A Framework to Resolve Ambiguities in a Multitarget Environment

by

Jurgen Wervers

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Tuesday July 4, 2023 at 13:00.

Student number:	4599136	
Project duration:	September 15, 2022 – July 4, 2023	
Thesis committee:	Dr. ir. J. N. Driessen,	TU Delft
	Dr. G. Joseph,	TU Delft
	Dr. F. Fioranelli,	TU Delft

TU Delft

Abstract

Ambiguities are an often encountered nuisance in signal processing and are the source of some of the fundamental trade-offs encountered in radar systems. The goal of this thesis is to extract unambiguous information about targets by combining a limited amount of measurements on a video integration level. A novel framework is proposed to reach this goal. At the heart of the framework lives a relevance vector machine which is extended to process the ambiguities on a video integration level and to work off-grid. The relevance vector machine is then extended to become the ambiguity aware relevance vector machine. This extension is either performed by a frequentist test or by estimating a posterior distribution. The frequentist test is used to test whether we can statistically significantly discern the returned output from ambiguities. The posterior is estimated according to Bayes' theorem and thus allows for the incorporation of prior information. In this thesis, the framework is specifically applied to Doppler processing of a pulse-Doppler radar system. Compared to existing methods for estimating unambiguous Doppler velocity in a multi-target environment, the framework provides a general increase in performance, allows for the incorporation of prior information, and is able to give a measure of confidence in the estimates. A simulation study is set up to show the performance increase. This simulation study also highlights the utility of incorporating prior information and the quantification of uncertainty.

Preface

Before you lies the thesis report that marks the end of my journey to obtain the degree of Master of Science in Electrical Engineering at the Delft University of Technology. I have written the thesis at the Microwave Sensing, Signals and Systems (MS3) group, where numerous resources, guidance, and support were available.

The topic of my thesis allowed me to be on the edge of two of my interests: electrical engineering and statistical modelling. The exploration of the topic allowed for a continuous pursuit of knowledge, directing me towards an investigation into topics within Bayesian statistics.

Finally, I would like to express my gratitude to those who supported me throughout the process.

- Hans, for guidance towards a topic that fits perfectly with my combined background as an engineer and econometrician. Also for the advice during the whole process.
- Francesco and Geethu, for taking the time and effort to be part of my thesis committee.
- Family and friends, for always being there and expressing sincere interest in the process.
- Veerle, for your continuous love and support.
- Ze-Sheng, for our tennis sessions. We challenged each other and progressed together on the court.
- Jippe, for being a friend and a great sparring partner on topics ranging from statistics and academic writing to career orientation.
- Bob, for your companionship in our shared office.
- Marien, Patrick, and Florens, for your cordial company during our daily lunch break.
- My eight roommates, for ensuring that there is always something happening at home.

Jurgen Wervers
Delft, June 2023

Contents

1 Introduction

Ambiguities are an often encountered nuisance in signal processing. In pulse-Doppler radars, we have the trade-off between unambiguous range and unambiguous Doppler. This trade-off is seen as a fundamental trade-off in the design of a pulse Doppler radar system [1, 2, 3]. In the estimation of direction of arrival (DOA) in a linear array, it is possible to encounter ambiguities as well. When the spacing between the sensors in the array is greater than half a wavelength, ambiguities will appear in the DOA estimation [4].

The principal motivation for a framework that is able to resolve ambiguities in a multi-target environment is to increase the performance of signal-processing algorithms. Especially in tracking algorithms, incorrect detections caused by ambiguities can easily lead to unreliable, missed, or late tracks.

1.1. Problem

Theoretically, we can resolve ambiguities by changing the ambiguity fold in each measurement or set of measurements. This shift of the ambiguity fold causes the ambiguities to shift as well, only the real target will stay in the same place across all the measurements [1, 5]. However, in practice, we often have a limited number of measurements. This limited number of measurements implies a limited amount of shifts of the ambiguity fold. Thus, we cannot resolve the ambiguity completely. We can however increase the unambiguous domain, except for the special case where the ambiguity folds are co-primes of each other.

Some factors make this theoretical solution more difficult. An example is target fluctuation from measurement to measurement. Target fluctuation implies that the received amplitude will vary from measurement to measurement, which makes it harder to match targets across multiple bursts with each other to resolve the ambiguities. Another factor complicating the solution is an unknown number of targets. When dealing with a single target, we can use standard techniques such as a Neyman-Pearson detector. In the case of multiple targets, especially when the exact number of targets is unknown, the problem becomes more complicated. The combination of an unknown amount of targets and target fluctuation makes the problem even more difficult to solve, and classical techniques do not provide us with a measure of how certain a given estimate is.

The specific problem is to accurately obtain information about the actual targets from the combination of a limited amount of measurements.

1.2. Prior art

There exists a selection of research articles concerning methods to resolve ambiguities. Most of the articles are specifically written with the application of radar in mind, as ambiguities in radar are inherent to pulse Doppler radar systems [3]. The existing methods can roughly be separated into two main classes: the hit-based coincidence methods and the particle-based methods. There are some methods that fall outside of these two classes.

The first class of algorithms are the hit-based coincidence type of algorithm. One of the earliest methods to resolve ambiguities is described in the first edition of the book of Skolnik [6]. The method of Skolnik is designed for a single target and based on pre-detected hits. The method is based on the Chinese remainder theorem. The Chinese remainder theorem provides a solution to two modular arithmetics with coprime moduli. The ambiguity folds thus need to be coprime of each other. Hovanessian [5] proposed a different method, claiming the method to be simpler to interpret and easier to implement compared to the method of Skolnik. The Hovanessian method does, however, suffer from the same drawbacks as the method of Skolnik. I.e., the ambiguity folds need to be coprimes of each other, the method is hit based, and assumes a single target. Reddy and Swamy [7] propose a coincidence type of algorithm that tackles some of the drawbacks of the methods of Skolnik and Hovanessian. The method no longer requires the ambiguity folds to be coprime with each other. The method of Reddy and Swamy also allows for multiple targets. The method is however still based on pre-detected hits and requires a minimum of four measurements with different ambiguity folds. The last method in the class of hit-based coincidence algorithms is the clustering algorithm of Trunk and Kim [8]. This method allows for more flexibility as it heuristically searches for clusters of hits, allowing for greater measurement errors.

The second class of algorithms is the set of particle-based algorithms. E.g., Cai *et al.* [9] and Bocquel *et al.* [10] propose tracking methods based on sequential Monte Carlo implementations that are able to resolve ambiguities. Due to the fact that these methods are tracking methods, they are designed to take information from past scans into account. Within the class of particle-based methods, there also exists a selection of research articles describing a method to retrieve unambiguous Doppler velocity estimates within a single burst by exploiting the range migration phenomenon [11, 12, 13, 14, 15]. These methods use Markov chain Monte Carlo (MCMC) samplers which are generally known to take a very long time to converge and do not have clear convergence measures. These drawbacks are addressed by [16] and [17] by using variational Bayesian inference instead of Markov chain Monte Carlo samplers. These methods are still only applicable when using a wideband radar that is able to observe range migration within a single burst.

An interesting approach that does not fit into one of the aforementioned classes, is the method of Shaban and Richards [18]. This method tries to reconstruct the scene of pre-processed hits. To prevent a lot of spurious components, Shaban and Richards suggest using $\mathcal{L}_1$ regularisation methods such as basis pursuit [19] or lasso [20].

Finally, we have the Feedback N-signal Orthogonal Matching Pursuit (FN-OMP) of Aouchiche *et al.* [21]. The FN-OMP method is the only method that provides a solution to the multi-target ambiguity problem on a video integration level. This method is chosen as a benchmark because this is the only method that works on a video integration level and is not a particle-based algorithm. The drawbacks of this specific method are that it cannot take into account prior information, it does not give a measure of confidence in the output, and it is based on OMP which is a greedy algorithm. Greedy algorithms tend to get stuck in local optima.

1.3. Research scope

The high-level goal is to formulate a framework that is able to resolve ambiguities in a multi-target environment. The framework should be able to incorporate prior information and quantify the uncertainty in estimates. In addition to that, it would be desirable that the framework has a clear convergence criterion. It would also be nice to have a framework that works on a video integration level, as processing on video integration level is known to increase detection performance as shown by, e.g., van Genderen and Meijer [22].

For feasibility, the goal of the research is restricted in three ways: the framework is formulated for white Gaussian noise environments, the tests of the framework are restricted to simulated data, and we specifically apply the framework to Doppler ambiguities. However, we stress that the framework is written down in probabilistic form and can thus be altered to accommodate other types of noise as well. The framework can also be applied to resolve ambiguities in other areas, such as DOA or range, with slight modifications.

We accomplish the goal of the research by answering the following research question, alongside three sub-questions:

- How can we formulate a framework that improves the state-of-the-art when it comes to resolving ambiguities in a multi-target environment when processing on a video level?
  - What model or statistical technique should drive the framework?
  - How can we incorporate a priori information?
  - How can we quantify the uncertainty in the estimates of the framework?

1.4. Main results and outline

In this thesis, we present a framework to resolve ambiguities in a multi-target environment. The framework is built up and analysed in five steps. Each of these steps corresponds to one of the chapters of the thesis. The five individual steps are written down below together with the main results of each chapter.

Introducing Doppler ambiguities and modelling with multiple measurement vectors is the first chapter. In this chapter, we introduce Doppler ambiguities in pulse Doppler radars systems. An introduction to Doppler ambiguities is desired, as we apply the framework to resolve Doppler ambiguities throughout the thesis. Next, we introduce the general form of signal processing problems with multiple measurement vectors. As mentioned in section 1.1, the problem is to obtain information on the actual targets by exploiting the coaction between multiple measurement vectors. This chapter provides insight into the assumptions that should be made to obtain this information when combining multiple measurement vectors. The chapter highlights the differences in assumptions that should be made when processing coherently compared to processing on a non-coherent video integration level.

