.

The *WLO(M)* for root-MUSIC computed by (5.25), differed from the *WLO* by the fact that (Fig. 7.6(b)): (1) The *WLO(M)* of the two noise subtraction approaches did not drop significantly by applying more subarrays. (2) The plain forward-only smoothing and plain forward/backward smoothing approaches increased their *WLO(M)* to the same level as those removing the estimated noise when many subarrays were used. (3) The *WLO(M)* values, for high numbers of subarrays, were higher.

Resolution (TCR) The trends were pretty much as for the former root-MUSIC approach, although somewhat more slowly changing for the different subarray sizes. More close sources were resolved by forward/backward smoothing and noise removal than by the other smoothing approaches (Tab. 7.6).

root-MUSIC (5.25)			
Measure	Resulted in	for $K \in$	when $\hat{\mathbf{R}}$ was estimated as
WLO	$> 64\%$	$[0, 8]$	$\hat{\mathbf{R}}_{fs}^K - \hat{\mathbf{N}}$
TCR	> 0.45	$[2, 11]$	$\hat{\mathbf{R}}_{fbs}^K - \hat{\mathbf{N}}$
WLO_{TCR}	$> 105\%$	$[5.8] \cup 20$	$\hat{\mathbf{R}}_{fbs}^K - \hat{\mathbf{N}}$
M-flops	$34 \rightarrow 20$	$[20, 29]$	$\hat{\mathbf{R}}_{fs}^K$ (plain)

Table 7.6: The best results for the alternative root-MUSIC algorithm, based on the average between signals of normal incidence and oblique angles.

WLO of closely spaced sources The trend for the WLO of the closely spaced but resolved sources was quite equal to what was obtained with the previous root-MUSIC method, but the results were now higher all over (Tab. 7.6). A larger difference between forward/backward smoothing and forward-only smoothing, both with noise removal, was now seen – forward/backward smoothing achieved much higher WLO values.

Computation Time This new approach for root-MUSIC was very time-efficient compared to the original root-MUSIC. The most time consuming method for estimating $\hat{\mathbf{R}}$ was now plain forward/backward smoothing, which used 98 M-flops when no subarrays were applied. The fastest method was forward-only smoothing and noise removal, it used 75 M-flops. When 29 subarrays were used, all approaches resulted in approximately the same need of M-flops – ranging from 20 to 24, for plain forward-only smoothing (Tab. 7.6), forward-only smoothing with noise removal, plain forward/backward smoothing, and slowest forward/backward smoothing with noise removal.

Conclusion To achieve the highest WLO , few subarrays should be used. This, again, led to roughly equal WLO whether forward-only smoothing or forward/backward smoothing was used – when noise was subtracted. The $WLO(M)$ was higher for forward/backward smoothing than for forward-only smoothing, both applying noise removal.

For few subarrays, roughly twice as many closely spaced sources were resolved by the forward/backward smoothing and noise removal approach than by the other approaches. Further, the WLO of these close sources was higher for forward/backward smoothing with noise removal.

Consequently, *forward/backward smoothing with noise removal should be applied to get the best results all over, smoothing with 2 subarrays* (Tab. 7.8).

7.1.7 ESPRIT

Peak-Width Ratio The *PWR* is computed in the exact same way for ESPRIT as for root-MUSIC; thus, the *PWR* measure is not applicable.

Width-Location Overlap The *WLO* of ESPRIT was always higher for forward/backward smoothing than for forward-only smoothing (Tab. 7.7). Both for signals of normal incidence and obliquely arriving signals. There was virtually no difference in *WLO* whether the estimated noise was removed or not.

Forward/backward smoothing resulted in a prominent increase of *WLO* as more subarrays were applied, but the gain in *WLO* was small when more than 5 subarrays were used (Fig. 7.7(a)).

When applying forward-only smoothing, another trend appeared. As for forward/backward smoothing, the increase in *WLO* was prominent from 0 to about 5 subarrays. When more subarrays were applied, a steady, though small, decrease of *WLO* was observed.

Almost similar results were observed for the *WLO(M)* measure, only that the differences between the various amounts of subarrays were somewhat smoother. In addition, the *WLO(M)* achieved by forward-only smoothing did not decrease for more than 5 subarrays, as the *WLO* did.

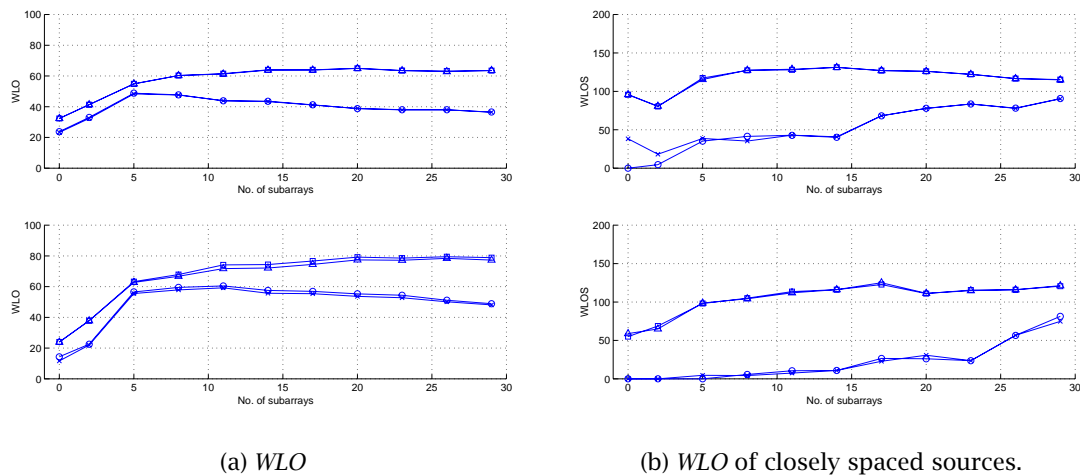


Figure 7.7: *WLO* results for ESPRIT. Upper part, normal incidence signals. Lower part, obliquely arriving signals. Plain forward-only smoothing \circ . Forward-only smoothing with noise removal \times . Plain forward/backward smoothing \square . Forward/backward smoothing with noise removal \triangle .

Resolution (TCR) ESPRIT's ability to resolve closely spaced sources was rather steady for different amounts of subarrays. Applying forward/backward smoothing

resolved far more closely spaced sources than what was achieved by forward-only smoothing, even though forward-only smoothing closed in on forward/backward smoothing when more than 20 subarrays were used. No difference was observed between noise removal and not removing the estimated noise (Tab. 7.7).

For signals of normal incidence, forward/backward smoothing and more than 5 subarrays ensured that between 0.5 and 0.6 close sources were resolved per snapshot – slowly increasing as more subarrays were used. Forward-only smoothing resolved few sources (0.1) when less than 14 subarrays were applied. As more subarrays were applied, a linear increase was observed – resulting in 0.4 resolved close sources for 29 subarrays.

The signals arriving from oblique angles gave fewer resolved close sources. Forward-only smoothing never obtained more than 0.1 resolved close sources per snapshot for these signals. Applying forward/backward smoothing gave somewhat better results, more than 0.2 close sources were resolved per snapshot when 2 or more subarrays were applied.

WLO of closely spaced sources The *WLO* of the closely spaced sources was mostly above 100% for forward/backward smoothing (Tab. 7.7). Forward-only smoothing gave less attractive results than forward/backward smoothing (Fig. 7.7(b)).

For normal incidence signals, the *WLO* for the forward/backward smoothing approach was only below 100% for 2, or less, subarrays. Forward-only smoothing always resulted in a *WLO* below 100%. This was also observed for the signals arriving from oblique angles (Fig. 7.7(b)).

Computation Time For ESPRIT, there was virtually no difference in speed whether noise was removed or not. Applying forward-only smoothing was faster than forward/backward smoothing, it used roughly 5.5 M-flops for no subarrays and 4.4 M-flops for 29 subarrays (Tab. 7.7). Forward/backward smoothing used 13.8 M-flops when no subarrays were applied, and 6.9 M-flops when 29 subarrays were used.

ESPRIT			
Measure	Resulted in	for $K \in$	when $\hat{\mathbf{R}}$ was estimated as
WLO	> 65%	[11, 29]	$\hat{\mathbf{R}}_{fbs}^K$ (both)
TCR	> 0.37	[5, 29]	$\hat{\mathbf{R}}_{fbs}^K$ (both)
WLO_{TCR}	> 115%	[8, 23]	$\hat{\mathbf{R}}_{fbs}^K$ (both)
M-flops	4.8 → 4.4	[20, 29]	$\hat{\mathbf{R}}_{fs}^K$ (both)

Table 7.7: The best results for the ESPRIT algorithm, based on the average between signals of normal incidence and oblique angles.

The reason for ESPRIT using fewer M-flops when not removing the noise, is that in this thesis ESPRIT is implemented such that it only computes the eigenvectors belonging to the 6 largest eigenvalues. Only the M eigenvectors corresponding to the signal subspace are needed in the algorithm. Noise will be found in the eigenvectors corresponding to the $L-M$ smallest eigenvalues, assuming M is chosen correctly. Thus, although all eigenvalues are affected by noise, the M first eigenvectors ideally do not contain any noise. Therefore, one can not expect a faster convergence for these eigenvectors, or an improvement of the WLO , by removing the noise. The computational burden of removing the estimated noise is larger than what is gained by a faster convergence for $6 - M$ of the six largest eigenvectors.

Conclusion As for root-MUSIC, the PWR is not applicable for ESPRIT.

ESPRIT performed equal whether or not the estimated noise was removed, but applying forward/backward smoothing instead of forward-only smoothing improved the performance of all measures to a great extent.

The WLO of ESPRIT was highest for 20 subarrays, but was only a few percentages lower for between 14 and 29 subarrays.

Roughly the same amount of closely spaced sources were resolved for between 5 and 29 subarrays. The WLO of the close sources was best when 17 subarrays were applied, the difference of this WLO for different subarray sizes was larger than for the WLO mentioned above.

ESPRIT was a fast method, no matter how $\hat{\mathbf{R}}$ was estimated, but it was advantageous to use many subarrays – and not remove the estimated noise.

Thus, *forward/backward smoothing with 17 subarrays is suggested* for ESPRIT (Tab. 7.8).

7.2 General Observations

There is a difference in observed Peak-Width Ratio between the obliquely arriving and normal incidence signals. The PWR is larger for signals arriving from oblique angles. This is mainly due to the fact that the Fourier method's resolution (peak-width) changes with angle, its mainlobe will be broader the further out to the side we look. Since the PWR is here the ratio of Fourier's peak-width to a high-resolution methods peak-width, the PWR will become larger for oblique angles. This actually means that the peak-width of the high-resolution methods is less sensitive to what angle the signal is impinging from, than the peak-width of the Fourier method is.

The results of MUSIC's WLO coincide well with Rao and Hari's theory [24]. Thus, the WLO is indeed a meaningful measure for all methods. Although, according to Rao and Hari [23], the WLO of ESPRIT should decrease when more than between 10 and 27 subarrays were used, both for the forward-only smoothing approach and the

forward/backward smoothing approach. This trend was clear solely for the forward-only smoothing approach in these experiments.

The *WLO* was generally higher for signals coming from oblique angles than for normal incidence signals. This was observed for all methods, except root-MUSIC computed via (5.23). The higher *WLO* is due to the fact that the peaks of the high-resolution methods are narrower, relative to the Fourier peaks, for signals arriving from oblique angles (see comment about root-MUSIC's *PWR* earlier). Hence, more of them will be covered by the broad Fourier peaks for oblique angles, *if their position relative to the midpoint of the Fourier peaks are not different for normal incidence and oblique angles*. The results indicate that this is the case!

Minimum-Norm (applying many subarrays), root-MUSIC (5.23) (all subarrays) and root-MUSIC (5.25) (applying many subarrays) gave a much higher *WLO(M)* than *WLO*. This indicates that they choose the highest peak somewhat randomly for the respective amounts of subarrays. The peaks appear to be roughly covered by those of the Fourier spectrum, but the highest peak does not match the highest peak of the Fourier method.

For flat sea floors, the transmit pulse covers a larger area of the bottom directly under the vessel than it does to the sides (Fig. 6.1). For simplicity, only flat sea floors are discussed here. For other sea floors, the pulse may cover larger areas of the bottom to the sides (eg the lower right corner of Fig. 6.1). The peaks in the spectrum should be narrower for signals that come from directions where the pulse covers a small area at one time instant. It appears that the Fourier method is unable to exploit this fact, while the high-resolution methods, to some degree, do.

The above observation about the pulse-coverage also leads to a higher ability of resolving closely spaced sources for signals of normal incidence. The reason is that for obliquely incoming signals the pulse has only covered a small area of the bottom. Therefore, any closely spaced sources insonified by the pulse to the far side are *really close* (Fig. 6.1). Thus, they are not as easy to resolve. Since this is observed for the *TCR*, the *TCR* measure of the resolution is indeed meaningful – although somewhat dangerous.

Note that since only raw data were used in the evaluation of the methods, only assumptions could be made regarding closely spaced sources. There was no exact knowledge of where closely spaced sources were present. Sources were defined as closely spaced when the locations of the peaks were less than 2.5 degrees apart. It was simply assumed that when two closely spaced peaks were observed in the spectrum, these two peaks were closely spaced sources that were resolved. This means that the Fourier method may also have resolved some of these close sources. The sources that were resolved by the Fourier method led to wrong *WLO* measures for the other methods. Thus, the *WLO* of the closely spaced sources is perhaps somewhat lower than it actually should be. A definition demanding less spacing would lead to few resolved closely spaced sources, and consequently less secure measures – the standard deviation of the measures would be too high. Thus, this approach was applied even though it may have lead to some errors of the *WLO* measure for the closely spaced sources.

The noise removal gave better results for few subarrays than for many, this may be due to the drastic smoothing effect of using many subarrays. Such “heavy” smoothing flattens the noise quite well, consequently - removing the estimated noise is of little relevance when many subarrays are used.

The execution time of ESPRIT and root-MUSIC would not be affected by changing the angular increments, for all other methods it would. Refining the angular search would lead to a greater computational burden for the spectral based methods, while a coarser angular search would speed them up. In any event, the time needed to compute the subspaces would not change. Unfortunately, computation of the subspaces is the greatest computational burden for the subspace based high-resolution methods, while for Minimum-Variance, it is computing the inverse of $\hat{\mathbf{R}}$.