An introduction to the relevance vector machine is the second step. In this chapter, we argue that the relevance vector machine is highly suitable to drive the framework. The relevance vector machine has strong parallels to the Swerling fluctuation model. The relevance vector machine also has highly desirable statistical properties, such as a global optimum at the maximally sparse solution and fewer local optima compared to competing methods. We then go on to derive the relevance vector machine for complex numbered problems and extend it to multiple measurement vectors, for both the coherent case as well as the non-coherent video integration level case.

Addressing the off-grid problem of the relevance vector machine is the third step. The relevance vector machine is an on-the-grid method, meaning that it is assumed that targets are located on predefined gridpoints. This assumption cannot hold in practice and leads to unsatisfactory performance. This chapter proposes two solutions to this problem. The first proposed solution is to extend the method of Dai *et al.* [23] to multiple measurement vectors. This extension is proposed in accordance with the assumptions made that are necessary to perform video-level processing. The second proposed solution is to use the VALSE algorithm of Badiu *et al.* [24] and pre-determine points of interest on the grid.

Extension to ambiguity aware relevance vector machine and incorporation of prior information is the final step to build the framework. In this chapter we explore methods to extend the relevance vector machine to the ambiguity aware relevance vector machine. We propose a frequentist test to test whether we can statistically significantly discern the returned relevance vector from possible ambiguities. We then take a Bayesian perspective and propose a way to generate a posterior distribution on each of the relevance vectors returned by the relevance vector machine. Since we handle the problem from a Bayesian point of view, we can take prior information into account. As this method provides a posterior, we get access to a measure to evaluate the confidence of the estimates of the framework.

The simulation study is the final chapter. A simulation study is performed to test the framework and compare it with existing methods. The framework tends to outperform the existing methods in general. We go on to show that the framework is more robust compared to the existing methods when it comes

to performance in particularly difficult cases. We then show that taking prior information into account increases the performance of the framework. Finally, we show that we can quantify the confidence in the estimates. By means of a qualitative comparison, we show that the framework comes close to an optimal MCMC sampler with respect to the quantification of confidence in the estimates.

2

Background on Doppler ambiguities and modelling with multiple measurement vectors

This chapter provides background information on Doppler ambiguities in pulsed Doppler radar systems and problems with multiple measurement vectors. Background information on Doppler ambiguities in pulsed Doppler radar systems is needed since the framework is specifically applied to Doppler processing in pulsed Doppler radar systems in this thesis. An introduction to problems with multiple measurement vectors is needed, as the problem statement of section 1.1 implies that the problem will become a multiple measurement vectors problem. This chapter specifically highlights the different assumptions that can be made concerning the measurement equations when working with multiple measurement vector problems.

2.1. Pulse Doppler radar basics

A radar system transmits electromagnetic waves toward some region of interest and then tries to detect reflections from objects in the region of interest.

The framework proposed in this thesis is specifically applied to pulsed Doppler radars. Doppler radar is a general term used to refer to any radar that uses the Doppler effect in any form. Pulsed refers to the practice of sending electromagnetic waves in specific short time windows. There are generally two types of Doppler radars, i.e., continuous wave and pulsed radars. Continuous wave radars continually transmit a signal, while the receiver is continuously receiving at the same time.

Continuous wave radars measure range with some sort of modulation on the signal. The modulation could be, e.g., frequency modulation. In frequency modulation we change the transmit frequency over time, essentially putting a timestamp on the transmitted signal. Continuous wave radars then measure Doppler by measuring phase change in outgoing and received signals. Due to the fact that continuous wave radars send and receive at the same time, there is a significant amount of transmitter leakage into the receiver. This transmitter leakage generally causes continuous wave radars to be restricted to the use of low transmit powers and therefore restricting the continuous wave radars to short-range applications. Due to the limitations of continuous wave radar systems, applications of such systems are usually simple, such as speed timing radars and altimeters [1]. As pulsed radar systems are capable of transmitting high signal power for a short amount of time and then switch off the transmitter to enable the receiver, they are more suitable for long-range applications. This causes pulsed Doppler radars to be most suitable for applications such as threat detection, ground-based, airborne and space-borne surveillance, as well as meteorological applications. Such applications do however require more complex systems compared to the usual applications of continuous wave radars [3].

As mentioned above, pulsed Doppler radars work by transmitting electromagnetic waves in short time

intervals. During the transmit time, the receiver is switched off to protect the receiver from the powerful signal of the transmitter. One of the key parameters to set here is the pulse repetition time (PRT).

2.1.1. Doppler ambiguities

The Doppler effect describes the changes of the electromagnetic waves when reflected off an object that is moving relative to the radar. The Doppler frequency f_d denotes the difference in frequency of the received and transmitted electromagnetic waves. The Doppler frequency f_d is given by

$$f_d = \frac{2v_r}{\lambda}, \tag{2.1}$$

where v_r and λ are the radial component of the velocity of the object relative to the radar and the wavelength of the transmitted signal, respectively. As the Doppler frequency is defined as the received frequency minus the transmitted frequency, we expect a positive shift when the object is moving towards the radar. The radial velocity is thus defined to be positive for objects moving towards the radar and negative for objects moving away from the radar.

Choosing a higher PRT will result in a higher unambiguous range. However, by increasing the PRT, the effective sample rate of the Doppler frequency is decreased. When using a pulsed radar, the Doppler frequency is sampled at each pulse. If we want to avoid aliasing, we should adhere to the Nyquist-Shannon sampling criterion. The Nyquist-Shannon sampling criterion states that the maximum frequency that can unambiguously be measured is half the sampling rate. Since the radar is sampling at a rate of $\frac{1}{\text{PRT}}$, we can write the maximum unambiguous Doppler shift as

$$f_{d,\text{Unambiguous}} = \pm\frac{1}{2\text{PRT}}. \tag{2.2}$$

Resulting in an ambiguity fold of

$$v_{r,\text{Fold}} = \frac{\lambda}{2\text{PRT}}. \tag{2.3}$$

2.2. Modelling with multiple measurement vectors

As mentioned in the problem statement of section 1.1, the problem is to get the information about the actual targets by exploiting the coaction between multiple measurement vectors. The problem will therefore become a multiple measurement vectors problem. In problems with multiple measurements, there are four main ways to jointly process the information gathered in the multiple vectors.

With the goal of processing on a video integration level in mind, we consider the following set of B measurement equations

$$\begin{aligned} \mathbf{t}_1 &= \mathbf{\Phi}_1\mathbf{w}_1 + \boldsymbol{\varepsilon}_1 \\ \mathbf{t}_2 &= \mathbf{\Phi}_2\mathbf{w}_2 + \boldsymbol{\varepsilon}_2 \\ &\vdots \\ \mathbf{t}_B &= \mathbf{\Phi}_B\mathbf{w}_B + \boldsymbol{\varepsilon}_B, \end{aligned} \tag{2.4}$$

where $\mathbf{t}_b \in \mathbb{C}^N$, $\mathbf{\Phi}_b \in \mathbb{C}^{N\times M}$ and $\mathbf{w}_b \in \mathbb{C}^M$, $\forall b \in \{1, 2, \dots, B\}$. The error term is denoted by $\boldsymbol{\varepsilon}_b$, $b \in \{1, 2, \dots, B\}$. There are typically four assumptions we can make on the structure of the problem and how to process all the measurement vectors jointly. An overview of these assumptions with the corresponding implications for Doppler processing is given in Table 2.1. The cases are worked out in more detail below.

Table 2.1: Tabular representation of the assumptions and their implications when applied to Doppler processing.

	$\mathbf{w}_i = \mathbf{w}_j$	$\mathbf{w}_i \neq \mathbf{w}_j$
$\mathbf{\Phi}_i = \mathbf{\Phi}_j$	Coherent processing. Do not expect changes in design matrix. Is unable to resolve ambiguities. I.e., constant carrier frequency and PRT.	Non-coherent processing. Do not expect changes in design matrix. Is unable to resolve ambiguities. E.g., varying carrier frequency in the exact proportion as PRT.
$\mathbf{\Phi}_i \neq \mathbf{\Phi}_j$	Coherent processing. Do expect changes in design matrix. Is able to resolve ambiguities. E.g., constant carrier frequency, varying PRT	Non-coherent processing. Do expect changes in design matrix. Is able to resolve ambiguities. E.g., varying carrier frequency and PRT independently of each other.

1. $\mathbf{\Phi}_i = \mathbf{\Phi}_j$ **and** $\mathbf{w}_i = \mathbf{w}_j \quad \forall i, j \in \{1, 2, \dots, B\}$. The joint measurement equation then becomes

$$\begin{bmatrix} \mathbf{t}_1 \\ \mathbf{t}_2 \\ \vdots \\ \mathbf{t}_B \end{bmatrix} = \begin{bmatrix} \mathbf{\Phi} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{\Phi} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{\Phi} \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ \mathbf{w} \\ \vdots \\ \mathbf{w} \end{bmatrix} + \boldsymbol{\varepsilon} \tag{2.5}$$

such an approach is only useful in the case that $\mathbb{E}[\mathbf{t}_i] = \mathbb{E}[\mathbf{t}_j]$, $\forall i, j \in \{1, 2, \dots, B\}$ and we thus have multiple realisations that are, apart from the noise, exactly the same.

In radar terms, this way of processing corresponds to an uneventful case. When using such a measurement model, the assumption is made that the situation being measured is exactly the same across all measurements. We thus do not expect any change in waveform, or target responses.

2. $\mathbf{\Phi}_i = \mathbf{\Phi}_j$ **and** $\mathbf{w}_i \neq \mathbf{w}_j \quad \forall i, j \in \{1, 2, \dots, B\}$. The joint measurement equation then becomes

$$\begin{bmatrix} \mathbf{t}_1 \\ \mathbf{t}_2 \\ \vdots \\ \mathbf{t}_B \end{bmatrix} = \begin{bmatrix} \mathbf{\Phi} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{\Phi} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{\Phi} \end{bmatrix} \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \vdots \\ \mathbf{w}_B \end{bmatrix} + \boldsymbol{\varepsilon}, \tag{2.6}$$

or

$$\left[\mathbf{t}_1, \mathbf{t}_2, \dots, \mathbf{t}_B\right] = \mathbf{\Phi} \left[\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_B\right] + \boldsymbol{\varepsilon}. \tag{2.7}$$

This means that we have different realisations of the same process. This is the approach that is most often used in literature to process multiple measurement vectors, e.g., in [25, 26, 27]. This however will not be able to resolve ambiguities. To resolve ambiguities, we need to measure with a varying ambiguity fold. Therefore, we need to take into account that the matrix $\mathbf{\Phi}$ is different across different measurement vectors.

This type of processing is used in multi-input and multi-output (MIMO) communication systems and biomedical signal processing. These assumptions can be used to model fixed capacity regions of MIMO communication channels with varying demand [26]. The assumptions are used in biomedical signal processing when modelling the brain. E.g., when the assumption is made that the variation in brain activity is such that while the activation magnitudes change, the activation sites themselves do not [25].

The combination of these assumptions is uncommon in Doppler processing. One example where this combination will occur is when the carrier frequency is changed from burst to burst and the PRT is also changed from burst to burst with the same ratio. The design matrices $\mathbf{\Phi}_b$ will then look exactly the same over the different bursts. The amplitudes will however vary as, except for a perfect sphere, the radar cross section of an object is dependent on the wavelength of the incidence waves [28, 29].

3. $\mathbf{\Phi}_i \neq \mathbf{\Phi}_j$ **and** $\mathbf{w}_i = \mathbf{w}_j \quad \forall i, j \in \{1, 2, \dots, B\}$. The joint measurement equation then becomes

$$\begin{bmatrix} \mathbf{t}_1 \\ \mathbf{t}_2 \\ \vdots \\ \mathbf{t}_B \end{bmatrix} = \begin{bmatrix} \mathbf{\Phi}_1 \\ \mathbf{\Phi}_2 \\ \vdots \\ \mathbf{\Phi}_B \end{bmatrix} \mathbf{w} + \boldsymbol{\varepsilon} \tag{2.8}$$

Here, we essentially model that we have consistent realisation over different processes. This corresponds to coherent processing, we assume that the coefficients stay the same over the individual bursts, but we allow the process itself to vary over the individual bursts.

This situation is encountered in Doppler processing when the PRT is changed from burst-to-burst, but the carrier frequency stays the same. Additionally, the assumption should hold that the objects do not rotate with respect to the radar between the bursts. The design matrices $\mathbf{\Phi}_b$ will then vary from burst to burst, but the target response amplitudes should be constant.

4. $\mathbf{\Phi}_i \neq \mathbf{\Phi}_j$ **and** $\mathbf{w}_i \neq \mathbf{w}_j \quad \forall i, j \in \{1, 2, \dots, B\}$. The joint measurement equation then becomes

$$\begin{bmatrix} \mathbf{t}_1 \\ \mathbf{t}_2 \\ \vdots \\ \mathbf{t}_B \end{bmatrix} = \begin{bmatrix} \mathbf{\Phi}_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{\Phi}_2 & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{\Phi}_B \end{bmatrix} \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \vdots \\ \mathbf{w}_B \end{bmatrix} + \boldsymbol{\varepsilon}, \tag{2.9}$$

Assuming independent error terms, this formulation will result in processing all measurements independently. To actually process the information jointly, we impose a joint prior on the weights of each individual measurement vector, e.g.

$$\begin{aligned} \mathbf{w}_1 &\sim \mathcal{CN}(\mathbf{0}, \mathbf{\Sigma}) \\ \mathbf{w}_2 &\sim \mathcal{CN}(\mathbf{0}, \mathbf{\Sigma}) \\ &\vdots \\ \mathbf{w}_B &\sim \mathcal{CN}(\mathbf{0}, \mathbf{\Sigma}), \end{aligned} \tag{2.10}$$

where the dependence is thus imposed by the prior on the weights, $\mathbf{w}$. Specifically, by the joint covariance matrix $\mathbf{\Sigma}$.

This set of assumptions corresponds to incoherent processing and is the most comprehensive for Doppler processing. This combination of assumptions allows for both the variation in the design matrices $\mathbf{\Phi}_b$, as well as variation in target response amplitudes. Allowing for changes in design matrices is necessary when changing, e.g., the carrier frequency and the PRT across bursts. The assumptions now give freedom to the response amplitudes of the targets. This freedom is necessary when, e.g., the carrier frequency is varied across bursts or when the objects rotate with respect to the radar in between bursts.

3

The relevance vector machine

The relevance vector machine will be the main driver of the framework. In this chapter, we argue why the relevance vector machine is suitable to drive the framework, show how the relevance vector machine can be derived for complex-valued problems, and introduce multiple measurement vector relevance vector machines.

The relevance vector machine is first introduced by Tipping [30] as a Bayesian treatment to solve generalised linear models of the functional form given by

$$y(\mathbf{x};\mathbf{w}) = \sum_{i=1}^{M-1} w_i K\left(\mathbf{x},\mathbf{x}_i\right) + w_0, \tag{3.1}$$

where $K\left(\mathbf{x},\mathbf{x}_i\right)$ is a kernel function, in the context of relevance vector machines treated as a set of basis functions. This treatment of the kernel function as basis functions makes the extension from the kernel function to general basis functions straightforward. The weight of each basis function is denoted by w_i, $i \in \{0,1,2,3,\dots,M-1\}$.

Tipping proposes a method to solve the form based on the work in automatic relevance determination of MacKay [31] and Neal [32]. The solution is derived to exhibit a couple of desirable characteristics. First of all, the relevance vector machine is derived to be a fully Bayesian solution for (3.1). Second, the Bayesian treatment of the problem allows us to estimate the parameters and hyperparameters directly from the data without some form of hyperparameter tuning. Third, the Bayesian treatment of the problem results in many of the posterior distributions for the weights spiked at zero. The weights that do not have a posterior distribution spiked at zero are called relevance vectors as an homage to automatic relevance determination.

In the relevance vector machine, we pose a parameterised Gaussian distributed prior on the weights. A Parameterised Gaussian is not known to lead to sparse solutions. However, in the relevance vector machine this parameterised Gaussian prior does, surprisingly, lead to sparse solutions. In [33], Tipping shows that by posing a gamma hyperprior and integrating out the hyperparameter from the conditional prior on the weights, the prior on the weights becomes a multivariate Student's-t distribution. The multivariate Student's-t distribution is a heavy-tailed distribution with a sharp peak on zero, such prior distributions are generally known to induce sparsity. By casting the relevance vector machine in a variational framework, Wipf and Rao [34] provide a rigorous proof that the complete data likelihood in combination with the shape of the heavy-tailed prior distribution of the weights induce solutions with minimum variance along most weights. These weights are the weights that are pruned by the relevance vector machine, causing the sparse solution.

The parameterised Gaussian distributed prior posed on the weights causes the relevance vector machine to have strong parallels to cases one and two of the Swerling fluctuation model. These parallels are especially desirable, as the application of the framework within this thesis is in Doppler processing

of pulsed Doppler radars. The exact connection between the Swerling cases and a complex Gaussian distribution on the weights is worked out in Appendix D.

Wipf and Rao [34] also show that the relevance vector machine has the same global minimum as an ℓ_0-norm optimisation problem, formulated as

$$\min \|\mathbf{x}\|_0 \quad \text{s.t.} \quad \mathbf{\Phi x} = \mathbf{y}. \tag{3.2}$$

This global minimum being equal to the ℓ_0-norm problem is in contrast to the popular basis pursuit framework of Chen *et al.* [19], which solves the following problem

$$\min \|\mathbf{x}\|_1 \quad \text{s.t.} \quad \mathbf{\Phi x} = \mathbf{y}. \tag{3.3}$$

The basis pursuit algorithm only has the same global optimum as the ℓ_0-norm problem under certain assumptions on the sparsity level $\|\mathbf{x}\|_0$ and the basis vector matrix $\mathbf{\Phi}$ [35].

Wipf and Rao [34] also show that the relevance vector machine suffers from fewer local minima compared to some widely used non-convex relaxations. These non-convex relaxations include the iterative least-squares scheme known as focal underdetermined system solver as described by [36], [37] and the reweighted ℓ_0 minimisation approach of Candes *et al.* [38].

3.1. The complex-valued relevance vector machine

In radar problems, we work with complex-valued data. The relevance vector machine in our framework should therefore be able to process complex-valued data. For a given input-target pair $\{\mathbf{x}_n, t_n\}$, $n \in \{1, 2, 3, \ldots, N\}$, the relevance vector machine of Tipping [30, 33] has a model specification given by

$$t_n = y(\mathbf{x}_n; \mathbf{w}) + \varepsilon_n. \tag{3.4}$$

Which we rewrite to the following canonical form

$$\mathbf{t} = \mathbf{\Phi w} + \boldsymbol{\varepsilon}, \tag{3.5}$$

where $\mathbf{t}$ is the target vector, $\mathbf{\Phi}$ is the matrix containing the basis functions, $\mathbf{w}$ is a vector containing weights and $\boldsymbol{\varepsilon}$ is a vector containing real-valued independent zero mean noise variables. I.e., $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$, where $\mathbf{0}$ is a zero vector and $\mathbf{I}$ is the identity matrix. The solution is derived based under the assumptions that in the canonical form of the problem, all values are real, i.e $\mathbf{t} \in \mathbb{R}^N$, $\mathbf{\Phi} \in \mathbb{R}^{N\times M}$, and $\mathbf{w} \in \mathbb{R}^M$.