7.3 Comparing the Best Results

In section 7.1, measures for each of the high-resolution methods were given. A conclusion of how to estimate the covariance matrix, $\hat{\mathbf{R}}$, for each method was drawn. That is, *the parameter choices for estimation of $\hat{\mathbf{R}}$ that gave the best results on an overall basis were found for each method.* In this section, the results of the methods are compared for these specific parameter combinations in table 7.8.

From table 7.8, we can see that Minimum-Variance and Eigen-Vector perform almost equal. Eigen-Vector has a higher *PWR*, *TCR* and *WLO* of the close sources (WLO_{TCR}), but the *WLO* is lower and it uses more M-flops than Minimum-Variance. Thus, Minimum-Variance is so far regarded as most promising of the two.

MUSIC has a higher *WLO* than Minimum-Variance, all though the *PWR* is lower. Also the *TCR* of MUSIC is lower, but the *WLO* of the close sources is slightly higher. Even though MUSIC uses many M-flops, its *WLO* makes it a strong candidate for high-resolution beamforming. We will have to see in section 7.4 if the difference in *WLO* compared to Minimum-Variance is significant for the quality of the different spectra.

Minimum-Norm has a high *PWR*. The *WLO* is lower than for both Minimum-Variance and MUSIC, but the $WLO(M)$ is equally high as MUSIC's *WLO*. The *TCR* is high, and so is the *WLO* of the close sources. Minimum-Norm uses less M-flops than MUSIC, but more than Minimum-Variance. The spectra in section 7.4 will be conclusive for whether Minimum-Norm is better than Minimum-Variance and MUSIC.

The *WLO* and $WLO(M)$ of root-MUSIC computed via (5.23) excludes it as a candidate for bathymetric measures.

The alternative root-MUSIC, given by (5.25), has a higher *WLO* than the former root-MUSIC, comparable to that of Minimum-Variance. The *TCR* is among the highest observed, but the *WLO* of the close sources is not among the highest. Root-MUSIC uses approximately the same amount of M-flops as MUSIC, therefore we have to view the different spectra in section 7.4 to see if the *WLO* difference is of any importance.

Method	$\hat{\mathbf{R}}$	PWR		WLO		TCR		WLO _{TCR}		M-flops
		normal	oblique	normal	oblique	normal	oblique	normal	oblique	
Minimum-Variance	$\hat{\mathbf{R}}_{f_{bs}}^{20}$ (plain)	2.1	2.2	64	75	0.41	0.15	114	99	24.3
Eigen-Vector	$\hat{\mathbf{R}}_{f_{bs}}^{20}$ (plain)	2.3	2.6	59	70	0.42	0.20	115	113	52.1
MUSIC	$\hat{\mathbf{R}}_{f_{bs}}^2 - \hat{\mathbf{N}}$	1.6	1.6	80	89	0.36	0.25	120	101	79.5
Minimum-Norm	$\hat{\mathbf{R}}_{f_{bs}}^{26} - \hat{\mathbf{N}}$	2.9	3.4	47(83)	54(77)	0.55	0.27	123	110	57.9
root-MUSIC (5.23)	$\hat{\mathbf{R}}_{f_{bs}}^{29} - \hat{\mathbf{N}}$	x	x	26(48)	18(43)	0.29	0.10	100	46	88.1
root-MUSIC (5.25)	$\hat{\mathbf{R}}_{f_{bs}}^2 - \hat{\mathbf{N}}$	x	x	62	77	0.53	0.59	97	102	80.6
ESPRIT	$\hat{\mathbf{R}}_{f_{bs}}^{17}$ (plain)	x	x	64	77	0.52	0.25	127	123	9.4

Table 7.8: The results achieved for each method when $\hat{\mathbf{R}}$ is computed to give the best over all performance. WLO(M) is given in parentheses for methods that have a significant difference in WLO and WLO(M).

ESPRIT is the fastest method. It has one of the highest observed *TCRs*, and the *WLO* of the close sources is the highest achieved. The actual *WLO* is comparable to Minimum-Variance's and that of the alternative root-MUSIC. This makes ESPRIT one of the most promising methods, the effect of the *WLO* on the spectra will become clear when we compare the different methods in the following sections.

7.4 Qualitative Observations

We are now ready to compare the different methods visually. It is interesting to see if the results found in the previous section enable us to get good spectra, also after bottom detection. First a few spectra are shown (Sec. 7.4.1), followed by the results of the simple bottom detection algorithm (Sec. 7.4.2). A few comments about the visual results for the covariance matrices that gave the worst quantitative results are also given (Sec. 7.4.3).

7.4.1 Spectra

The spectra of three different pings are shown here. First a ping with a slowly varying bottom and low background noise is shown (Fig. 7.8/7.9). Next, we show a more complicated bottom with high background noise (Fig. 7.10/7.11). Finally, another complicated bottom with secondary echoes is shown (Fig. 7.12/7.13).

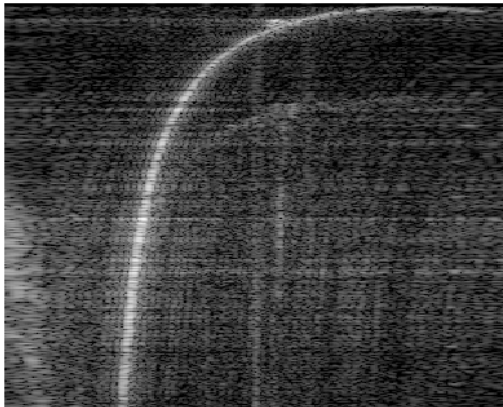
From the spectra in figures 7.8, 7.10 and 7.12, it is seen that the mainlobe of both Minimum-Variance, Eigen-Vector and MUSIC is narrower than that of the Fourier method. This was also found for the quantitative measures (Sec. 7.1). It can be seen from the different methods' *PWR* values. We can also see that the noise whitening effect of setting MUSIC's eigenvalues to 1 does indeed work, there is less background noise in the MUSIC spectra than the others.

Now, let us look at Minimum-Norm and the parametric methods root-MUSIC and ESPRIT (Fig. 7.9, 7.11 and 7.13).

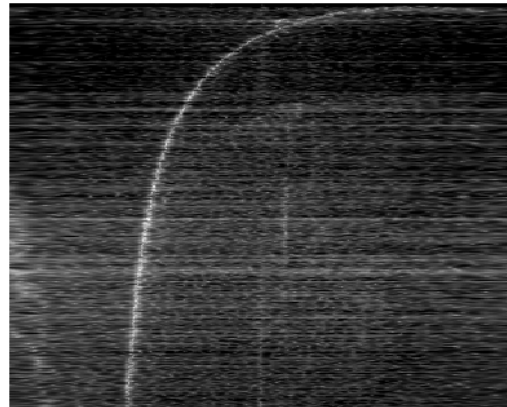
The Minimum-Norm spectra also have a narrower mainlobe than the Fourier spectra. As for MUSIC, there appears to have been a whitening of the background noise. It does not seem as if the low *WLO* of Minimum-Norm degrades the spectra severely. For root-MUSIC, computed by (5.23), the low *WLO* has a large effect. The "spectra" have very low quality. The "spectrum" quality of the alternative root-MUSIC, computed via (5.25), is much better, as anticipated by the *WLO* results (Tab. 7.8). The bottom curves are less "blurred" than for the former root-MUSIC. The "spectra" computed by ESPRIT appear to be even better. Less noise is seen than for root-MUSIC, and for figure 7.9 the bottom curve of ESPRIT is less "blurred".

It might seem as if MUSIC, Minimum-Norm, root-MUSIC and ESPRIT have more problems with their spectra for the complicated bottoms than the other methods. This is not clear! We might just be deceived by our eyes. Since the background noise is

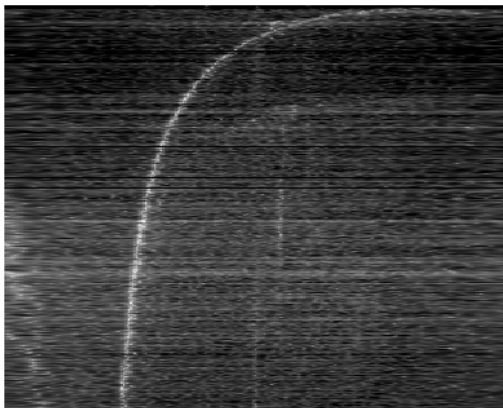
whitened (actually there is no background noise for the parametric methods), it is much easier to see the problematic parts of the spectra. These problematic parts seem less severe for the Fourier method, Minimum-Variance and Eigen-Vector, since they are buried in noise – it becomes difficult to distinguish visually between interesting peaks and noise. Thus, we need to see how the results of the bottom detection algorithm turn out.



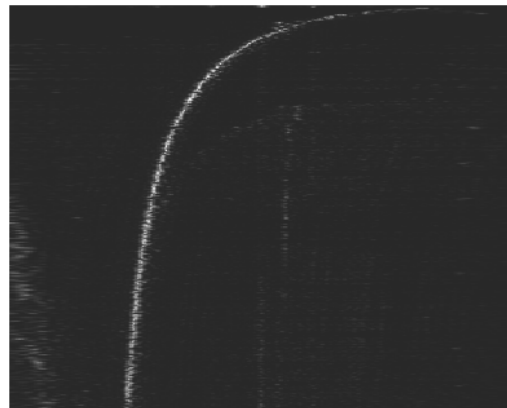
(a) Fourier Transform.



(b) Minimum-Variance.

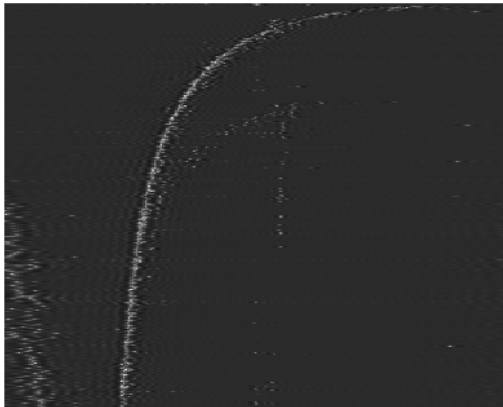


(c) Eigen-Vector.



(d) MUSIC.

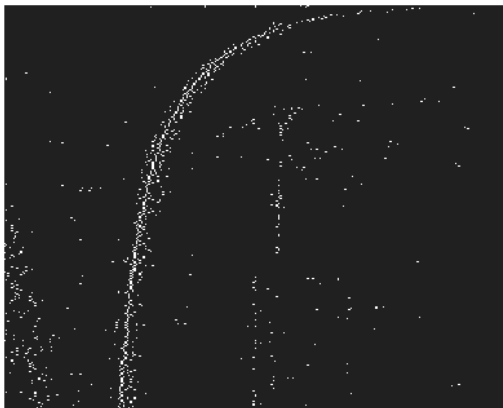
Figure 7.8: Spectra after a log transform. Ping shot the 17th April. Vertical axis: time (800 snapshots). Horizontal axis: -70 to 70 degrees (280 beams).



(a) Minimum-Norm after a log transform.



(b) root-MUSIC, computed via (5.23).

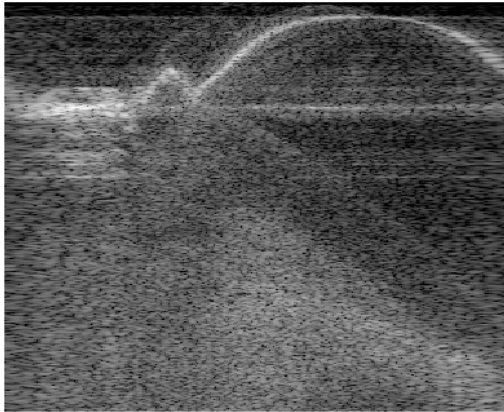


(c) root-MUSIC, computed via (5.25).

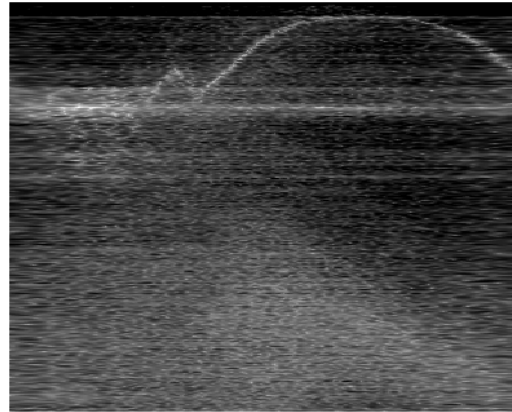


(d) ESPRIT.

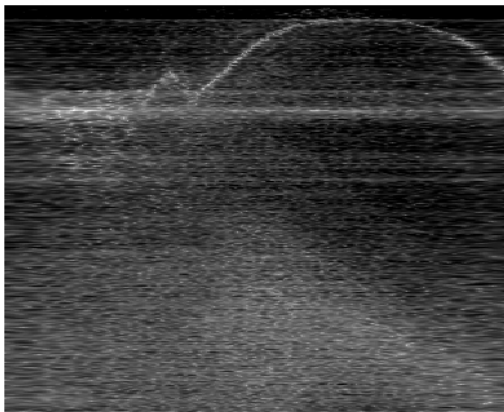
Figure 7.9: Spectra shown for the same ping as in Fig. 7.8. Vertical axis: time (800 snapshots). Horizontal axis: -70 to 70 degrees (280 beams).



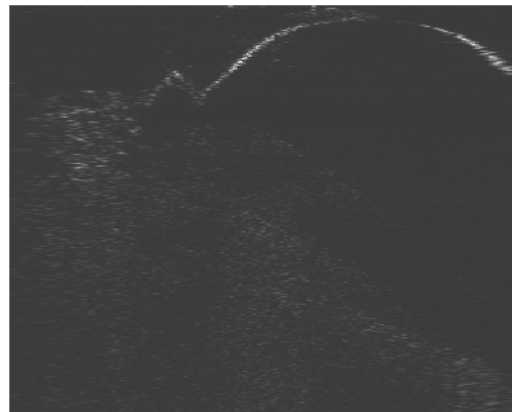
(a) Fourier Transform.



(b) Minimum-Variance.

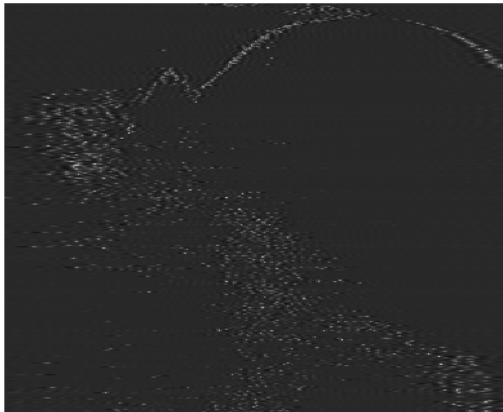


(c) Eigen-Vector.