Here we give a version of the relevance vector machine that drops the assumption of all values being real and assumes $\mathbf{t} \in \mathbb{C}^N$, $\mathbf{\Phi} \in \mathbb{C}^{N\times M}$, and $\mathbf{w} \in \mathbb{C}^M$. The noise term $\boldsymbol{\varepsilon}$ is also modelled as an independent zero-mean complex Gaussian random variable. I.e., $\boldsymbol{\varepsilon} \sim \mathcal{CN}(\mathbf{0}, \sigma^2\mathbf{I})$. Whenever we refer to complex Gaussian random variables in this thesis, we implicitly refer to circularly symmetric complex Gaussian random variables. The definition of the circularly symmetric complex Gaussian random variable and the corresponding properties and theorems are given in Kay [39, Ch. 15]. Taking the complex Gaussian noise term into account, we write

$$p(t_n|y(\mathbf{x}_n; \mathbf{w}), \sigma^2) = \mathcal{CN}(t_n|y(\mathbf{x}_n; \mathbf{w}), \sigma^2) \tag{3.6}$$

Hence, the complete data likelihood is written as

$$p\left(\mathbf{t} \mid \mathbf{w}, \sigma^2\right) = \mathcal{CN}\left(\mathbf{t}|\mathbf{\Phi w}, \sigma^2\mathbf{I}\right) = \frac{1}{\pi^N \left|\sigma^2\mathbf{I}\right|} \exp\left\{-\frac{1}{\sigma^2}\|\mathbf{t} - \mathbf{\Phi w}\|_2^2\right\}. \tag{3.7}$$

We define a zero-mean complex Gaussian prior on the weights $\mathbf{w}$. The prior is given as

$$p\left(\mathbf{w}|\boldsymbol{\alpha}\right) = \prod_{i=0}^{M-1} \mathcal{CN}(w_i|0, \alpha_i^{-1}), \tag{3.8}$$

where $\boldsymbol{\alpha}$ is an M-length vector with hyperparameters. By defining the matrix $\mathbf{A} = \mathrm{diag}(\alpha_0, \ldots, a_{M-1})$, we can rewrite the prior distribution on the weights as

$$p\left(\mathbf{w}|\boldsymbol{\alpha}\right) = \mathcal{CN}\left(\mathbf{w}|\mathbf{0}, \mathbf{A}^{-1}\right) = \frac{1}{\pi^M \left|\mathbf{A}^{-1}\right|} \exp\left\{-\mathbf{w}^H \mathbf{A} \mathbf{w}\right\}. \tag{3.9}$$

The marginal likelihood is then given by

$$p\left(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2\right) = \mathcal{CN}\left(\mathbf{t}|\mathbf{0}, \mathbf{B} + \boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^H\right), \tag{3.10}$$

where $\mathbf{B}$ is defined as $\sigma^2\mathbf{I}$. Maximising the marginal likelihood $p\left(\mathbf{t}|\boldsymbol{\alpha}, \sigma^2\right)$ over each individual α_i and σ^2 gives the update equations for the individual α_i's and σ^2, respectively. The update for the individual α_i's is given by

$$\alpha_i = \frac{\gamma_i}{\boldsymbol{\mu}_i^2}, \tag{3.11}$$

where

$$\gamma_i = 1 - \alpha_i \boldsymbol{\Sigma}_{ii} \tag{3.12}$$

and $\boldsymbol{\mu}_i$ is defined as the expectation of the posterior distribution on the weight $\mathbf{w}_i$. The update equation for σ^2 is given by

$$\sigma^2 = \frac{\left\|\mathbf{t} - \boldsymbol{\Phi}\boldsymbol{\mu}\right\|_2^2}{N - \sum_{i=0}^{M-1} \gamma_i}. \tag{3.13}$$

The posterior on the weights $p\left(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2\right)$ is given by

$$p\left(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}, \sigma^2\right) = \mathcal{CN}\left(\mathbf{w}|\boldsymbol{\Sigma}\boldsymbol{\Phi}^H\mathbf{B}^{-1}\mathbf{t}, \boldsymbol{\Sigma}\right), \tag{3.14}$$

where $\boldsymbol{\Sigma} = \left(\mathbf{A} + \boldsymbol{\Phi}^H\mathbf{B}^{-1}\boldsymbol{\Phi}\right)^{-1}$.

A complete derivation of the marginal likelihood, the posterior distribution on the weights, and the update equations is given in Appendix A.

The relevance vector machine algorithm is given by iterating over the update equations of $\alpha_i, i \in \{0, 1, \ldots, M-1\}$ and σ^2 given in (3.11) and (3.13) together with both the first and second order moments of the posterior distribution of the weights (3.14), denoted by $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, respectively. The iteration continues until a convergence criterion is met. In practice, we see that most of the values of $\alpha_i, i \in \{0, 1, \ldots, M-1\}$ tend to infinity. When the value of α_i tends to infinity, it implies that the posterior distribution of the weight (3.14) becomes a Dirac delta distribution centred on zero. This posterior distribution implies that the corresponding weight will be equal to zero almost surely and can thus be removed from the model.

3.2. Extension to multiple measurement vectors

Having the complex-valued relevance vector machine, we would like a relevance vector machine that is able to use the information obtained across multiple bursts for our framework. By combining the information contained in multiple bursts with different PRTs, the framework should be able to resolve ambiguities, or at least increase the unambiguous domain significantly. For both the coherent and incoherent case, we consider a collection of B measurement equations as given in (2.4).

3.2.1. Coherent case

We first consider the case of coherent processing, i.e., we assume that $\mathbf{w}_i = \mathbf{w}_j \quad \forall i, j \in \{1, 2, \ldots, B\}$, where B is the total amount of bursts. In this case, we can write the joint measurement equation as case 3 of section 2.2. This approach directly translates to the relevance vector machine. We can just "stack" the measured data and the matrices of basis vectors for each individual burst to get the formulation as in (2.8). By defining

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\Phi}_1 \\ \boldsymbol{\Phi}_2 \\ \vdots \\ \boldsymbol{\Phi}_B \end{bmatrix} \quad \text{and} \quad \mathbf{t} = \begin{bmatrix} \mathbf{t}_1 \\ \mathbf{t}_2 \\ \vdots \\ \mathbf{t}_B \end{bmatrix}, \tag{3.15}$$

we can use the relevance vector machine to solve for $\mathbf{w}$.

3.2.2. Non-coherent case

For the non-coherent case, we have $\mathbf{\Phi}_i \neq \mathbf{\Phi}_j$ and $\mathbf{w}_i \neq \mathbf{w}_j \quad \forall i,j \in \{1,2,\dots,B\}$. This corresponds to case four of section 2.2. Like in case four of section 2.2, we write the joint measurement vectors equation as

$$\begin{bmatrix} \mathbf{t}_1 \\ \mathbf{t}_2 \\ \vdots \\ \mathbf{t}_B \end{bmatrix} = \begin{bmatrix} \mathbf{\Phi}_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{\Phi}_2 & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{\Phi}_B \end{bmatrix} \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \vdots \\ \mathbf{w}_B \end{bmatrix} + \boldsymbol{\varepsilon}, \tag{3.16}$$

where we define a joint prior on the weights of each individual measurement vector.

$$\begin{aligned} \mathbf{w}_1 &\sim \mathcal{CN}\left(\mathbf{0}, \mathbf{A}^{-1}\right) \\ \mathbf{w}_2 &\sim \mathcal{CN}\left(\mathbf{0}, \mathbf{A}^{-1}\right) \\ &\vdots \\ \mathbf{w}_B &\sim \mathcal{CN}\left(\mathbf{0}, \mathbf{A}^{-1}\right), \end{aligned} \tag{3.17}$$

where $\mathbf{A}$ is a diagonal matrix, like in the regular relevance vector machine, defined as $\mathbf{A} = \mathrm{diag}(\alpha_0, \dots, \alpha_{M-1})$. This is the formulation of the multitask relevance vector machine (MRVM) of Ji *et al.* [40]. In the multitask relevance vector machine, we see the processing of the B bursts as individual tasks that are connected via the prior on the weights. We can update the estimates for the weights $\boldsymbol{\mu}_b,\ b \in \{1,2,\dots,B\}$ and the covariance matrix $\mathbf{\Sigma}_b,\ b \in \{1,2,\dots,B\}$ for each individual burst with the regular update equations

$$\boldsymbol{\mu}_b = \mathbf{\Sigma}_b \mathbf{\Phi}_b^H \mathbf{B}^{-1} \mathbf{t}_b \quad \text{and} \quad \mathbf{\Sigma}_b = \left(\mathbf{A} + \mathbf{\Phi}_b^H \mathbf{B}^{-1} \mathbf{\Phi}_b\right)^{-1}, \tag{3.18}$$

respectively. $\mathbf{B}$ is still defined as $\mathbf{B} = \sigma^2 \mathbf{I}$. If we then define

$$\begin{aligned} \boldsymbol{\mu} &= \begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \\ \vdots \\ \boldsymbol{\mu}_B \end{bmatrix}, \qquad \mathbf{\Sigma} = \begin{bmatrix} \mathbf{\Sigma}_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_2 & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{\Sigma}_B \end{bmatrix}, \quad \mathbf{\Phi} = \begin{bmatrix} \mathbf{\Phi}_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{\Phi}_2 & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{\Phi}_B \end{bmatrix}, \\ \mathbf{t} &= \begin{bmatrix} \mathbf{t}_1 \\ \mathbf{t}_2 \\ \vdots \\ \mathbf{t}_B \end{bmatrix}, \qquad \mathbf{A}_B = \begin{bmatrix} \mathbf{A} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{A} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{A} \end{bmatrix}, \quad \text{and} \quad \mathbf{B}_B = \begin{bmatrix} \mathbf{B} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{B} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{B} \end{bmatrix}, \end{aligned} \tag{3.19}$$

we can follow the same manipulations of the logarithm of the marginal likelihood of $\mathbf{t}$ as done in the single measurement vector case. Doing so results in

$$\alpha_i = \frac{\gamma_i}{\sum_{b=1}^{B} \boldsymbol{\mu}_{b;i}^2}, \tag{3.20}$$

where

$$\gamma_i = B - \alpha_i \sum_{b=1}^{B} \mathbf{\Sigma}_{b;i,i} \tag{3.21}$$

and

$$\sigma^2 = \frac{\sum_{b=1}^{B} \left\| \mathbf{t}_b - \mathbf{\Phi}_b \boldsymbol{\mu}_b \right\|_2^2}{\sum_{b=1}^{B} \left(N - M + \sum_{i=0}^{M-1} \mathbf{\Sigma}_{b;i,i} \right)}. \tag{3.22}$$

$\mathbf{\Sigma}_{b;i,i}$ and $\boldsymbol{\mu}_{b,i}$ denote the (i,i)'th and i'th element of $\mathbf{\Sigma}_b$ and $\boldsymbol{\mu}_b$, respectively. The complete derivation of the update equations is given in Appendix B. We have now specified the full multitask relevance vector machine. The algorithm is given by first iterating over the update equations for the first and second order moments of the posterior of the weights for each individual task, as given in (3.18). We then update the joint variables $\alpha_i, i \in \{0,1,\dots,M-1\}$ and σ^2 by (3.20) and (3.22), respectively. As in the case of the regular relevance machine, the iteration continues until a convergence criterion is met. We also see that most of the values of $\alpha_i, i \in \{0,1,\dots,M-1\}$ tend to infinity. When the value of α_i tends to infinity, it still implies that the posterior distributions of the weights $\mathbf{w}_{b;i}\ \forall b \in \{1,2,\dots,B\}$ become Dirac delta distributions centred on zero. This distribution implies that these weights will be equal to zero almost surely and can thus be removed from the model.

4

Addressing the off-grid problem

In practice, the targets will never lie exactly on one of the gridpoints of a relevance vector machine. When not accounted for, the mismatch between the gridpoints and the actual target will lead to poor performance of the relevance vector machine. To mitigate the mismatch problem, we introduce two methods that can be applied within the framework. The first method iteratively updates the gridpoints, while the second method pre-processes the grid.

4.1. Off-grid relevance vector machines

To iteratively update the gridpoints used in the relevance vector machine, we turn to off-grid relevance vector machines. The most used off-grid relevance vector machine is proposed by Yang *et al.* [41] and is based on the method described in the earlier paper of Zhu *et al.* [42]. The paper of Zhu *et al.* [42] proposes to extend the measurement vector with a Gaussian variable that models the mismatch between the grid and the target and solves the problem using the sparse total least squares framework. Yang *et al.* [41] argue that the variable modelling the mismatch between girdpoint and actual target should be modelled as a uniform random variable. Yang *et al.* [41] then propose a solution within the relevance vector machine framework that relies on a linearisation of the measurement vector obtained by taking the first order Taylor expansion. The solution relies on a constrained optimisation problem that needs to be performed in each iteration. This optimisation problem proves to be non-trivial. Yang *et al.* [41] provide an implementation to solve the problem in a single step. However, we found that this solution often diverges and gets stuck at one of its constraints and thus leads to poor performance. The polynomial-root based method of Dai *et al.* [23] provides an update equation for the grid points themselves instead of a variable that models the distance between gridpoint and the actual target. Here, we introduce the polynomial-root based method of Dai *et al.* [23] and extend it to accommodate for multiple measurement vectors.

4.1.1. Polynomial-root based off-grid relevance vector machine

The polynomial-root based method exploits the structure of the collection of basis functions used in the relevance vector machine when estimating, e.g., Doppler frequency or direction of arrival. In the relevance vector machine, we work with the matrix consisting of basis vectors $\boldsymbol{\Phi}$. Writing the matrix in terms of its basis vectors gives

$$\boldsymbol{\Phi} = \left[\boldsymbol{\phi}\left(\theta_1\right), \boldsymbol{\phi}\left(\theta_2\right), \ldots, \boldsymbol{\phi}\left(\theta_M\right)\right], \tag{4.1}$$

where the individual basis vectors can be written as

$$\boldsymbol{\phi}\left(\theta_m\right) = \left[v\left(\theta_m\right)^0, v\left(\theta_m\right)^1, \ldots, v\left(\theta_m\right)^{N-1}\right]^T. \tag{4.2}$$

In the case of Doppler, $v\left(\theta_m\right)$ is given by

$$v\left(\theta_m\right) = \exp\left\{\frac{j2\pi 2T_0\theta_m f_c}{c}\right\}, \tag{4.3}$$

where T_0, f_c and c are the pulse repetition time, the carrier frequency and the speed of light, respectively. Dai *et al.* [23] propose an expectation-maximisation step to update the gridpoints $\boldsymbol{\theta}$. The updated estimate of $\hat{\boldsymbol{\theta}}$ is given by maximising

$$\begin{aligned}
&\mathbb{E}_{p\left(\mathbf{w}|\mathbf{t},\sigma^2,\boldsymbol{\alpha};\hat{\boldsymbol{\theta}}\right)}\left[\ln p\left(\mathbf{t} \mid \mathbf{w},\sigma^2;\hat{\boldsymbol{\theta}}\right)\right] \\
&= -\sigma^{-2}\mathbb{E}_{p\left(\mathbf{w}|\mathbf{t},\sigma^2,\boldsymbol{\alpha};\hat{\boldsymbol{\theta}}\right)}\left[\left\|\mathbf{t}-\boldsymbol{\Phi}_{\hat{\boldsymbol{\theta}}}\mathbf{w}\right\|_2^2\right] \\
&= -\sigma^{-2}\left\|\mathbf{t}-\boldsymbol{\Phi}_{\hat{\boldsymbol{\theta}}}\boldsymbol{\mu}\right\|_2^2 - \sigma^{-2}\operatorname{tr}\left\{\boldsymbol{\Phi}_{\hat{\boldsymbol{\theta}}}\boldsymbol{\Sigma}\boldsymbol{\Phi}_{\hat{\boldsymbol{\theta}}}^H\right\}.
\end{aligned} \tag{4.4}$$

where we ignored the terms independent of $\hat{\boldsymbol{\theta}}$. To get an update equation for a gridpoint $\hat{\theta}_m, m \in \{1,2,\dots,M\}$, we first take the derivative of (4.4) with respect to $v\left(\hat{\theta}_m\right)$, equate the expression to 0 and solve it for $v\left(\hat{\theta}_m\right)$.

$$\begin{aligned}
&\frac{\partial\left(-\sigma^{-2}\left\|\mathbf{t}-\boldsymbol{\Phi}_{\hat{\boldsymbol{\theta}}}\boldsymbol{\mu}\right\|_2^2 - \sigma^{-2}\operatorname{tr}\left\{\boldsymbol{\Phi}_{\hat{\boldsymbol{\theta}}}\boldsymbol{\Sigma}\boldsymbol{\Phi}_{\hat{\boldsymbol{\theta}}}^H\right\}\right)}{\partial v\left(\hat{\theta}_m\right)} \\
&= -\sigma^{-2}\frac{\partial\left\|\mathbf{t}-\boldsymbol{\Phi}_{\hat{\boldsymbol{\theta}}}\boldsymbol{\mu}\right\|_2^2}{\partial v\left(\hat{\theta}_m\right)} - \sigma^{-2}\frac{\partial\operatorname{tr}\left\{\boldsymbol{\Phi}_{\hat{\boldsymbol{\theta}}}\boldsymbol{\Sigma}\boldsymbol{\Phi}_{\hat{\boldsymbol{\theta}}}^H\right\}}{\partial v\left(\hat{\theta}_m\right)} \\
&= -\sigma^{-2}\left(\frac{\partial\boldsymbol{\phi}\left(\hat{\theta}_m\right)}{\partial v\left(\hat{\theta}_m\right)}\right)^H\left(\boldsymbol{\phi}\left(\hat{\theta}_m\right)|\boldsymbol{\mu}_m|^2 - \boldsymbol{\mu}_m^*\boldsymbol{\varepsilon}_{-m}\right) - \sigma^{-2}\left(\frac{\partial\boldsymbol{\phi}\left(\hat{\theta}_m\right)}{\partial v\left(\hat{\theta}_m\right)}\right)^H\boldsymbol{\Phi}_{\hat{\boldsymbol{\theta}}}\boldsymbol{\Sigma}_m \\
&= -\sigma^{-2}\left(\frac{\partial\boldsymbol{\phi}\left(\hat{\theta}_m\right)}{\partial v\left(\hat{\theta}_m\right)}\right)^H\left(\boldsymbol{\phi}\left(\hat{\theta}_m\right)|\boldsymbol{\mu}_m|^2 - \boldsymbol{\mu}_m^*\boldsymbol{\varepsilon}_{-m}\right) - \sigma^{-2}\left(\frac{\partial\boldsymbol{\phi}\left(\hat{\theta}_m\right)}{\partial v\left(\hat{\theta}_m\right)}\right)^H\left(\boldsymbol{\Sigma}_{m,m}\boldsymbol{\phi}\left(\hat{\theta}_m\right) + \sum_{i\neq m}\boldsymbol{\Sigma}_{i,m}\boldsymbol{\phi}\left(\hat{\theta}_i\right)\right) \\
&= -\sigma^{-2}\left(\frac{\partial\boldsymbol{\phi}\left(\hat{\theta}_m\right)}{\partial v\left(\hat{\theta}_m\right)}\right)^H\left(\boldsymbol{\phi}\left(\hat{\theta}_m\right)\left(|\boldsymbol{\mu}_m|^2 + \boldsymbol{\Sigma}_{m,m}\right) + \sum_{i\neq m}\boldsymbol{\Sigma}_{i,m}\boldsymbol{\phi}\left(\hat{\theta}_i\right) - \boldsymbol{\mu}_m^*\boldsymbol{\varepsilon}_{-m}\right) = 0,
\end{aligned} \tag{4.5}$$

where $\boldsymbol{\mu}_m$, $\boldsymbol{\Sigma}_m$ and $\boldsymbol{\Sigma}_{m,i}$ denote the m'th element, the m'th column and the (m,i)'th element of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, respectively. $\boldsymbol{\varepsilon}_{-m}$ is defined as $\mathbf{t}-\sum_{i\neq m}\boldsymbol{\mu}_i\boldsymbol{\phi}\left(\theta_i\right)$. I.e., it is the residual between the observed signal and the reconstructed signal with the m'th basis function removed. The steps in (4.5) are not explicitly shown in [23], but are shown here as they are instrumental in extending this method to multiple measurement vectors.