(d) MUSIC.

Figure 7.10: Spectra after a log transform. Ping shot the 28th May. Vertical axis: time (900 snapshots). Horizontal axis: -70 to 70 degrees (280 beams).



(a) Minimum-Norm after a log transform.



(b) root-MUSIC, computed via (5.23).

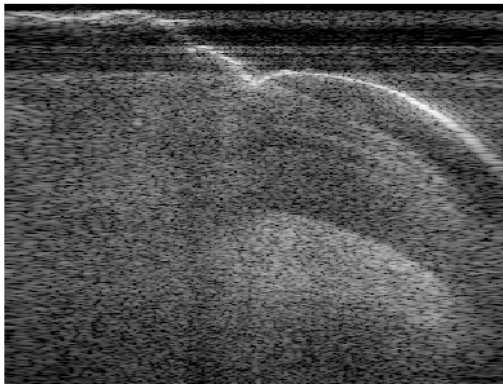


(c) root-MUSIC, computed via (5.25).

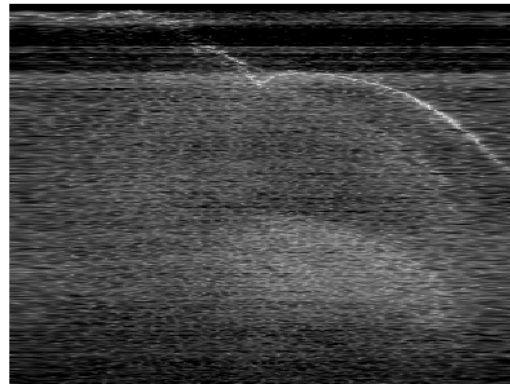


(d) ESPRIT.

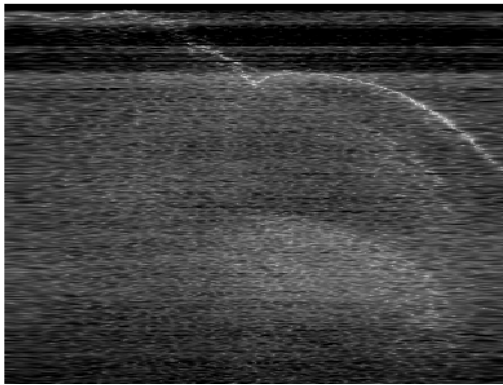
Figure 7.11: Spectra shown for the same ping as in Fig. 7.10. Vertical axis: time (900 snapshots). Horizontal axis: -70 to 70 degrees (280 beams).



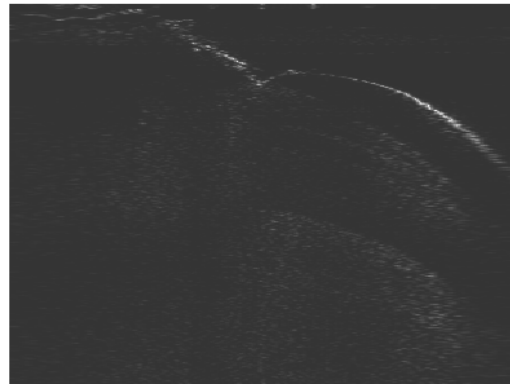
(a) Fourier Transform.



(b) Minimum-Variance.

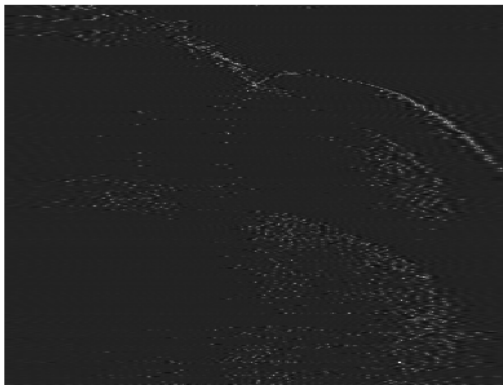


(c) Eigen-Vector.

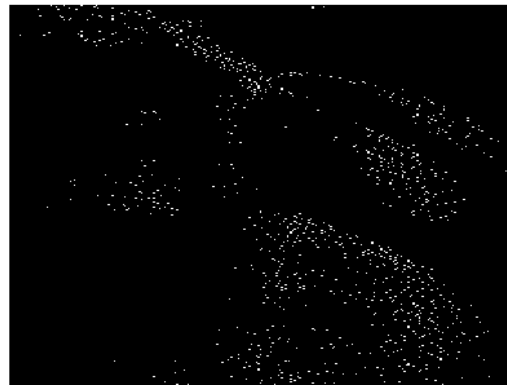


(d) MUSIC.

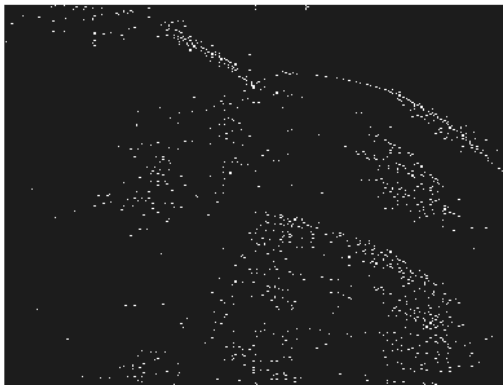
Figure 7.12: Spectra after a log transform. Another ping shot the 28th May. Vertical axis: time (1000 snapshots). Horizontal axis: -70 to 70 degrees (280 beams).



(a) Minimum-Norm after a log transform.



(b) root-MUSIC, computed via (5.23).



(c) root-MUSIC, computed via (5.25).



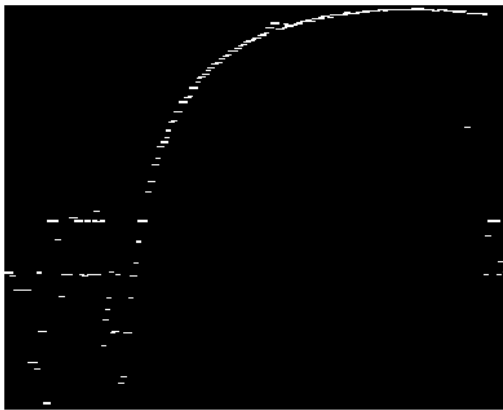
(d) ESPRIT.

Figure 7.13: Spectra shown for the same ping as in in Fig. 7.12. Vertical axis: time (1000 snapshots). Horizontal axis: -70 to 70 degrees (280 beams).

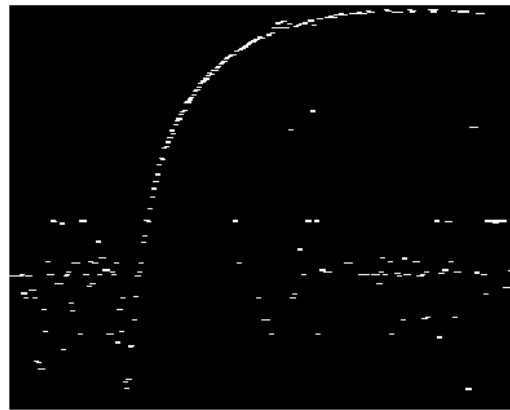
7.4.2 After bottom detection

Can the observations in previous section be transferred to the results after applying the simple bottom detection algorithm given in section 6.2? Let us look at some pictures resulting from this bottom detection algorithm.

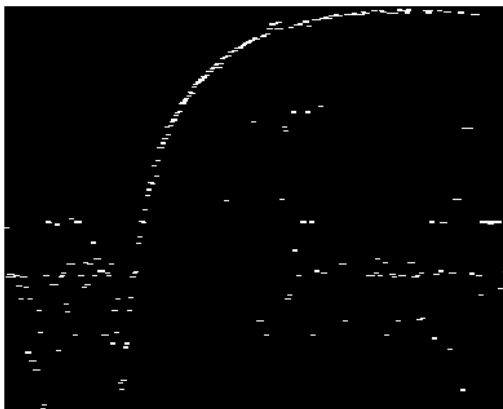
Note that it may appear as if bottom is detected for the same angle for several snapshots. This is not the case! It appears so, since every point in the spectra that is detected as bottom gives its value to its 8 neighboring “pixels”. This is done in order to make it easier to see all bottom points after detection. If only a single “pixel” was given color for each bottom point, we would hardly see the bottom at all after detection.



(a) Fourier Transform.



(b) Minimum-Variance.



(c) Eigen-Vector.



(d) MUSIC.

Figure 7.14: Spectra after a simple bottom detection algorithm. Same ping as in Fig. 7.8/7.9. Vertical axis: time. Horizontal axis: -70 to 70 degrees.

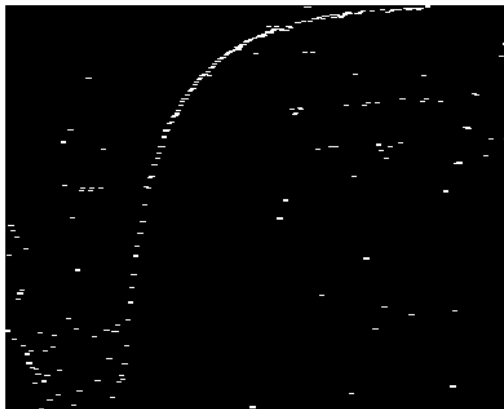
First we look at the results of applying the bottom detection algorithm on the spectra in figure 7.8 and 7.9. This is a very slow varying bottom with a low background



(a) Minimum-Norm.



(b) root-MUSIC, computed via (5.23).



(c) root-MUSIC, computed via (5.25).



(d) ESPRIT.

Figure 7.15: Spectra after a simple bottom detection algorithm. Same ping as in Fig. 7.8/7.9. Vertical axis: time. Horizontal axis: -70 to 70 degrees.

noise in the Fourier spectrum (Fig. 7.8(a)). Hence, one can expect the simple bottom detection algorithm to give fairly good results.

In section 7.4.1, it was found that MUSIC whitened the noise. Still, even though there is somewhat less noise to be seen for MUSIC (Fig. 7.14(d)) compared to Minimum-Variance and Eigen-Vector, the result of the bottom detection is better for the Fourier method (Fig. 7.14(a)) than for MUSIC with regard to erroneous bottom detections. Nevertheless, the bottom curve seems to be better connected for MUSIC than for the Fourier method, at least for late incoming signals.

Minimum-Norm (Fig. 7.15(a)) has a bottom curve that after bottom detection appears to be more crooked in some areas than that of MUSIC (Fig. 7.14(d)), but the noise level is acceptable. For root-MUSIC, computed by (5.23), it is difficult to distinguish between the bottom curve and the background noise (Fig. 7.15(b)). Computing the “spectrum” for root-MUSIC via the alternative (5.25), enables us to pick out the bottom curve. Still, there is more noise for root-MUSIC (Fig. 7.15(c)) after bottom detection than for MUSIC (Fig. 7.14(d)). ESPRIT has a well connected bottom curve after bottom detection, although some noise is still present (Fig. 7.15(d)).

Now, let us look at a more complicated bottom (Fig. 7.10(a)). The more complicated bottom yields no clear secondary echo. Rather, it gives several different reflections observed as a high background noise level at late times. This has little effect on the bottom detection of the Fourier spectrum (Fig. 7.16(a)), but what about the other methods?

For this bottom, it is clear that Minimum-Variance (Fig. 7.16(b)) and Eigen-Vector (Fig. 7.16(c)) have difficulties with distinguishing between noise and actual bottom returns. MUSIC (Fig. 7.16(d)) gives a better bottom curve, although some noise is still seen after bottom detection, especially to the far left. Minimum-Norm (Fig. 7.17(a)) has large problems with noise, and gives an almost identical bottom detection as root-MUSIC (Fig. 7.17(b)). The results of the alternative root-MUSIC are better (Fig. 7.17(c)), almost equal to what is achieved by ESPRIT (Fig. 7.17(d)), but somewhat more noisy. ESPRIT experiences the same problems as MUSIC to the far left.

Maybe we were not deceived by our eyes during the visual examination of the spectra in last section after all? At least not for the bottom returns to the far left.

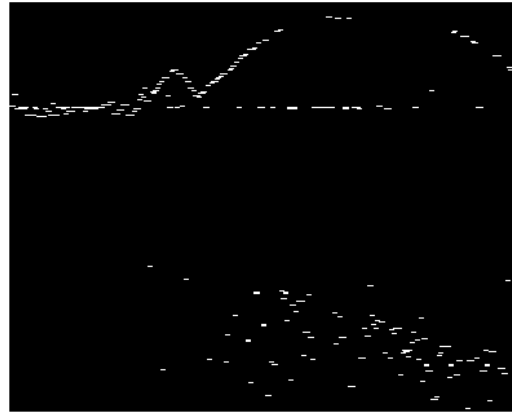
For some sea floors, there might be cases where no (or very weak) echoes are returned for some time periods. This is the case for the bottom observed in figures 7.18 and 7.19. In the Fourier spectrum of this bottom, several secondary echoes are observed (Fig. 7.12(a)).

As we can easily observe from the Fourier method, Minimum-Variance and Eigen-Vector (Fig. 7.18(a), 7.18(b) and 7.18(c)), there is a small period in time where no echoes are detected as bottom.

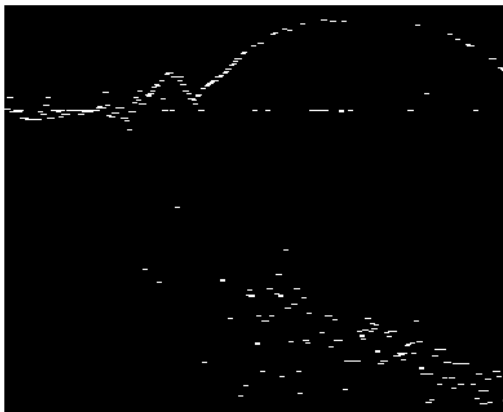
When the noise eigenvalues are weighted, or not used at all, the information about the strength of the noise is lost. The result of this is seen after bottom detection for MUSIC (Fig. 7.18(d)). MUSIC sets the L - M smallest eigenvalues to unity. For MUSIC, it seems as if the weak echoes discarded as noise (the gap in time) in the



(a) Fourier Transform.



(b) Minimum-Variance.



(c) Eigen-Vector.



(d) MUSIC.

Figure 7.16: Spectra after a simple bottom detection algorithm. Same ping as in Fig. 7.10/7.11. Vertical axis: time. Horizontal axis: -70 to 70 degrees.



(a) Minimum-Norm.



(b) root-MUSIC, computed via (5.23).



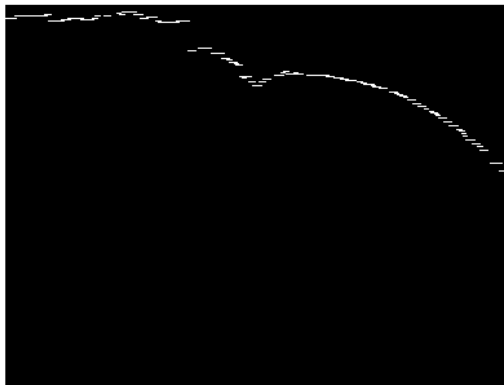
(c) root-MUSIC, computed via (5.25).



(d) ESPRIT.

Figure 7.17: Spectra after a simple bottom detection algorithm. Same ping as in Fig. 7.10/7.11. Vertical axis: time. Horizontal axis: -70 to 70 degrees.

bottom detection of the Fourier method have a higher strength than the echoes for the snapshots previous of the time gap, interpreted as bottom for the Fourier method. This is also observed for Minimum-Norm (Fig. 7.19(a)), which combines the noise eigenvectors without using the eigenvalues. For Minimum-Norm, more noise is seen after bottom detection, as for root-MUSIC (Fig. 7.19(b)) computed via (5.23). Both methods are unable to give a well defined bottom curve, but detect echoes in the time period where a gap is observed after bottom detection for the Fourier method. The alternative root-MUSIC, given by (5.25), also experiences some problems for the earliest received echoes (Fig. 7.19(c)), the bottom curve is not well defined here. ESPRIT has a better connected bottom curve for the earliest echoes (Fig. 7.19(d)), though not quite good. Also ESPRIT detects echoes in the time gap of Fourier.



(a) Fourier Transform.



(b) Minimum-Variance.



(c) Eigen-Vector.



(d) MUSIC.

Figure 7.18: Spectra after a simple bottom detection algorithm. Same ping as in Fig. 7.12/7.13. Vertical axis: time. Horizontal axis: -70 to 70 degrees.



(a) Minimum-Norm.



(b) root-MUSIC, computed via (5.23).



(c) root-MUSIC, computed via (5.25).



(d) ESPRIT.

Figure 7.19: Spectra after a simple bottom detection algorithm. Same ping as in Fig. 7.18. Vertical axis: time. Horizontal axis: -70 to 70 degrees.

Noise after bottom detection

From the figures we have shown so far, it is clear that some of the high-resolution methods suffer from noise after bottom detection. This means that some noise is interpreted as bottom, and hence some of the real bottom may be lost by applying this simple bottom detection algorithm. A good example of this is figure 7.16(c), where noise at late times is interpreted as bottom, leading to a badly connected bottom curve at earlier times. A second example is figure 7.18(d). Although little noise is observed, echoes are found in a time period where a gap is observed for the Fourier method (Fig. 7.18 and 7.20). This example is less clear, it might be that the results of the Fourier method are wrong.

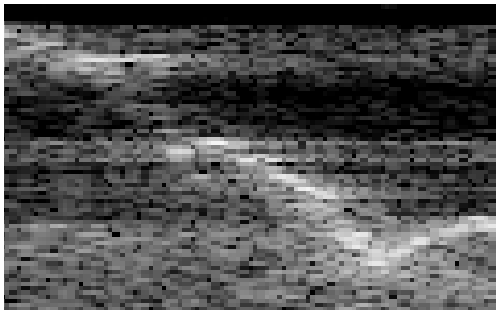
First, we discuss the gap in time (Fig. 7.20). Like MUSIC, Minimum-Norm and root-MUSIC, ESPRIT finds echoes in an area where the Fourier method has a clear gap. What does this mean? Is the bottom detection of the Fourier method (Fig. 7.20(b)) correct? Or, does the bottom detected for the Fourier method contain some noise? Is it possible that the three most modern high-resolution methods studied here (MUSIC, Minimum-Norm and ESPRIT) all interpret noise as bottom *in the same manner*, and that this noise happens to coincide so well with the rest of the bottom curve? Close inspection of figure 7.20(a) indicates that there are peaks in the Fourier spectrum in the same area that bottom is detected for MUSIC (Fig. 7.20(d)) and ESPRIT (Fig. 7.20(f)).

Is the bottom detection of the conventional Fourier method in error? Or, is it simply that since the information about the strength of the noise is not used for these high-resolution beamformers, some of the peaks of the high-resolution methods are given too much weight? Especially the results of ESPRIT are somewhat strange (Fig. 7.20(f)). The only physical explanation of these results is that the pulse has been “trapped” in the bottom for a few snapshots, and that the high resolution of ESPRIT enables it to detect these echoes when they return from a seemingly impossible direction/place. Close inspection of the alternative root-MUSIC (Fig. 7.19(c)) actually shows that it too detects echoes from this strange direction in the shadow of the former incoming echoes. If this physical explanation is correct, it seems that MUSIC’s lower resolution as a spectral method [22] has some advantages over the parametric methods, it still has higher resolution than the Fourier method.

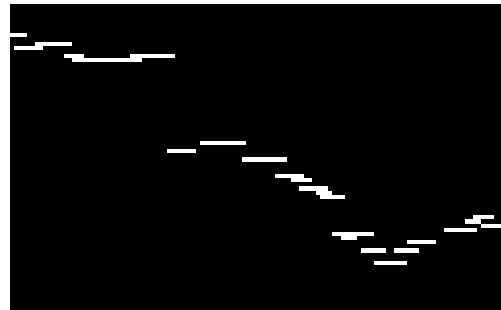
The number of signals, M , is probably chosen too high at the times where noise is observed. In the end of section 6.2, another strategy for choosing M was mentioned. M was suggested restricted when the standard deviation among the 6 largest eigenvalues was low. When this deviation is low, it is likely that few signals are present.

Let us look at some pictures where this new strategy for estimating M was used (Fig. 7.21 and 7.22). Here M is set to zero when the standard deviation among the six largest eigenvalues is lower than 1.3×10^5 , a quite high standard deviation is thus accepted. The number had to be set this high in order to obtain the gap in time for the other beamformers too.

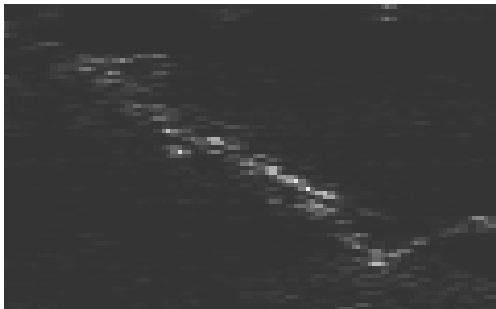
For Eigen-Vector (Fig. 7.21(a)), a better defined bottom curve is observed after bottom detection for this choice of M , even though a new problem emerges. A stripe is



(a) A small portion of the original Fourier spectrum.



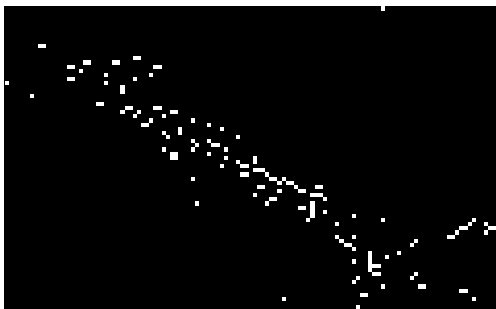
(b) The portion of the Fourier spectrum after bottom detection.



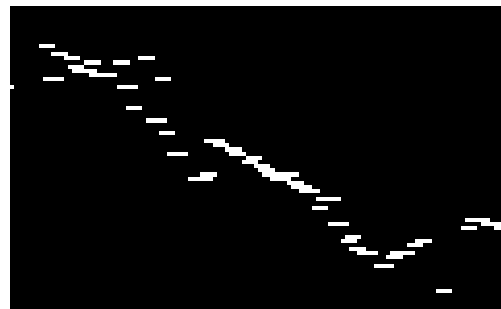
(c) The same portion of the original MUSIC spectrum.



(d) The result for MUSIC after bottom detection of the same portion.



(e) Identical portion of the ESPRIT "spectrum".



(f) The result for ESPRIT after bottom detection. Same portion.

Figure 7.20: A small area covering the time gap for the Fourier method (Fig. 7.18(a)) is selected to compare the bottom detection results of the Fourier method, MUSIC and ESPRIT. Vertical axis: time. Horizontal axis: angle.

present after bottom detection. Part of this stripe was also observed for the former way of estimating M (Fig. 7.16(c)), and part of it is seen after bottom detection for the Fourier method (Fig. 7.16(a)).

Part of the noise also disappears for MUSIC after bottom detection (Fig. 7.21(b)). Still, few differences are observed for the actual bottom curve compared to the former estimation technique for M (Fig. 7.16(d)). It may actually seem as if there is a larger gap in the bottom curve for the earliest echoes.

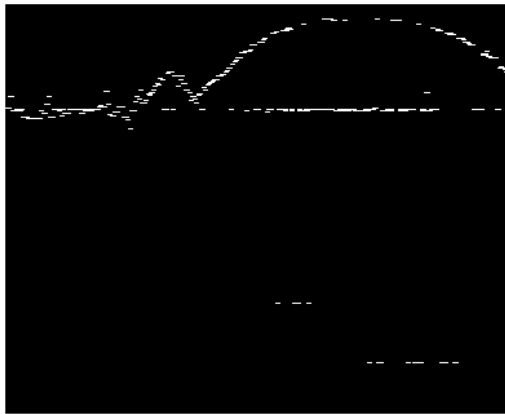
For Minimum-Norm (Fig. 7.21(c)/7.17(a)), the differences are significant. The new way of estimating M has a large impact on the bottom detection, almost all noise disappears and the bottom curve is well defined. Though the results in the left part are not quite good.

Root-MUSIC, defined by (5.23), manages to discard almost all noise after bottom detection for the new way of estimating M (Fig. 7.21(d)/7.17(b)). Nevertheless, the bottom curve is not well defined.

The bottom curve of the alternative root-MUSIC is better defined than that of the former root-MUSIC, but not as well connected. As for MUSIC, some of the bottom at early times is now discarded (Fig. 7.21(e)/7.17(c)) and replaced by noise at late times. The negative effect on the bottom curve is larger for the alternative root-MUSIC than for MUSIC, even though $\hat{\mathbf{R}}$ was computed in the same way.

For ESPRIT (Fig. 7.21(f)/7.17(d)), it is clear that less noise is accepted as bottom returns after bottom detection for the new estimation technique of M . Although, also here some of the bottom curve is lost, and the results in the left part of the figure are not quite satisfactory.

The effect of the new estimation technique of M for the bottom in figure 7.22, is mainly that now all the high-resolution methods result in a bottom with a gap in time - as the Fourier method (Fig. 7.18(a)). Since some of what was obviously bottom was lost by this new strategy in the former example, it is not clear that the new bottom detection results are correct. It might just be that some of the actual bottom is discarded compared to what was observed in figure 7.18 and 7.19. If we estimate the expected standard deviation for snapshots that are known to have no signals in them, it is found that M should be set to zero when the standard deviation is lower than 2.2×10^3 . This might indicate that M was too hard restricted in figures 7.21 and 7.22, where M was set to zero when the standard deviation was 1.3×10^5 - an order of 100 higher.



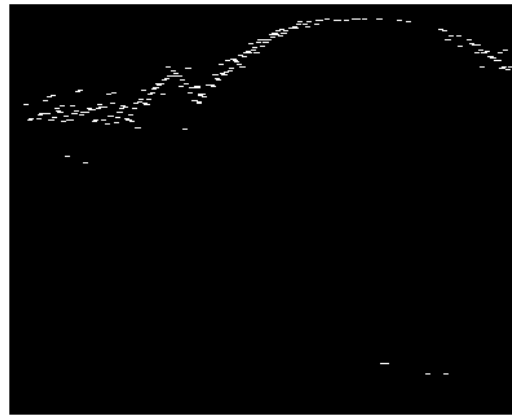
(a) Eigen-Vector



(b) MUSIC



(c) Minimum-Norm



(d) root-MUSIC, computed via (5.23).



(e) root-MUSIC, computed via (5.25).



(f) ESPRIT

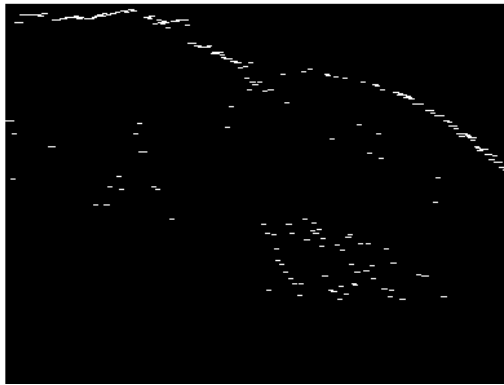
Figure 7.21: Effect on the simple bottom detection algorithm when trying to estimate M in a smarter way. Same ping as in Fig. 7.16 and 7.17. Vertical axis: time. Horizontal axis: -70 to 70 degrees.



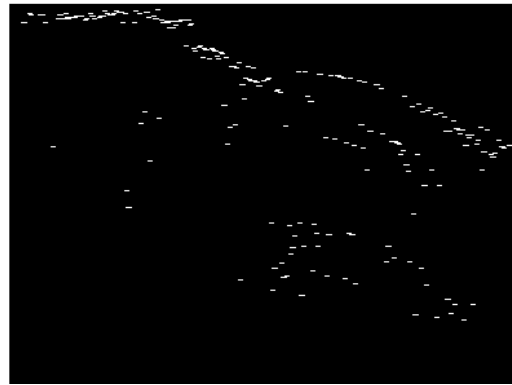
(a) Eigen-Vector



(b) MUSIC



(c) Minimum-Norm



(d) root-MUSIC, computed via (5.23).



(e) root-MUSIC, computed via (5.25).



(f) ESPRIT

Figure 7.22: Effect on the simple bottom detection algorithm when trying to estimate M in a smarter way. Same ping as in Fig. 7.18 and 7.19. Vertical axis: time. Horizontal axis: -70 to 70 degrees.