If we define

$$\zeta^{(m)} = \left(|\boldsymbol{\mu}_m|^2 + \boldsymbol{\Sigma}_{m,m}\right) \tag{4.6}$$

and

$$\boldsymbol{\varphi}^{(m)} = \sum_{i\neq m}\boldsymbol{\Sigma}_{i,m}\boldsymbol{\phi}\left(\theta_i\right) - \boldsymbol{\mu}_m^*\boldsymbol{\varepsilon}_{-m}, \tag{4.7}$$

we can write (4.5) as

$$\left[v\left(\hat{\theta}_m\right), 1, v\left(\hat{\theta}_m\right)^{-1}, \dots, v\left(\hat{\theta}_m\right)^{-(N-2)}\right]\begin{bmatrix} \frac{N(N-1)}{2}\zeta^{(m)} \\ \boldsymbol{\varphi}_2^{(m)} \\ 2\boldsymbol{\varphi}_3^{(m)} \\ \vdots \\ (N-1)\boldsymbol{\varphi}_N^{(m)} \end{bmatrix} = 0, \tag{4.8}$$

where $\boldsymbol{\varphi}_i$ denotes the i'th element of $\boldsymbol{\varphi}$. Giving us the polynomial we need to solve for its roots. A method to solve a polynomial for its root is given in Appendix C. As the polynomial is of order $N-1$, the polynomial will have $N-1$ roots. By definition, the root to be chosen should have an absolute value

of one. However, due to noise the root might not exactly lie on the unit circle and therefore we choose the root with the absolute value closest to one. This root is denoted as $v\left(\hat{\theta}_m^*\right)$. The updated gridpoint is then calculated by

$$\hat{\theta}_m^* = \frac{angle\left(v\left(\hat{\theta}_m^*\right)\right)c}{2\pi T_0 f_c}. \tag{4.9}$$

We accept this updated gridpoint when the proposed $\hat{\theta}_m^*$ lies within the set of $\left[\frac{\theta_{m-1}+\theta_m}{2}, \frac{\theta_m+\theta_{m+1}}{2}\right]$.

4.1.2. Multiple measurement vectors

We now extend the polynomial-root based method to multiple measurement vectors. Let $T_1, T_2, \ldots, T_B$ be the pulse repetition times of all the individual bursts. If $\frac{T_i}{T_j} \in \mathbb{Q}$, $\forall \{T_i, T_j\}$, where $i, j \in \{1, 2, \ldots, B\}$, then $\forall b \in \{1, 2, \ldots, B\}$, $\exists c_b \in \mathbb{N}$ such that we can write $T_b = c_b * T_0$, where T_0 is the greatest common divisor of $T_1, T_2, \ldots, T_B$. So, we can write

$$\begin{aligned} \boldsymbol{\Phi}_1 &= \left[\boldsymbol{\phi}_1\left(\theta_1\right), \boldsymbol{\phi}_1\left(\theta_2\right), \ldots, \boldsymbol{\phi}_1\left(\theta_M\right)\right] \\ \boldsymbol{\phi}_2 &= \left[\boldsymbol{\phi}_2\left(\theta_1\right), \boldsymbol{\phi}_2\left(\theta_2\right), \ldots, \boldsymbol{\phi}_2\left(\theta_M\right)\right] \\ &\vdots \\ \boldsymbol{\Phi}_B &= \left[\boldsymbol{\phi}_B\left(\theta_1\right), \boldsymbol{\phi}_B\left(\theta_2\right), \ldots, \boldsymbol{\phi}_B\left(\theta_M\right)\right], \end{aligned} \tag{4.10}$$

where the individual basis vectors can be written as

$$\boldsymbol{\phi}_b\left(\theta_m\right) = \left[v_b\left(\theta_m\right)^0, v_b\left(\theta_m\right)^1, \ldots, v_b\left(\theta_m\right)^{N-1}\right]^T. \tag{4.11}$$

$v_b\left(\theta_m\right)$ is given by

$$v_b\left(\theta_m\right) = \exp\left\{\frac{j2\pi 2T_b\theta_m f_c}{c}\right\} = \exp\left\{\frac{j2\pi 2c_b T_0\theta_m f_c}{c}\right\}. \tag{4.12}$$

We can thus write

$$\boldsymbol{\phi}_b\left(\theta_m\right) = \left[v_0\left(\theta_m\right)^0, v_0\left(\theta_m\right)^{c_b}, \ldots, v_0\left(\theta_m\right)^{c_b(N-1)}\right]^T, \tag{4.13}$$

where

$$v_0 = \exp\left\{\frac{j2\pi 2T_0\theta_m f_c}{c}\right\}. \tag{4.14}$$

In the case of multiple measurement vectors, the expectation-maximisation step is given by maximising

$$\begin{aligned} &\mathbb{E}_{p\left(\mathbf{w}|\mathbf{t},\sigma^2,\boldsymbol{\alpha};\hat{\boldsymbol{\theta}}\right)}\left[\ln p\left(\mathbf{t} \mid \mathbf{w}, \sigma^2; \hat{\boldsymbol{\theta}}\right)\right] \\ &= -\sigma^{-2}\sum_{b=1}^{B}\mathbb{E}_{p\left(\mathbf{w}|\mathbf{t},\sigma^2,\boldsymbol{\alpha};\hat{\boldsymbol{\theta}}\right)}\left[\left\|\mathbf{t_b} - \boldsymbol{\Phi}_{b,\hat{\boldsymbol{\theta}}}\mathbf{w_b}\right\|_2^2\right] \\ &= -\sigma^{-2}\sum_{b=1}^{B}\left(\left\|\mathbf{t_b} - \boldsymbol{\Phi}_{b,\hat{\boldsymbol{\theta}}}\boldsymbol{\mu}_b\right\|_2^2 - \sigma^{-2}\,\mathrm{tr}\left\{\boldsymbol{\Phi}_{b,\hat{\boldsymbol{\theta}}}\boldsymbol{\Sigma}_b\boldsymbol{\Phi}_{b,\hat{\boldsymbol{\theta}}}^H\right\}\right). \end{aligned} \tag{4.15}$$

As in the case of a single measurement vector, we again take the derivative of (4.15), but now with respect to $v_0\left(\hat{\theta}_m\right)$, equate the expression to 0 and solve it for $v_0\left(\hat{\theta}_m\right)$. Following the same steps as in (4.5) results in

$$\begin{aligned} &\frac{\partial\left(-\sigma^{-2}\sum_{b=1}^{B}\left(\left\|\mathbf{t_b} - \boldsymbol{\Phi}_{b,\hat{\boldsymbol{\theta}}}\boldsymbol{\mu}_b\right\|_2^2 - \sigma^{-2}\,\mathrm{tr}\left\{\boldsymbol{\Phi}_{b,\hat{\boldsymbol{\theta}}}\boldsymbol{\Sigma}_b\boldsymbol{\Phi}_{b,\hat{\boldsymbol{\theta}}}^H\right\}\right)\right)}{\partial v_0\left(\hat{\theta}_m\right)} \\ &= -\sigma^{-2}\sum_{b=1}^{B}\left(\frac{\partial\boldsymbol{\phi}_b\left(\hat{\theta}_m\right)}{\partial v_0\left(\hat{\theta}_m\right)}\right)^H\left(\boldsymbol{\phi}_b\left(\hat{\theta}_m\right)\left(|\boldsymbol{\mu}_{b;m}|^2 + \boldsymbol{\Sigma}_{b;m,m}\right) + \sum_{i\neq m}\boldsymbol{\Sigma}_{b;i,m}\boldsymbol{\phi}_b\left(\hat{\theta}_i\right) - \boldsymbol{\mu}_{b;m}^*\boldsymbol{\varepsilon}_{b;-m}\right) = 0, \end{aligned} \tag{4.16}$$

where $\boldsymbol{\mu}_{b;m}$ and $\boldsymbol{\Sigma}_{b;m,i}$ denote the m'th element and the (m,i)'th element of $\boldsymbol{\mu}_b$ and $\boldsymbol{\Sigma}_b$, respectively. $\boldsymbol{\varepsilon}_{b;-m}$ is defined as $\mathbf{t}_b - \sum_{i\neq m} \boldsymbol{\mu}_{b;i}\boldsymbol{\phi}_b\left(\theta_i\right)$. I.e., it is the residual between the observed in the b'th burst and the reconstructed signal with the m'th basis function removed.

Using (4.13) and

$$\left(\frac{\partial \boldsymbol{\phi}_b\left(\hat{\theta}_m\right)}{\partial v_0\left(\hat{\theta}_m\right)}\right) = \left[0, c_b v_0\left(\hat{\theta}_k\right)^0, 2c_b v_0\left(\hat{\theta}_k\right)^1, 3c_b v_0\left(\hat{\theta}_k\right)^2, \ldots, (N-1)c_b v_0\left(\hat{\theta}_k\right)^{(N-1)c_b-1}\right], \tag{4.17}$$

we write the polynomial for multiple bursts as

$$\sum_{b=1}^{B}\left[v_0\left(\hat{\theta}_m\right)^{c_b}, 1, v_0\left(\hat{\theta}_m\right)^{-c_b}, \ldots, v_0\left(\hat{\theta}_m\right)^{-(N-1)c_b-1}\right]\begin{bmatrix} \frac{N(N-1)c_b}{2}\zeta_b^{(m)} \\ c_b\boldsymbol{\varphi}_{b;2}^{(m)} \\ 2c_b\boldsymbol{\varphi}_{b;3}^{(m)} \\ \vdots \\ (N-1)c_b\boldsymbol{\varphi}_{b;N}^{(m)} \end{bmatrix} = 0, \tag{4.18}$$

where $\zeta_b^{(m)}$ and $\boldsymbol{\varphi}_b^{(m)}$ are defined as

$$\zeta_b^{(m)} = \left(|\boldsymbol{\mu}_{\mathbf{b};\mathbf{m}}|^2 + \boldsymbol{\Sigma}_{b;m,m}\right) \tag{4.19}$$

and

$$\boldsymbol{\varphi}_b^{(m)} = \sum_{i\neq m}\boldsymbol{\Sigma}_{b;i,m}\boldsymbol{\phi}_b\left(\theta_i\right) - \boldsymbol{\mu}_{b;m}^{*}\boldsymbol{\varepsilon}_{b;-m}, \tag{4.20}$$

respectively. This new polynomial is a sum of B polynomials. Each individual polynomial corresponds to a single measurement vector. The order of the new polynomial is $\max_b\{c_b\}(N-1)$, $b \in 1,2,\ldots,B$. Again, as is the case with a single measurement vector, the root to be chosen should have an absolute value of one. However, due to noise the root might not exactly lie on the unit circle and therefore we choose the root with the absolute value closest to one. This root is denoted as $v\left(\hat{\theta}_m^*\right)$. The updated gridpoint is then calculated by (4.9). We accept this updated gridpoint when the proposed $\hat{\theta}_m^*$ lies within the set of $\left[\frac{\theta_{m-1}+\theta_m}{2}, \frac{\theta_m+\theta_{m+1}}{2}\right]$.

4.2. Pre-processing of grid points

Instead of using the proposed method to iteratively update the gridpoints of the relevance vector machine of subsection 4.1.2, we can pre-determine points of interest on the grid. Such a scheme is more heuristic than iteratively searching for the optimal positions of gridpoints, but does significantly lower the computational cost as we have to compute the points of interest once, compared to updating each individual gridpoint in every iteration for the proposed method in subsection 4.1.2. To determine points of interest on the grid, we propose to use the variational Bayesian line spectral estimation algorithm of Badiu *et al.* [24], also known as VALSE.

4.2.1. Variational Bayesian inference

Before deriving VALSE, we introduce Variational Bayesian inference. Variational Bayesian inference is a method used to approximate posterior distributions. In Bayesian statistics, we often encounter models for which the posterior distribution is not straightforward to compute or characterise due to the fact that we encounter intractable integrals. These situations drive us towards alternative strategies such as MCMC sampling or the use of variational Bayesian inference.

MCMC techniques are a class of algorithms in which we construct a Markov chain on the set of latent variables of a given statistical model that has the posterior of those latent variables as stationary distribution. By sampling long enough, we will thus sample from the true posterior distribution. We then make an empirical estimate of the posterior distribution from the collected samples of that Markov chain. MCMC techniques thus guarantee to sample exactly from the target distribution in its limit, variational Bayesian inference methods cannot make such a guarantee as we rely on approximations. However,

an MCMC sampler takes a long time to converge and does not scale well with increasing data. There also do not exist clear convergence measures to know whether we are actually sampling from the limiting distribution. Variational Bayesian inference does not suffer from those drawbacks as it generally converges quicker compared to MCMC samplers, scales better with increasing data, and clearly shows convergence.

The name variational Bayesian inference is derived from calculus of variations. In calculus of variations, we work with functionals. A functional is a mapping that takes in a function as input and then returns an output corresponding to the functional. A typical example of a functional is the entropy of a random variable. The entropy takes in a probability density function and gives a scalar as output value. For a detailed description of calculus of variations, corresponding definitions, theorems and rules, the reader is referred to the book of Sagan [43].

In variational Bayesian inference, we rewrite the problem to an optimisation problem in which we try to optimise a similarity measure, such as the Kullback–Leibler divergence, over a restricted set of functions. As this similarity measure takes in a function and returns a value, it is a functional, which we try to optimise over a restricted set of functions. The most commonly used restriction is the factorisation assumption, or mean field approximation [44]. We thus search for the function within the restricted set of functions that is most similar to the exact density.

Now consider a model in which we have a set of observations $\mathbf{x} = \{x_1, \dots, x_N\}$ and a set of latent variables and parameters $\mathbf{z} = \{z_1, \dots, z_M\}$. We have a model that specifies the joint distribution on $p(\mathbf{x}, \mathbf{y})$. We would now like to find approximations for the posterior distribution of the latent variables $p(\mathbf{z}|\mathbf{x})$ as well as for the model evidence $p(\mathbf{x})$. Let us first rewrite the log marginal probability as

$$\begin{aligned}
\ln p(\mathbf{x}) &= \int q(\mathbf{z}) \mathrm{d}\mathbf{z} \ln\{p(\mathbf{x})\} \\
&= \int q(\mathbf{z}) \ln\{p(\mathbf{x})\} \mathrm{d}\mathbf{z} \\
&= \int q(\mathbf{z}) \ln\left\{\frac{p(\mathbf{x},\mathbf{z})}{p(\mathbf{x}|\mathbf{z})}\right\} \mathrm{d}\mathbf{z} \\
&= \int q(\mathbf{z}) \left(\ln\{p(\mathbf{x},\mathbf{z})\} - \ln\{p(\mathbf{x}|\mathbf{z})\}\right) \mathrm{d}\mathbf{z} \\
&= \int q(\mathbf{z}) \left(\ln\left\{\frac{p(\mathbf{x},\mathbf{z})}{q(\mathbf{z})}\right\} - \ln\left\{\frac{p(\mathbf{x}|\mathbf{z})}{q(\mathbf{z})}\right\}\right) \mathrm{d}\mathbf{z} \\
&= \int q(\mathbf{z}) \ln\left\{\frac{p(\mathbf{x},\mathbf{z})}{q(\mathbf{z})}\right\} \mathrm{d}\mathbf{z} - \int q(\mathbf{z}) \ln\left\{\frac{p(\mathbf{x}|\mathbf{z})}{q(\mathbf{z})}\right\} \mathrm{d}\mathbf{z} \\
&= \mathcal{L}(q) + \mathrm{KL}(q,p),
\end{aligned} \tag{4.21}$$

where $q(\mathbf{z})$ is an arbitrary probability density and we have defined the evidence lower bound $\mathcal{L}(q)$ and the Kullback-Leibler divergence $\mathrm{KL}(q,p)$ as

$$\mathcal{L}(q) = \int q(\mathbf{z}) \ln\left\{\frac{p(\mathbf{x},\mathbf{z})}{q(\mathbf{z})}\right\} \mathrm{d}\mathbf{z} \tag{4.22}$$

and

$$\mathrm{KL}(q,p) = -\int q(\mathbf{z}) \ln\left\{\frac{p(\mathbf{x}|\mathbf{z})}{q(\mathbf{z})}\right\} \mathrm{d}\mathbf{z} \tag{4.23}$$

respectively. The Kullback-Leibler divergence has as property that $\mathrm{KL}(q,p) \geq 0$, with $\mathrm{KL}(q,p) = 0$ if and only if $q(\mathbf{x}) = p(\mathbf{x}|\mathbf{z})$ [45]. Given this property and the equality in (4.21) it is implied that $\mathcal{L}(q) \leq \ln p(\mathbf{x})$. So $\mathcal{L}(q)$ is thus a lower bound on the log model evidence $\ln p(\mathbf{x})$ and therefore called the evidence lower bound. We thus would like to maximise the evidence lower bound $\mathcal{L}(q)$ with respect to $q(\mathbf{z})$, which is equivalent to minimising the Kullack-Leibler divergence. If we do not restrict the set of distributions over which we maximise the evidence lower bound, we would find the optimal solution at $q(\mathbf{z}) = p(\mathbf{z}|\mathbf{x})$. Implying that we again find the true posterior which we wanted to avoid due to the fact that it is intractable.

To avoid the same intractable solution, we consider a restricted set of distributions $q(\mathbf{z})$. Within this set, we look for the distribution that minimises the Kullback-Leibler divergence. Generally, when restricting the set of distributions over which we minimise the Kullback-Leibler divergence, we want to make the restriction as moderate as possible. The only purpose served by the restriction is to retain tractability, a light restriction can not cause over-fitting as the unrestricted optimum is the true posterior distribution $p(\mathbf{x}|\mathbf{y})$. The specific restriction we will consider is the mean field approximation [44]. In the mean field approximation, we partition the set of latent variables and parameters $\mathbf{z}$ into K pairwise disjoint sets, $\mathbf{z}_i, i \in \{1, \dots, K\}$. We then make the assumption that $q(\mathbf{z})$ factorises over the defined pairwise disjoint sets, i.e.,

$$q(\mathbf{z}) = \prod_{i=1}^{K} q_i(\mathbf{z}_i). \tag{4.24}$$

This factorisation approach results in a trade-off. The lower the number of pairwise disjoint sets, the better the factorisation will approximate the true posterior. However, by using few sets, the problem remains intractable.