7.4.3 Summary of the Observations

From the observations in section 7.4.1 and 7.4.2, we gather that largely there is less noise in the *DOA* estimates of MUSIC than for Minimum-Variance, Eigen-Vector and Minimum-Norm. The alternative root-MUSIC also has slightly more noise than MUSIC, while ESPRIT performs roughly equal to MUSIC.

From the discussions of figures 7.21 and 7.22, it is clear that MUSIC is more robust than Eigen-Vector and Minimum-Norm with regard to estimation of M . MUSIC is also somewhat more stable than the alternative root-MUSIC. ESPRIT is also stable, but its high resolution may actually cause problems.

Spectra and bottom detection results were also computed when $\hat{\mathbf{R}}$ was estimated in the “worst possible” way for each method. These results indicated that neither Minimum-Variance nor Eigen-Vector were robust regarding estimation of $\hat{\mathbf{R}}$. MUSIC was quite robust, even though some noise was seen. Also ESPRIT had problems with noise, but was almost as robust as MUSIC. Both root-MUSIC approaches suffered from noise, but were not as “hopeless” as Minimum-Norm. This observation is also supported by the quantitative measures in section 7.1.

Thus, we conclude that MUSIC and ESPRIT are the two most promising beamformers for high-resolution bathymetric mapping of the sea-floor.

Chapter 8

Discussion

An important issue in developing next generation high-performance multibeam echo sounders is to increase the mapping resolution. Research has been done to improve the along-track resolution using synthetic aperture methods (eg [20]). At the same time, modern high-resolution beamforming methods are interesting for improvement of the cross-track resolution.

Using raw data from Simrad EM3000 multibeam echo sounders, we tested several high-resolution beamformers. Estimation of the spatial covariance matrix, $\hat{\mathbf{R}}$, plays an important role in high-resolution beamforming. In addition to forward/backward averaging and subarray smoothing, a noise covariance subtraction aimed to reduce structured noise was applied. For most of the high-resolution methods, this noise subtraction improved the performance. Quantitative measures were developed and used as part of the evaluation. MUSIC and ESPRIT gave the best results according to the evaluation. This was supported by visual examination of different spectra and results of a simple bottom detection algorithm for these spectra.

All the high-resolution beamformers reduced the beamwidth and improved the resolution. This can be seen from the *PWR* (p. 50) values which were all higher than 1 (Tab. 7.8), and the figures in section 7.4, where all high-resolution beamformers gave narrower spectrum peaks than the Fourier method. The bias of the methods was not easy to estimate. We compared the location of peaks in spectra of the Fourier method with those in spectra of high-resolution beamformers. MUSIC and ESPRIT had the highest *WLO* (p. 52) values and are considered as the most reliable (Tab. 7.8). ESPRIT is computationally efficient, while MUSIC is much more time demanding than the Fourier method. MUSIC has to compute the $L - M$ smallest eigenvectors, while ESPRIT only need the M largest ones. In our case, L , the number of sensors is 80, and M , the number of signals is typically 2 to 3. The simple bottom detection algorithm included more noise peaks for MUSIC and ESPRIT than for the Fourier transform.

8.1 Stationarity Problems: Causes and Cures

The sonar scenario studied in this thesis is quite different from other scenarios where high-resolution beamforming methods are used (eg [11]). In a bathymetric sonar application, the number of stationary snapshots can be very low compared to those for, say, a passive sonar. Typically one can only assume the scene to be stationary for a few snapshots, depending upon the sample rate and the pulse length. For the EM3000 this means that averaging in time is not available for enhancing the signal to noise ratio (*SNR*). This problem could be reduced by sampling at a higher rate, and thereafter claiming stationarity for a few snapshots. By that approach, one could improve the estimation of the covariance matrix and enhance the *SNR*. The beamformers would perform better.

Such an approach is perhaps best suited in a deep ocean echo sounder, as the signals could be claimed to be stationary for a longer period of time there compared to in a shallow water application. For instance is the pulse of the Simrad EM12 sampled 3 times, to some extent enabling us to perform time averaging.

The ultimate solution would of course be to sample at such a high rate that one could assume the signals to be stationary for quite a few snapshots. By using a simple update of some ring-buffer containing the $\hat{\mathbf{R}}$'s to be averaged, we could compute the *DOA* of the snapshot in the middle each time a new snapshot was taken. In addition, this would give a higher temporal resolution.

8.1.1 Violating the Stationarity Criterion

The effect of applying the forward-only approach, given by (4.7), for improving the estimated covariance matrix may not be as severe for the Fourier method as for the high-resolution beamformers. Since the Fourier method has a rather low resolution, a slight difference in source direction for say $N = 3$ may not be noticed at all. High-resolution methods, at least the parametric, might experience more problems with this approach. For $N = 3$ one could then stand the risk of finding 3 closely spaced peaks, depending upon how high the resolution actually was, and whether the 3 echoes were coherent. The forward/backward smoothing, defined by (4.9), would only make things worse, since this would decorrelate the 3 signals.

8.1.2 Along-track Resolution

Another problem is the assumption that the signals arrive only from the y - z plane (Fig. 3.3). This assumption is violated when the along-track resolution is low. It seems that the cross-track resolution and the along-track resolution should be increased simultaneously. Unfortunately, this would exclude another possibility for improvement of the estimation of $\hat{\mathbf{R}}$ - averaging of succeeding pings. If the vessel did not move too quick, the beam along-track would overlap for a few pings, actually ensuring stationarity.

Such averaging of succeeding pings could naturally be combined with averaging of snapshots, giving an even lower variance of the *DOA* estimates. It is not easy to say which is the more expensive approach. (1) Slow mapping, allowing overlap of succeeding pings. (2) Several receive arrays in parallel (used for stacking), enabling higher along-track resolution. In addition, this approach should give better estimates of $\hat{\mathbf{R}}$. The estimates should perhaps be further improved by a higher sampling rate and averaging of succeeding snapshots.

8.2 Sea-bed Imaging

Some of the high-resolution beamformers are incapable of sea-bed imaging. Root-MUSIC and ESPRIT define the “power” to equal 1 in the signal direction, while MUSIC, Minimum-Norm and Eigen-Vector measure the “distance” from the noise subspace to the signal subspace. To come about this, one *could* (beware of the bias) first find the *DOA* by any of the high-resolution beamformers (preferably MUSIC or ESPRIT), and then compute the spatial spectrum for *only* the signal direction via the Fourier method. This would not lead to a significant extra computational load, relative to the load of the high-resolution beamformer, as we now avoid searching through all angles with the Fourier method.

Minimum-Variance, on the other hand, tries to minimize the effect of signals arriving from other directions than the current direction of look. Although Minimum-Variance does not measure the true power of the echoes, this might indicate that Minimum-Variance is even more suited for sea-bed imaging (ignoring its bias) than the Fourier method. After all, the Fourier method simply maximizes the entire power, not only from the interesting direction as Minimum-Variance.

8.3 Bottom Detection

Generally, the high-resolution methods that gave well connected bottom curves after bottom detection also had fewer gaps in time than the Fourier method. That is to say, they were better connected than the Fourier method in areas where there is only a slight difference in *DOA* from one snapshot to the next. This is due to the broad mainlobe of the Fourier method, which also covers some of the area of the mainlobe of the succeeding snapshot. Thus, some of the peaks of the Fourier method will not be chosen by the simple bottom detection algorithm applied here, since they may lay in the “shadow” of the peak of the previous snapshot. Hopefully, more sophisticated bottom detection algorithms are applied in real bathymetric sonars!

On the other hand, in areas where several echoes impinge on the array simultaneously, from approximately the same *DOAs*, the Fourier method gave better connected bottom curves than the sub-space based methods. This is due to the estimation of M . As an example: If the sonar was applied in an area where the sea-floor was shaped as a perfect half-pipe, all echoes would impinge on the array simultaneously.

The number of signals had to be set as high as the number of beams used in the Fourier transform to obtain the same estimate of the sea-floor. This is impossible, M can not even be as high as the number of sensors (Sec. 3.2).

Since Minimum-Norm and the original root-MUSIC pick the highest peak somewhat randomly, these methods must be considered less suited for bathymetric mapping. The M peaks are roughly covered by those of the Fourier spectrum, but the highest peaks in the spectra produced by the original root-MUSIC and Minimum-Norm do not match the highest peaks of the FT (Sec. 7.2). Consider the simple bottom detection algorithm from section 6.2, what will the effect of a random choice of the highest peak result in for this bottom detection method? The answer is that when running through all snapshots for each angle value, one might very well get a degraded picture of the bottom. Peaks that should have been accepted may be excluded, while erroneous peaks might be approved as bottom points.

8.4 Where do we go?

More research has to be done in order to apply high-resolution beamforming in a multibeam sonar product. The discussion about estimation of M , at the end of section 7.4, points out one of the major problems with high-resolution beamformers. These beamformers are very dependent upon the assumptions that are made. If the assumptions are erroneous, it is not necessarily easy to come about the problems caused, or mistakes made, by these assumptions. Thus, until more research is done on high-resolution beamformers, it is safest to combine them with the conventional beamformer applied today. For this purpose, beamspace processing [11] is probably best suited. Although beamspace processing depends on an estimate of the signal directions, it reduces the dimensionality of the problem, and can also improve the resolution of spectral-based methods [11].

Today, high-resolution beamformers are probably more suited for passive sonars. For instance, if a submarine tried to find a vessel far away, this vessel would move slow relative to its distance. Thus, it would be possible to apply time averaging without a drastic loss of resolution. Searches for submarines, by sub-surface explosions, also enable us to perform time averaging. I assume the signal from such an explosion is fairly long, and could be sampled several times and then summed. This would give a good estimate of the whereabouts of the submarine.

Research Before developing a new system for bathymetric measures, at least three topics should be looked into:

- *Estimation of M .* If the mainlobe of the Fourier method was wider than expected for some angle, one could assume that M was higher than 1 in this area. We could also apply the so far estimated slope of the sea-floor, and combine the

height of the peak with this information. If we had a “database” containing information about typical numbers of M for different peak-widths, peak-heights and bottom slopes, it could enable us to obtain a “qualified guess” of M 's size.

- *Performance analysis of different beamspace methods.* This analysis should preferably be applied to raw data, where more ground truth about the bottom depth was available (eg a basin, in which different objects could be placed). This would enable a better analysis of the performance than what was achieved by the different assumptions that had to be taken in this thesis.
- *Compare the beamspace methods with the subarray interferometry* suggested in [32], since the interferometry approach assumes a single scatter. The beamspace approaches should apply the estimated M to find all signals within the area enveloped by the Fourier mainlobe.

Having performed this research, the question of implementation enters the scene. Is it possible, in an on-line system of today, to perform time averaging, obtain a Fourier spectrum for sea-bed imaging, estimate M and finally apply a high-resolution/beamspace method for bottom detection within the peaks of the Fourier spectrum? This is unclear, but as the computing power is becoming cheaper we might some day be able to. The beamspace root-MUSIC proposed in [33] allows for a reduced polynomial, which combined with the alternative root-MUSIC proposed in this thesis might outperform ESPRIT regarding computational efficiency. If a beamspace ESPRIT algorithm is available, this can be expected to be even more efficient.

Appendix A

Notation and Abbreviations

This appendix gives a short introduction to the mathematical notation that is used in this thesis. It also contains a table with the abbreviations that are used throughout the thesis (Tab. A.1).

Throughout the thesis different letters, boldface letters, capital letters and boldface capital letters are encountered. Small letters, like a , are mathematical sizes. Boldface letters, such as \mathbf{a} , are vectors and boldface capital letters are matrices. Otherwise, capital letters are only used for the Fourier transform and as abbreviations.

The symbol $*$ in \mathbf{a}^* means the complex conjugate of the vector \mathbf{a} . It is also possible to take the transpose of a vector or matrix, while conjugating it. For instance \mathbf{A}^H , where H is the Hermitian (conjugate transpose). The symbol \dagger is used to denote the pseudoinverse of a matrix, as in \mathbf{A}^\dagger .

Notation	Meaning
Signal Related	
ω	Frequency ($\omega = 2\pi f$)
f_0	Center frequency (300158 Hz)
c	Speed (1500 m/s)
θ	Incidence angle on the array
k	Wavenumber ($ \mathbf{k} $)
M	Number of signals
Array Related	
d	Intersensor spacing
L	Number of sensors
K	Number of subarrays
f_s	Sampling frequency (14293 Hz)
Abbreviations	
\mathcal{E}	Expectation
$\hat{\mathbf{R}}$	Spatial covariance matrix
SNR	Signal to Noise Ratio
DOA	Direction Of Arrival
Evaluation Abbreviations	
PWR	Peak-Width Ratio
$PWR(M)$	Peak-Width Ratio between M peaks
WLO	Width Location Overlap
$WLO(M)$	Width Location Overlap of M peaks
TCR	Total of Close but Resolved sources
WLO_{TCR}	WLO of the closely spaced sources
M-flops	Mega floating point operations

Table A.1: Abbreviations and the most important quantities in the thesis.

Appendix B

Paper written for Oceans '99

This appendix contains an article written for the conference Oceans '99.

OCEANS '99 MTS/IEEE

September 13-16, 1999

Washington State Convention Center & The Sheraton Hotel
Seattle, Washington USA

The article was written during ultimo May, primo June. The article can also be found in the proceedings from the conference [25].

High-resolution beamforming for multibeam echo sounders using raw EM3000 data

Are Rønhovde

Department of Informatics
University of Oslo
P. O. Box 1080, Blindern
N-0316 Oslo, Norway

Luren Yang

SINTEF-ECY
P. O. Box 124, Blindern
N-0314 Oslo, Norway

Torfinn Taxt

Department of Physiology
University of Bergen
Årstadveien 19
N-5009 Bergen, Norway

Sverre Holm

Department of Informatics
University of Oslo
P. O. Box 1080, Blindern
N-0316 Oslo, Norway

Abstract — This paper deals with high-resolution beamforming for multibeam echo sounders with a linear receive-array. A fundamental limitation of the conventional beamformer is that the angular resolution depends upon the size of the receive-array. The use of subarray interferometry may increase the mapping resolution for oblique beams, but this method is not valid in the case of multiple scatterers. Therefore, it is interesting to examine how modern high-resolution beamformers work for multibeam echo sounders. Early research for this purpose, based on data simulation, showed promising results. In this work, we apply high-resolution beamformers to raw sonar data recorded at the receiver elements of Simrad EM3000 multibeam echo sounders. The estimation of spatial covariances plays an important role in high-resolution beamforming. A method to improve the estimation by subtracting noise covariances is applied. Among several beamformers we tested, MUSIC and ESPRIT gave the best results. Compared to the Fourier transform beamformer, high-resolution beamformers clearly improved the resolution. However, the noise level might also increase. It seems that a combined use of conventional and high-resolution beamforming may improve the sonar performance.

I. INTRODUCTION

Multibeam echo sounding is currently one of the most important technologies for seafloor mapping and imaging. It involves sending narrow band pulsed acoustic signals towards the seafloor, and analyzing the echo signals for bathymetry and image formation. The configuration of the transducer arrays and the resulting beam patterns for these types of sonars are shown in Fig. 1. The transmitting array is mounted parallel to the motion direction of the sonar platform. It transmits a fan-shaped beam which is narrow in fore-aft direction and wide athwartships. The receive-array is mounted perpendicular to the transmit-array. A number of receive beams are formed at different angles. For each receive beam, the bottom echo from the intersection of the transmit and the receive beam footprint is detected. The echo arrival time and the angle of the receive beam provide information for bathymetry, and the backscatter-

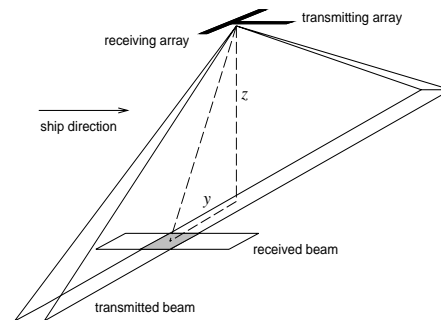


Fig. 1: *T* configuration of linear transducer arrays and the resulting beam patterns of a multibeam echo sounder system.

ing strength is used for seabed imaging. A transmission and receiving circle is called a ping. Sea-bottom maps and images are formed by combining data from many consecutive pings. De Moustier [1] explained the principles of the multibeam echo sounders and analyzed several commercial systems.

Two important issues in developing multibeam systems are the mapping resolution and the bottom detection accuracy. Current multibeam echo sounders use a conventional method, usually the Fast Fourier Transform, to form receiving beams. The beamwidth is normally 1.5 to 5 degrees athwartships. The spatial resolution is unsatisfactory, especially for beams of oblique incidence angles. For oblique beams, the cross-track length of the receive beam footprints are large, and bottom echo detection based on the beam amplitude values is inaccurate. The bottom detection accuracy and the mapping resolution can be improved by using subarray interferometry [2], since for an oblique beam, the pulse length is much shorter than the cross-track length of the footprint. However, the method assumes a single scatterer within each beam footprint. It works well when the bottom curve is smooth within a footprint, but fails for complex bottom curves with multiple scatterers. Thus, it is interesting to examine how modern high-resolution beamformers work for multibeam echo sounders.

A review of high-resolution beamforming is given in [3].

Some research work has been done to apply high-resolution beamforming to multibeam echo sounders. Pantartzis *et al.* [4] evaluated the performance of four methods, i.e. the Yule-Walker method for AR model, the unconstrained least square method for AR model, Minimum-Variance, and MUSIC. Their work was based on simulated data, generated by a sonar simulation software. The simulator creates synthetic reverberation by randomly distributing ideal point scatterers in the volume and at the boundaries of the ocean, and coherently combining the individual echoes at the receivers. The bottom was modeled as a flat, fine, sand surface. The MUSIC method produced narrower spectrum peaks than the other methods, and therefore showed higher resolution potential. Compared to the conventional Fourier beamforming, it also provided a reduction of root mean squared error in bottom echo detection. MUSIC was later evaluated by using a sea floor model for a rough surface [5], and gave reliable bathymetric estimates. Talukdar [6] examined the application feasibility of AR spectrum estimation in the Sea Beam 2100 multibeam system design. Simulation results indicated potential for improving bottom detection accuracy.

The above studies were based on data simulation. In this work, we apply high-resolution beamforming to raw Simrad EM3000 sonar data and examine the beamforming performance for high frequency shallow water sonars. The methods tested so far are Minimum-Variance, Eigen-Vector, MUSIC, Minimum-Norm, root-MUSIC and ESPRIT.

II. SIGNAL MODEL

The following signal model is assumed: The receive-array of the multibeam echo sounder is a uniform linear array. The array consists of sensors with the same directional sensitivity, equally spaced along the y -axis. The ship runs in the x direction and the depth is measured along the z -axis. The directional sensitivity of each sensor, g , is such that the array mainly collects energy from the y - z plane. Also, g is assumed independent of the signal's direction θ . Let $s_m(t)$ be a signal coming from direction θ_m . Its steering vector is then

$$\mathbf{a}(\theta_m) = g[1 \ e^{-jkd \sin \theta_m} \ \dots \ e^{-j(L-1)kd \sin \theta_m}]^T \quad (1)$$

Here, k is the wavenumber, d is the interelement distance and L is the number of sensors. The output vector of the array for signal $s_m(t)$ is

$$\mathbf{y}_m(t) = \mathbf{a}(\theta_m)s_m(t) \quad (2)$$

which says that a far-field signal has linear phase shifts over the array.

Further, if M signals impinge on the array at one time instant, the output of the array will be

$$\mathbf{y}(t) = \sum_{m=1}^M \mathbf{a}(\theta_m)s_m(t) \quad (3)$$

This can be written as

$$\mathbf{y}(t) = \mathbf{A}(\theta)\mathbf{s}(t) \quad (4)$$

where $\mathbf{A}(\theta) = [\mathbf{a}(\theta_1) \ \dots \ \mathbf{a}(\theta_M)]$ is a steering matrix and $\mathbf{s}(t) = [s_1(t) \ \dots \ s_M(t)]^T$ is a vector of signals. Adding noise to (4) yields

$$\mathbf{y}(t) = \mathbf{A}(\theta)\mathbf{s}(t) + \mathbf{n}(t) \quad (5)$$

This vector is the "actual" output of the array, and from it we estimate the covariance matrix $\hat{\mathbf{R}}$.

$$\hat{\mathbf{R}} = \frac{1}{N} \sum_{t=1}^N \mathbf{y}(t)\mathbf{y}^H(t) \quad (6)$$

Here, $\mathbf{y}^H(t)$ is the Hermitian of $\mathbf{y}(t)$. N is the number of samples taken over a period in time that is assumed to give stationary results for the signals. In narrow band multibeam echo sounders, the sampling frequency is determined by the bandwidth, a signal pulse is often sampled only once or twice. The received signal is therefore not stationary in time, and we assume that $N = 1$.

III. ESTIMATION OF COVARIANCES

High-resolution beamformers are often used in applications where signals are stationary over several samples (e.g. [3]). In these cases, large amount of data are available for estimating the covariance matrix $\hat{\mathbf{R}}$. In our case $N = 1$, and thus the estimated covariance matrix must be improved in some way.

The matrix $\hat{\mathbf{R}}$ can be decomposed into eigenvalues and eigenvectors. In matrix form, the eigenvalues are $\hat{\Lambda}$ and the eigenvectors are $\hat{\mathbf{V}}$. Further, some of these eigenvectors ideally span the signal subspace, while the others span the noise subspace. In reality, we split the matrix $\hat{\mathbf{V}}$ into $\hat{\mathbf{V}}_s$ and $\hat{\mathbf{V}}_n$. Here, $\hat{\mathbf{V}}_s$ spans the signal+noise subspace and $\hat{\mathbf{V}}_n$ spans the noise subspace. Methods that work with these subspaces are often called subspace methods.

We need to improve $\hat{\mathbf{R}}$ in the cases where coherent signals are present. The reason is that the high-resolution methods may fail to produce peaks at the directions of arrival (DOA), since one or more of the eigenvectors in the signal subspace will diverge into the noise subspace in coherent cases [3]. We may also want to achieve a smaller variance in the estimates, this can be done using the same operations on $\hat{\mathbf{R}}$, even though no coherent signals are present.

To improve $\hat{\mathbf{R}}$, different approaches are available. Among them are forward/backward averaging and spatial smoothing. The forward/backward averaging [7, 8] works by left and right multiplying $\hat{\mathbf{R}}^*$ with the matrix \mathbf{J} , where \mathbf{J} is an exchange matrix consisting of an identity matrix of size $L \times L$ flipped left/right. The new estimate is

$$\hat{\mathbf{R}}_{FB} = \frac{1}{2} \left(\hat{\mathbf{R}} + \mathbf{J}\hat{\mathbf{R}}^*\mathbf{J} \right) \quad (7)$$

This strategy only works for two coherent signals, but will then restore the rank of the signal covariance matrix [3]. When there are more than two coherent signals, we use a spatial smoothing of $\hat{\mathbf{R}}$. The idea is to split the array in several equal and overlapping subarrays, which are assumed to have identical steering vectors, except for their phase factor [3]. The subarrays are of size L_K , where $L_K = L - K + 1$ and K is the number of subarrays. The resulting covariance matrix is

$$\hat{\mathbf{R}}_{SS} = \frac{1}{K} \sum_{k=1}^K \hat{\mathbf{R}}_k \quad (8)$$

where each $\hat{\mathbf{R}}_k$ is of size $L_K \times L_K$ and is part of $\hat{\mathbf{R}}$, centered along its diagonal.

It is also possible to combine the two strategies above, by first applying (8) and then (7). This generally results in better estimates for the subspace methods except for the MUSIC algorithm [7]. For state space methods/ESPRIT, Rao and Hari [8] have found that such a combination is preferable to only applying spatial smoothing.

Rao and Hari have also developed some theory for optimal number of subarrays in [7, 8]. Here they find that for two sources the optimal choice of K is $(L + 1)/5$ for high to moderate SNR. While for lower signal-to-noise ratio (SNR), K is best chosen as $(L + 1)/8$. These choices are valid for the ESPRIT and Minimum-Norm algorithms. When it comes to MUSIC, minimal smoothing appears to be desirable to obtain reliable DOA estimates [7].

An additional strategy for improving $\hat{\mathbf{R}}$ is used. The idea is to estimate the structured noise present in the array. This noise is estimated from the samples taken before the first bottom echo returns, and is later subtracted from $\hat{\mathbf{R}}$ at each sample. If no structured noise is present, the effect of this will be none. But, if the array in some way produces a systematically wrong output, this output will be corrected. It can be viewed as calibrating the array. The noise is estimated as

$$\hat{\mathbf{N}} = \frac{1}{N_n} \sum_{t=1}^{N_n} \mathbf{y}(t) \mathbf{y}^H(t) \quad (9)$$

where N_n is the number of samples used to estimate the noise. When applied, this noise removal is done before the operations given by (7) and (8). Note that since finding the eigenvectors of $\hat{\mathbf{R}}$ is an iterative procedure one might expect the procedure to converge faster when having removed the noise covariance.

IV. HIGH-RESOLUTION BEAMFORMING

Beamforming is a spatial filtering which separates signals coming from different propagation directions. Methods such as delay-and-sum and Fourier transform (FT) are called conventional methods, as they have a relatively long history. These methods are well understood and common in array signal processing. However, they suffer from a

fundamental limitation: their performance is directly dependent on the physical size of the array. Modern high-resolution beamforming methods are designed to overcome this limitation, to some extent. These methods make assumptions on signal and noise characteristics, and are aimed at resolving closely spaced signal sources. However, the results of these methods are sensitive to the assumptions made.

Minimum-Variance proposed by Capon [9] tries to optimize the beamforming process according to the time-varying covariance matrix. Its spectrum is given by

$$P_{MV}(\theta) = \frac{1}{\mathbf{a}^H(\theta) \hat{\mathbf{R}}^{-1} \mathbf{a}(\theta)} \quad (10)$$

The method minimizes the power contributed by noise and signals originating from other directions than the current steering direction, while maintaining a fixed gain at the steering direction. Its performance is therefore dependent upon the SNR [3]. Since we only use one sample for estimating $\hat{\mathbf{R}}$, $\hat{\mathbf{R}}^{-1}$ may not exist. The pseudoinverse is therefore used instead.

The Eigen-Vector and MUSIC algorithms are based on rewriting the Minimum-Variance algorithm [10].

Eigen-Vector uses only the noise subspace in the covariance matrix. The spectrum is

$$P_{EV}(\theta) = \frac{1}{\mathbf{a}^H(\theta) \hat{\mathbf{V}}_n \hat{\mathbf{\Lambda}}_n^{-1} \hat{\mathbf{V}}_n^H \mathbf{a}(\theta)} \quad (11)$$

The modification of the covariance matrix will lift the signal peaks out of the noise in the spectrum. Here, $\hat{\mathbf{\Lambda}}_n^{-1}$ may not exist, and again the pseudoinverse is used.

MUSIC In the MUSIC algorithm, the eigenvalues are set to unity. The spectrum becomes

$$P_{MU}(\theta) = \frac{1}{\mathbf{a}^H(\theta) \hat{\mathbf{V}}_n \hat{\mathbf{V}}_n^H \mathbf{a}(\theta)} \quad (12)$$

Setting the eigenvalues to one can be viewed as whitening the noise subspace [10].

Minimum-Norm introduces a weight matrix [3] in the MUSIC spectrum (12)

$$P_{MN}(\theta) = \frac{1}{\mathbf{a}^H(\theta) \hat{\mathbf{V}}_n \hat{\mathbf{V}}_n^H \mathbf{W} \hat{\mathbf{V}}_n \hat{\mathbf{V}}_n^H \mathbf{a}(\theta)} \quad (13)$$

Here, $\mathbf{W} = \mathbf{e}_1 \mathbf{e}_1^T$ where \mathbf{e}_1 equals the first column of an $L \times L$ identity matrix. Weighting the MUSIC algorithm in this way should yield a lower bias and a better resolution than the original MUSIC algorithm (e.g. [3]).

Root-MUSIC For the MUSIC algorithm there is a polynomial-rooting version available, known as root-MUSIC [11]. The polynomial of $z = e^{jkd \sin \theta}$ can be expressed as

$$p(z) = \sum_{k=-(L-1)}^{L-1} c_k z^k \quad (14)$$

where c_k is the sum of the elements on the k 'th diagonal of $\widehat{\mathbf{V}}_n \widehat{\mathbf{V}}_n^H$. Here $k < 0$ gives the k 'th subdiagonal, while $k > 0$ yields the k 'th superdiagonal. The roots of the $2(L-1)$ degree polynomial $p(z)$ lie as mirrored pairs around the unit circle. In root-MUSIC, the M largest roots, $\hat{z}_1, \dots, \hat{z}_M$, that lie inside or on the unit circle are taken as the DOA estimates. The angles of the M signals are then computed as

$$\hat{\theta}_m = \arccos\left(\frac{1}{kd} \arg\{\hat{z}_m\}\right), \quad m = 1, 2, \dots, M \quad (15)$$

ESPRIT The idea behind ESPRIT is to split the array in two subarrays, separated by a known displacement vector $\mathbf{\Delta}$ of magnitude Δ [12]. The DOA estimates are then angles of arrival with respect to the direction of $\mathbf{\Delta}$. For the uniform linear array we can split the array into two subarrays separated by distance d . Both subarrays are placed along the y -axis. The subarrays will be denoted Y' and Y'' . The signals received at sensor i in each of the subarrays, can be expressed as:

$$\begin{aligned} y_i'(t) &= \sum_{m=1}^M s_m(t) a_i(\theta_m) + n_{y_i}'(t), \\ y_i''(t) &= \sum_{m=1}^M s_m(t) e^{jkd \sin \theta_m} a_i(\theta_m) + n_{y_i}''(t), \end{aligned} \quad (16)$$

If we combine the sensor outputs in each subarray into vectors, we can write (16) as

$$\begin{aligned} \mathbf{y}'(t) &= \mathbf{A}' \mathbf{s}(t) + \mathbf{n}_{y'}(t), \\ \mathbf{y}''(t) &= \mathbf{A}'' \mathbf{s}(t) + \mathbf{n}_{y''}(t) \end{aligned} \quad (17)$$

Here \mathbf{A}' is the upper $(L-1) \times M$ part of the steering matrix \mathbf{A} given in (4). Let

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}' \\ \text{last row} \end{bmatrix} = \begin{bmatrix} \text{first row} \\ \mathbf{A}'' \end{bmatrix} \quad (18)$$

This gives

$$\mathbf{A}'' = \mathbf{A}' \mathbf{\Phi} \quad (19)$$

Further, $\mathbf{s}(t)$ is the $M \times 1$ vector of signals, and the matrix $\mathbf{\Phi}$ is a diagonal $M \times M$ matrix containing the phase delays between the subarrays for the M signals

$$\mathbf{\Phi} = \text{diag}\{e^{j\phi_1}, \dots, e^{j\phi_M}\} \quad (20)$$

where $\phi_m = kd \sin \theta_m$. The problem of estimating the DOAs is then to find these phase delays, hence finding

$\mathbf{\Phi}$. The unitary matrix $\mathbf{\Phi}$ relates the measurements from subarray Y' to those from subarray Y'' . Let

$$\begin{aligned} \bar{\mathbf{y}}(t) &= \begin{bmatrix} \mathbf{y}'(t) \\ \mathbf{y}''(t) \end{bmatrix} = \bar{\mathbf{A}} \mathbf{s}(t) + \mathbf{n}_y(t) \\ \bar{\mathbf{A}} &= \begin{bmatrix} \mathbf{A}' \\ \mathbf{A}'' \mathbf{\Phi} \end{bmatrix}, \quad \mathbf{n}_y(t) = \begin{bmatrix} \mathbf{n}_{y'}(t) \\ \mathbf{n}_{y''}(t) \end{bmatrix} \end{aligned} \quad (21)$$

The structure of $\bar{\mathbf{A}}$ is then exploited to obtain estimates of the diagonal elements of $\mathbf{\Phi}$.

We now decompose the signal subspace, \mathbf{V}_s , in the following way

$$\mathbf{V}_s = \begin{bmatrix} \mathbf{V}_{Y'} \\ \mathbf{V}_{Y''} \end{bmatrix} = \begin{bmatrix} \mathbf{A}' \mathbf{T} \\ \mathbf{A}'' \mathbf{\Phi} \mathbf{T} \end{bmatrix} \quad (22)$$

Here \mathbf{T} is a unique nonsingular matrix such that $\mathbf{V}_s = \bar{\mathbf{A}} \mathbf{T}$. Define the rank M matrix

$$\mathbf{V}_{Y'Y''} \stackrel{\text{def}}{=} [\mathbf{V}_{Y'} | \mathbf{V}_{Y''}] \quad (23)$$

This implies that there exists a unique rank M matrix $\mathbf{F} \in \mathcal{C}^{2M \times M}$ such that

$$\begin{aligned} \mathbf{0} &= [\mathbf{V}_{Y'} | \mathbf{V}_{Y''}] \mathbf{F} = \mathbf{V}_{Y'} \mathbf{F}_{Y'} + \mathbf{V}_{Y''} \mathbf{F}_{Y''} \\ &= \mathbf{A}' \mathbf{T} \mathbf{F}_{Y'} + \mathbf{A}'' \mathbf{\Phi} \mathbf{T} \mathbf{F}_{Y''} \end{aligned} \quad (24)$$

\mathbf{F} spans the null-space of $[\mathbf{V}_{Y'} | \mathbf{V}_{Y''}]$. We can now rearrange (24), by defining the matrix $\mathbf{\Psi}$.

$$\mathbf{\Psi} \stackrel{\text{def}}{=} -\mathbf{F}_{Y'} [\mathbf{F}_{Y''}]^{-1} \quad (25)$$

Eq. (24) now becomes,

$$\mathbf{A}' \mathbf{T} \mathbf{\Psi} = \mathbf{A}'' \mathbf{\Phi} \mathbf{T} \Rightarrow \mathbf{A}' \mathbf{T} \mathbf{\Psi} \mathbf{T}^{-1} = \mathbf{A}'' \mathbf{\Phi} \quad (26)$$

With \mathbf{A} of full rank, this implies

$$\mathbf{T} \mathbf{\Psi} \mathbf{T}^{-1} = \mathbf{\Phi} \quad (27)$$

Thus, the eigenvalues of $\mathbf{\Psi}$ equal the diagonal elements of $\mathbf{\Phi}$ ($\mathbf{\Psi}$ and $\mathbf{\Phi}$ are related through a similarity transformation). The columns of \mathbf{T} are the eigenvectors of $\mathbf{\Psi}$.

Now, expressing $\mathbf{\Phi}$ in (22) as (27) we obtain the following relation between $\mathbf{V}_{Y'}$ and $\mathbf{V}_{Y''}$

$$\mathbf{V}_{Y''} = \mathbf{V}_{Y'} \mathbf{\Psi} \quad (28)$$

We can now compute the eigenvalues of $\mathbf{\Psi}$ using either a least-square or a total-least-square criterion. The diagonal elements in $\mathbf{\Phi}$ will equal the eigenvalues, and by applying (15) to these, the DOAs are estimated.

V. EXPERIMENTS

A. Data

The methods were tested by using raw sonar data recorded at the array receive elements of Simrad EM3000 echo sounders. The transmit signal of this sonar is a rectangular pulse of a sinusoidal wave with pulse length 150 μs

and center frequency 300158 Hz. The receive array is a uniform linear array of 80 elements. The distance between two neighboring elements is 2.55 mm. The received signals are demodulated and separated into in-phase and quadrature components, and then sampled at 14293 Hz. The sampled data are represented as two bytes signed integers.

To evaluate the high-resolution beamformers, 7 ping series were recorded in shallow water near Horten, Norway. The sea bottom varies from flat to quite rough. The sonar head was tilted 45 degrees to the side for some ping series. The bottom depth was between 2 to 20 meters.

B. Evaluation Methods

The performance of a beamformer can be described by parameters such as beamwidth, bias, and variance. However, these parameters are usually difficult to measure when the bottom truth is unknown. The Full Width Half Maximum (FWHM) is commonly used to measure the beamwidth, but it says nothing about how accurate a beamformer estimates the DOA. If one considers the highest peaks in two different spectra, for the same sample, it is not enough to have a smaller FWHM (better resolution) to be regarded as the best result. The best result must also have its peak at the source's direction. Since we do not really know the *true* bottom in these experiments, we assume that the FT spectrum "tells" us what is reasonably close to the truth. Assume that the FWHM of a peak defines a beam. If a high-resolution method has a beam which is narrower than that of the FT beam, and in addition has its peak located within the area covered by the FT beam, this method will be considered as better than the FT. The following quantitative measures for assessing the performance of a beamformer have been developed.

Peak-Width Ratio (PWR) between the FT and a high-resolution beamformer (HR) is given by

$$\text{PWR}(M) = \frac{1}{\mathcal{N}} \sum_{n=1}^{\mathcal{N}} \frac{\sum_{m=1}^M \text{FWHM}_{\text{FT}}(n, m)}{\sum_{m=1}^M \text{FWHM}_{\text{HR}}(n, m)} \quad (29)$$

where n is an index for a sample and m an index for a peak. For each sample, the FWHM of the M highest peaks obtained by the FT are compared to those obtained by the high-resolution beamformer. The ratio is then averaged for all \mathcal{N} samples used for measuring the performance. If PWR is larger than one, the high-resolution method has a better resolution than the FT.

Width-Location Overlap (WLO) gives the percentage of the high-resolution method's beams which are located within the FT beams. It is given by

$$\text{WLO}(M) = \frac{1}{\mathcal{N}M} \sum_{n=1}^{\mathcal{N}} \sum_{j=1}^M \frac{\sum_{i=1}^M \text{BEAM}_{\text{HR}}(n, i) \cap \text{BEAM}_{\text{FT}}(n, j)}{\text{BEAM}_{\text{HR}}(n, i)} \quad (30)$$

This value is also averaged for all \mathcal{N} samples and all M beams in each sample.

PWR tells us how narrow the beams are, and WLO describes the localization of the peaks. Both of the measures have a parameter M which is the number of the peaks to be considered for each sample. The value of M should equal the number of signals. We computed $\text{PWR}(M)$ and $\text{WLO}(M)$ for estimated number of signals, and found that $\text{PWR}(M)$ and $\text{WLO}(M)$ were highly correlated to $\text{PWR}(1)$ and $\text{WLO}(1)$, respectively. (Only Minimum-Norm and root-MUSIC had much higher $\text{WLO}(M)$ than $\text{WLO}(1)$. It seems that the two methods choose the highest peak at random.) Therefore, the highest peak ($M = 1$) is used in our evaluation. In the rest of the paper, the parameter is omitted when $M = 1$. The two measures can be computed separately for early received samples consisting of specularly reflected signals and later received samples consisting of obliquely reflected signals. The values of these two time intervals were also highly correlated and are therefore averaged.

Computing time MATLAB was used to implement the algorithms and the computing time was measured by the floating point operation counter (`flops`). The average amount of Mega-flops per sample was measured.

C. Results

For each high-resolution beamformer, we tested different covariance estimation methods, i.e. forward/backward averaging, spatial smoothing with subarrays, and noise covariance subtraction. The experiments showed that forward/backward averaging always improved the results, more or less, and should therefore always be applied. For spatial smoothing, different beamformers required different number of subarrays. Noise covariance subtraction generally improved the results, but not always.

The quantitative measures were computed and averaged over 22 pings randomly chosen from the 7 available ping series.

Eigen-Vector generally gave high PWR values (2 to 3.5). The largest WLO was obtained for about 20 subarrays. The use of noise covariance subtraction slightly reduced the WLO values. The amount of Mega-flops dropped from 124 to 38 with increasing numbers of subarrays. For this method, we suggest to use 20 subarrays without noise covariance subtraction.

Minimum-Variance gave results which were very similar to those of Eigen-Vector. This is not surprising, since the Eigen-Vector algorithm is the same as the Minimum-Variance algorithm except using fewer eigenvectors.

MUSIC was not capable of achieving equally high PWR values as Eigen-Vector. Higher PWR values were obtained by increasing the number of subarrays. The PWR went

from 1.5 for no subarray to almost 3 for 29 subarrays. This is quite opposite from what would be the case for the FT, but can be explained by the fact that MUSIC's resolution increases with the amount of available data. For the MUSIC algorithm, use of subarrays seems to simulate an increase in available data. The WLO values were highest when using few subarrays. This supports Rao's statement in [7] "minimal smoothing appears to be desirable to obtain reliable DOA estimates". The WLO dropped from 85% to 71% when going from no subarray to 8 subarrays. Results generally improved when subtracting the noise covariance. Computing time was reduced by increasing the number of subarrays, from 80 Mega-flops for 2 subarrays, to 28 Mega-flops for 29 subarrays. Considering WLO as the most important measure, we suggest to use 2 subarrays with noise covariance subtraction for this beamformer.

Minimum-Norm had a high PWR (between 2.5 and 3.5), independent of the number of subarrays. WLO, however, depended upon the number of subarrays and the use of noise covariance subtraction. This value was normally very low when noise subtraction was not used. For few subarrays Minimum-Norm used about 137 Mega-flops, dropping to 51 for 29 subarrays. The best results were obtained by using 26 subarrays and noise subtraction. This is the suggested way of estimating the covariance matrix. The high number of subarrays is somewhat larger than what was stated for Minimum-Norm in [7].

Root-MUSIC gave poor results. The PWR measure is not applicable for this beamformer, since in a root-MUSIC spectrum, the peaks are just delta pulses with no width. Our test showed that the WLO values were low and the computing time was large. For this beamformer, we suggest to use 29 subarrays and a noise covariance subtraction.

ESPRIT was less dependent on the covariance estimation methods than the other beamformers. Still, forward/backward averaging and spatial smoothing improved the performance. The results were almost equal whether one applied the noise subtraction or not. Just as for the root-MUSIC beamformer, the PWR measure is not applicable to ESPRIT. The WLO values were highest for 11 to 29 subarrays. According to Rao [8], one can expect a decrease of WLO when using more than $(L+1)/5$ subarrays, which means 16 subarrays for the Simrad EM3000 sonar. This decrease was, however, only observed for specularly reflected signals when using more than 20 subarrays. ESPRIT was very efficient regarding computation. It used 14 Mega-flops for no subarray. The Mega-flops dropped to 7 for 29 subarrays. For this beamformer, we suggest to use 17 subarrays without any noise covariance subtraction.

The beamformers are compared for the suggested covariance matrix estimation methods in Table 1.

In addition to the quantitative measures, we present some spectra in Fig. 2, and examine the results visually.

Table 1: A comparison of beamforming methods.

Method	PWR	WLO	Mega-flops
FT	1.0	100%	19
Eigen-Vector	2.4	65%	52
MUSIC	1.6	84%	80
Minimum-Norm	3.1	51%	58
root-MUSIC	"∞"	22%	88
ESPRIT	"∞"	71%	9

We also apply a simple bottom detection algorithm to investigate if the spectra provide information for accurate bottom detection. For each direction, this bottom detection algorithm simply finds the time sample where the spectrum power is largest.

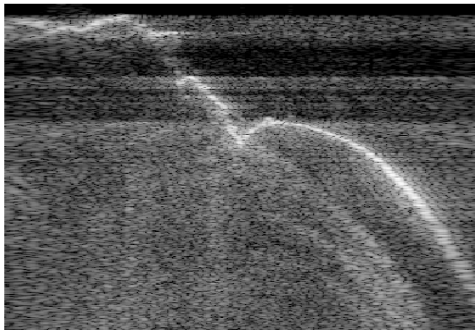
VI. DISCUSSION

An important issue in developing next generation high-performance multibeam echo sounders is to increase the mapping resolution. Much research has been done to improve the along-track resolution using synthetic aperture method (e.g. [13]). At the same time, modern high-resolution beamforming methods are interesting for improving the cross-track resolution.

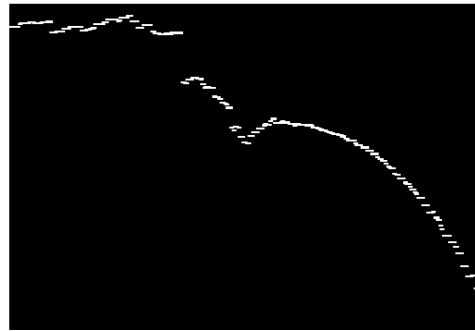
Using raw data from Simrad EM3000 multibeam echo sounders, we tested several high-resolution beamformers. Estimation of spatial covariance matrix plays an important role in high-resolution beamforming. In addition to forward/backward averaging and subarray smoothing, a noise covariance subtraction aimed to reduce structured noise was applied. For most of the high-resolution methods, this noise subtraction did improve the performance. Quantitative measures were developed and used as part of the evaluation. MUSIC and ESPRIT gave the best results according to the evaluation.

All the high-resolution beamformers reduced the beamwidth and improved the resolution. This can be seen from the PWR measures which were all larger than 1, and from Fig. 2 where MUSIC and ESPRIT gave narrower spectrum peaks than FT. The bias of the methods are not easy to estimate. We compared the location of peaks of FT spectra with those of the spectra of high-resolution beamformers. MUSIC and ESPRIT had the highest WLO values and are considered as the most reliable. ESPRIT is computationally efficient, while MUSIC is much more time demanding than FT. MUSIC has to compute the $L - M$ smallest eigenvectors, while ESPRIT only need the M largest ones. In our case, L , the number of sensors is 80, and M , the number of signals is typically 2 to 3. Fig. 2 shows that the high-resolution spectra have a higher noise level than the FT spectra. The simple bottom detection algorithm included more noise peaks for MUSIC and ESPRIT than for FT.

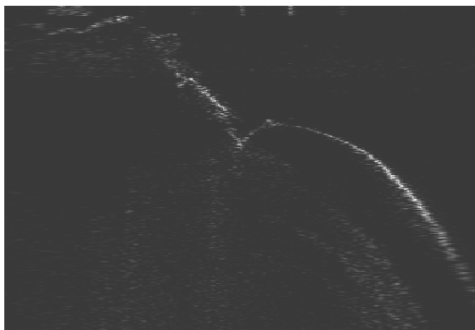
One problem when using high-resolution beamforming is that the signals are not stationary in time. Data avail-



(a) FT spectrum of a ping.



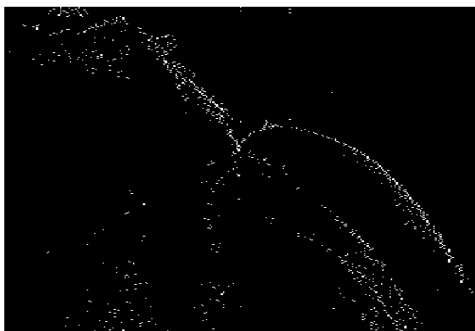
(b) Bottom detected from the FT spectrum.



(c) MUSIC spectrum. Forward/backward averaging applied. Smoothed with 2 subarrays. Noise covariances subtracted.



(d) Bottom detected from the MUSIC spectrum.



(e) ESPRIT spectrum. Forward/backward averaging applied. Smoothed with 17 subarrays.



(f) Bottom detected from the ESPRIT spectrum.

Fig. 2: Spectra for different beamformers (left) and the results of a simple bottom detection algorithm (right). The horizontal axis is the direction, from -70 to 70 degrees (angular increments: 0.5 degrees). The vertical axis is the sample number, from 250 on the top until 750 on the bottom. The sonar is tilted 45 degrees to the side.

able for estimating the covariance matrix is limited to only one sample. Increasing the sampling rate may provide a few stationary samples and thus improve the estimation of the covariance matrix. Another problem is the assumption that the signals arrive only from the y - z plane. This assumption is violated when the along-track resolution is low. It seems that the cross-track resolution and the along-track resolution should be increased simultaneously.

More research has to be done in order to apply high-resolution beamforming in a multibeam sonar product. Unlike conventional beamforming methods, high-resolution methods do not try to provide a “good spectrum shape”. Instead, they try to give narrow spectrum peaks for the signals to be detected. These spectra may be good for bottom detection but not necessarily for seabed imaging. Therefore, high-resolution beamformers should be used together with a conventional beamformer such as the FT beamformer. This is possible as the computing power is becoming cheaper. A rough bottom detection may then first be applied based on the FT spectrum and afterwards small objects can be detected using a high-resolution beamformer.

REFERENCES

- [1] C. de Moustier, State of the art in swath bathymetry survey systems, *Int. Hydr. Rev.* **65**(2): 25–54, 1988.
- [2] L. Yang, and T. Taxt, Multibeam sonar bottom detection using multiple subarrays In *Proc. OCEANS'97, Vol. II* pp. 932–938, 1997.
- [3] H. Krim, and M. Viberg, Two decades of array signal processing research: the parametric approach, *IEEE Signal Proc. Magazine* **13**(4): 67–94, 1996.
- [4] D. Pantzartzis, C. de Moustier and D. Alexandrou, Application of high-resolution beamforming to multibeam swath bathymetry, In *Proc. OCEANS'93 Vol. II* pp. 77–82, 1993.
- [5] D. Pantzartzis, D. Alexandrou and V. Premus, High-resolution bathymetric simulations based on Kirchhoff scattering theory and anisotropic seafloor modeling, In *Proc. Int. Conf. Acoustics, Speech and Signal Processing Vol. II* pp. 353–356, 1994.
- [6] K. K. Talukdar, Shallow water bathymetric data improvement in SeaBeam 2100 multibeam systems, In *Proc. Int. Symp. Underwater Technology*, pp. 299–303, 1998.
- [7] B. D. Rao, and K. V. S. Hari, Effect of spatial smoothing on the performance of MUSIC and the minimum-norm method, *IEE Proc. Pt. F* **137**(6): 449–458, 1990.
- [8] B. D. Rao, and K. V. S. Hari, Effect of spatial smoothing on state space methods/ESPRIT, In *IEEE ASSP 5th Workshop on Spectrum Estimation and Modeling* pp. 377–381, 1990.
- [9] J. Capon, High-resolution frequency-wavenumber spectrum analysis, *Proc. IEEE* **57**(8): 1408–1418, 1969.
- [10] D. H. Johnson, and D. E. Dudgeon, *Array Signal Processing, Concepts and Techniques*, Prentice Hall, 1993.
- [11] K. R. Srinivas, and R. V. Umaphathi, Finite data performance of MUSIC and Minimum Norm methods, *IEEE Trans. Aerosp. Electr. Sys.* **30**(1): 161–174, 1994.
- [12] R. Roy, and T. Kailath, ESPRIT—estimation of signal parameters via rotational invariance techniques, *IEEE Trans. Acoust. Speech Signal Process.* **37**(7): 984–995, 1989.
- [13] M. A. Pinto, Use of frequency and transmitter location diversities for ambiguity suppression in synthetic aperture sonar systems, In *Proc. OCEANS'97, Vol. I* pp. 363–368, 1997.

Bibliography

- [1] J. Augustin, R. Le Suave, X. Lurton, M. Voisset, S. Dugelay, and C. Satra. Contribution of the multibeam acoustic imagery to the exploration of the sea-bottom. *Marine Geophysical Researches*, 18:459–486, 1996.
- [2] K. M. Buckley and X.-L. Xu. Spatial-spectrum estimation in a location sector. *IEEE Trans. Acoustics Speech and Signal Processing*, 38(11):1842–1852, Nov. 1990.
- [3] C. de Moustier. State of the art in swath bathymetry survey systems. *International Hydro-graphic Review*, 2(65):25–54, July 1988.
- [4] B. Friedlander. The root-MUSIC algorithm for direction finding with interpolated arrays. *Signal Processing*, 30(1):15–29, Jan. 1993.
- [5] M. H. Hayes. *Statistical Digital Signal Processing and Modeling*. Wiley, 1996.
- [6] T.-P. Jäntti. Trials and experimental results of the ECHOS XD multibeam echosounder. *IEEE Journal of Oceanic Engineering*, 14(4):306–313, Oct. 1989.
- [7] D. H. Johnson and D. E. Dudgeon. *Array Signal Processing, concepts and techniques*. Signal Processing Series. Prentice Hall, 1993.
- [8] S. M. Kay. *Modern Spectral Estimation, theory and application*. Signal Processing Series. Prentice Hall, 1988.
- [9] P. H. Kraeutner, J. S. Bird, and S. Fraser. Principal components array processing for swath acoustic mapping. *OCEANS '97*, pages 1246–1254, 1997.
- [10] H. Krim, P. Forster, and J. G. Proakis. Operator approach to performance analysis of Root-MUSIC and Root-Min-Norm. *IEEE Trans. Signal Processing*, 40(7):1687–1695, July 1992.
- [11] H. Krim and M. Viberg. Two decades of array signal processing research: The parametric approach. *IEEE Signal Processing Magazine*, pages 67–94, July 1996.
- [12] R. Kumaresan and D. W. Tufts. Estimating the angles of arrival of multiple plane waves. *IEEE Trans. Aerospace and Electr. Syst.*, AES-19(1):134–139, Jan. 1983.
- [13] F. Li and Y. Lu. Unified bias analysis of subspace-based DOA algorithms. *Control and Dynamic Systems*, 77:149–192, 1996.
- [14] X. Lurton, S. Dugelay, and J. Augustin. Analysis of multibeam echosounder signals from the deep seafloor. In *OCEANS '94*, volume 2, pages 213–218, 1994.

- [15] T. Lyche. *Numerical Linear Algebra*. University of Oslo, 1992.
- [16] A. V. Oppenheim and R. W. Schaffer. *Discrete-Time Signal Processing*. Signal Processing Series. Prentice Hall, 1989.
- [17] B. Ottersten, M. Viberg, and T. Kailath. Analysis of subspace fitting and ML techniques for parameter estimation from sensor array data. *IEEE Trans. Signal Processing*, 40(3):590–599, Mar. 1992.
- [18] D. Pantzartzis, D. Alexandrou, and V. Premus. High-resolution bathymetric simulations based on kirrchhoff scattering theory and anisotropic seafloor modelling. In *IEEE Int. Conf. Acoust., Speech, Sign. Proc.*, volume 2, pages 353–356, 1994.
- [19] D. Pantzartzis, C. de Moustier, and D. Alexandrou. Application of high-resolution beamforming to multibeam swath bathymetry. In *OCEANS '93*, volume 2, pages 77–82, 1993.
- [20] M. A. Pinto. Use of frequency and transmitter location diversities for ambiguity suppression in synthetic aperture sonar systems. In *OCEANS '97*, volume 1, pages 363–368, 1997.
- [21] F. Pøhner. Current North Sea technology for offshore mapping and inspection of the seabed. *Offshore Technology Conference*, 1998.
- [22] B. D. Rao and K. V. S. Hari. Performance analysis of root-MUSIC. *IEEE Trans. Acoustics Speech and Signal Processing*, 37(12):1939–1949, Dec. 1989.
- [23] B. D. Rao and K. V. S. Hari. Effect of spatial smoothing on state space methods/ESPRIT. *IEEE ASSP 5th Workshop Spectrum Est. Modeling*, pages 377–381, Oct. 1990.
- [24] B. D. Rao and K. V. S. Hari. Effect of spatial smoothing on the performance of MUSIC and the minimum-norm method. *IEE Proceedings*, 137, Pt. F(6):449–458, Dec. 1990.
- [25] A. Rønhovde, L. Yang, T. Taxt, and S. Holm. High-resolution beamforming for multibeam echo sounders using raw EM3000 data. In *OCEANS '99*, volume 2, pages 923–930, Sept. 1999.
- [26] R. Roy and T. Kailath. ESPRIT- Estimation of Signal Parameters via Rotational Invariance Techniques. *IEEE Trans. Acoustics Speech and Signal Processing*, 37(7):984–995, July 1989.
- [27] K. R. Srinivas and R. V. Umamathi. Finite data performance of MUSIC and Minimum Norm methods. *IEEE Trans. Aerospace and Electr. Syst.*, 30(1):161–174, Jan. 1994.
- [28] A. O. Steinhardt and B. D. van Veen. Adaptive beamforming. *International Journal of Adaptive Control and Signal Processing*, 3:253–281, 1989.

-
- [29] K. K. Talukdar. Shallow water bathymetric data improvement in Seabeam 2100 multibeam systems. In *Proc. Int. Symp. Underwater Technology*, pages 299–303, 1998.
- [30] J. N. Wright. Image formation in diagnostic ultrasound. In *IEEE International Ultrasonic Symposium Short Course*, pages 1–22, 1997.
- [31] X.-L. Xu and K. M. Buckley. Bias and variance of direction-of-arrival estimates from MUSIC, MIN-NORM, and FINE. *IEEE Trans. Signal Processing*, 42(7):1812–1816, July 1994.
- [32] L. Yang and T. Taxt. Multibeam sonar bottom detection using multiple subarrays. In *Proc. OCEANS '97*, volume 2, pages 932–938, 1997.
- [33] M. D. Zoltowski, G. M. Kautz, and S. D. Silverstein. Beamspace root-MUSIC. *IEEE Trans. Signal Processing*, 41(1):344–364, Jan. 1993.