Given the factorised form of $q(\mathbf{z})$, we would like to optimise the evidence lower bound $\mathcal{L}(q)$ with respect to all factors of the factorisation $q_i(\mathbf{z}_i)$. This optimisation is carried out sequentially by optimising over each individual factor. To do this, we first isolate the dependence, specifically on the factor concerning the j'th set, $q_j(\mathbf{z}_j)$. By plugging the factorised form of $q(\mathbf{z})$ into the evidence lower bound, we write

$$\mathcal{L}(q) = \int \prod_{i=1}^{K} q_i(\mathbf{z}_i) \left(\ln\left\{p(\mathbf{x}, \mathbf{z})\right\} - \sum_{i=1}^{K} \ln\left\{q_i(\mathbf{z}_i)\right\} \right) \mathrm{d}\mathbf{z}. \tag{4.25}$$

We then collect all terms depending on the j'th factor. Resulting in

$$\begin{aligned} \mathcal{L}(q) &= \int q_j(\mathbf{z}_j) \left(\int \ln\left\{p(\mathbf{x}, \mathbf{z})\right\} \prod_{i \neq j} q_i(\mathbf{z}_i) \mathrm{d}\mathbf{z}_i \right) \mathrm{d}\mathbf{z}_j - \int q_j(\mathbf{z}_j) \ln\left\{q_j(\mathbf{z}_j)\right\} \mathrm{d}\mathbf{z}_j + \mathrm{Const} \\ &= \int q_j(\mathbf{z}_j) \mathbb{E}_{i \neq j}\left[\ln p(\mathbf{x}, \mathbf{z})\right] \mathrm{d}\mathbf{z}_j - \int q_j(\mathbf{z}_j) \ln\left\{q_j(\mathbf{z}_j)\right\} \mathrm{d}\mathbf{z}_j + \mathrm{Const}, \end{aligned} \tag{4.26}$$

where Const encompasses all terms independent of the j'th factor and $\mathbb{E}_{i \neq j}\left[\ln p(\mathbf{x}, \mathbf{y})\right]$ denotes the expectation of $\ln p(\mathbf{x}, \mathbf{y})$ with respect to $q_i(\mathbf{z}_i), \forall i \neq j$, given by

$$\mathbb{E}_{i \neq j}\left[\ln p(\mathbf{x}, \mathbf{z})\right] = \int \ln\left\{p(\mathbf{x}, \mathbf{z})\right\} \prod_{i \neq j} q_i(\mathbf{z}_i) \mathrm{d}\mathbf{z}_i. \tag{4.27}$$

Having the expression where we have isolated the factor corresponding to the j'th set $q_j(\mathbf{z}_j)$, we now keep all other factors fixed and maximise $\mathcal{L}(q)$ with respect to $q_j(\mathbf{z}_j)$. We do this by making the observation that $\mathcal{L}(q)$ can be written as the negative Kullback-Leibler divergence between $\exp\left\{\mathbb{E}_{i \neq j}\left[\ln p(\mathbf{x}, \mathbf{z})\right]\right\}$ and $q_j(\mathbf{z}_j)$. I.e.,

$$\begin{aligned} \mathcal{L}(q) &= \int q_j(\mathbf{z}_j) \mathbb{E}_{i \neq j}\left[\ln p(\mathbf{x}, \mathbf{z})\right] \mathrm{d}\mathbf{z}_j - \int q_j(\mathbf{z}_j) \ln\left\{q_j(\mathbf{z}_j)\right\} \mathrm{d}\mathbf{z}_j + \mathrm{Const} \\ &= \int q_j(\mathbf{z}_j) \ln\left\{\exp\left\{\mathbb{E}_{i \neq j}\left[\ln p(\mathbf{x}, \mathbf{z})\right]\right\}\right\} \mathrm{d}\mathbf{z}_j - \int q_j(\mathbf{z}_j) \ln\left\{q_j(\mathbf{z}_j)\right\} \mathrm{d}\mathbf{z}_j + \mathrm{Const} \\ &= \int q_j(\mathbf{z}_j) \ln\left\{\frac{\exp\left\{\mathbb{E}_{i \neq j}\left[\ln p(\mathbf{x}, \mathbf{z})\right]\right\}}{q_j(\mathbf{z}_j)}\right\} \mathrm{d}\mathbf{z}_j + \mathrm{Const} \\ &= -\mathrm{KL}\left(q_j(\mathbf{z}_j), \exp\left\{\mathbb{E}_{i \neq j}\left[\ln p(\mathbf{x}, \mathbf{z})\right]\right\}\right) + \mathrm{Const}. \end{aligned} \tag{4.28}$$

Optimising $\mathcal{L}(q)$ is thus equivalent to minimising the Kullback Leibler divergence between $\exp\left\{\mathbb{E}_{i \neq j}\left[\ln p(\mathbf{x}, \mathbf{z})\right]\right\}$ and $q_j(\mathbf{z}_j)$. As the Kullback Leibler divergence is minimised for an equality between $\exp\left\{\mathbb{E}_{i \neq j}\left[\ln p(\mathbf{x}, \mathbf{z})\right]\right\}$

and $q_j(\mathbf{z}_j)$, we arrive at the following solution for the optimisation of $\mathcal{L}(q)$

$$q_j(\mathbf{z}_j) = \exp\left\{\mathbb{E}_{i\neq j}\left[\ln p(\mathbf{x},\mathbf{z})\right]\right\} + \text{Const}, \tag{4.29}$$

where the constant should be set by normalising the distribution [46]. Resulting in

$$q_j(\mathbf{z}_j) = \frac{\exp\left\{\mathbb{E}_{i\neq j}\left[\ln p(\mathbf{x},\mathbf{z})\right]\right\}}{\int \exp\left\{\mathbb{E}_{i\neq j}\left[\ln p(\mathbf{x},\mathbf{z})\right]\right\} \mathrm{d}\mathbf{z}}. \tag{4.30}$$

Giving us an update expression for $q_j(\mathbf{z}_j)$ in terms of all other factors $q_i(\mathbf{z}_i), i \neq j$. Such an optimisation scheme ensures convergence [47].

4.2.2. VALSE

VALSE is proposed by Badiu *et al.* [24] and provides us with the perfect tool to make estimates of the components to add to the grid of the relevance vector machine.

VALSE is given by the following model specification

$$\mathbf{t} = \sum_{i=1}^{M} w_i \boldsymbol{\phi}_i, \tag{4.31}$$

where $\mathbf{t}$ is the vector of observations, w_i is the weight corresponding to the i'th component, and $\boldsymbol{\phi}_i$ is the the i'th vector of some function mapping its argument $[-\pi,\pi) \to \mathbb{C}^N$. In our case, $\boldsymbol{\phi}_i$ will correspond to the i'th relevance vector.

We model the weights as complex Gaussians that are activated by Bernoulli variables s_i. The total set of indicators resulting from the Bernoulli variables represents the support of the model and is denoted by $\mathbf{s}$. (s_i, w_i) thus follows a Bernoulli-Gaussian process. I.e.,

$$p_{w_i|s_i}\left(w_i \mid s_i;\tau\right) = \left(1-s_i\right)\delta\left(w_i\right) + s_i \mathcal{CN}\left(w_i;0,\tau\right), \tag{4.32}$$

with

$$p_{s_i}\left(s_i\right) = \rho^{s_i}(1-\rho)^{(1-s_i)}. \tag{4.33}$$

The parameter ρ governs how likely any component is present. The set of frequencies, in our case Doppler frequencies, is denoted by $\boldsymbol{\theta}$. The prior on the frequencies is given by

$$p_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \prod_{i=1}^{M} p_{\theta_i}\left(\theta_i\right), \tag{4.34}$$

where each $p_{\theta_i}\left(\theta_i\right)$ is a von Mises pdf. For more details of the von Mises distribution, its properties and other types of wrapped and directional distributions, the reader is referred to [48]. For a completely uninformative prior on the Doppler velocity, we initialise the concentration parameters of the von Mises distributions as zero. The error term $\boldsymbol{\varepsilon}$ is assumed to be zero mean complex Gaussian noise with variance ν. This noise term results in the model likelihood

$$p_{\mathbf{t}|\boldsymbol{\theta},\mathbf{w}}\left(\mathbf{t}|\boldsymbol{\theta},\mathbf{w};\nu\right) = \mathcal{CN}\left(\mathbf{t}\middle|\sum_{i=1}^{M} w_i \boldsymbol{\phi}\left(\theta_i\right), \nu\mathbf{I}\right). \tag{4.35}$$

We now have three parameters: ν, ρ and τ. We denote these parameters collectively as $\boldsymbol{\beta} = \{\nu,\rho,\tau\}$.

In the end, we want to get a posterior on the weights, the support and the frequencies

$$p_{\boldsymbol{\theta},\mathbf{w},\mathbf{s}|\mathbf{t}}\left(\boldsymbol{\theta},\mathbf{w},\mathbf{s}|\mathbf{t};\boldsymbol{\beta}\right) = \frac{p_{\mathbf{t},\boldsymbol{\theta},\mathbf{w},\mathbf{s}}\left(\mathbf{t},\boldsymbol{\theta},\mathbf{w},\mathbf{s};\boldsymbol{\beta}\right)}{p_{\mathbf{t}}\left(\mathbf{t};\boldsymbol{\beta}\right)}, \tag{4.36}$$

where the joint pdf is given by the model likelihood multiplied with the defined priors. I.e.,

$$p_{\mathbf{t},\boldsymbol{\theta},\mathbf{w},\mathbf{s}}\left(\mathbf{t},\boldsymbol{\theta},\mathbf{w},\mathbf{s};\boldsymbol{\beta}\right)=p_{\mathbf{t}|\boldsymbol{\theta},\mathbf{w}}\left(\mathbf{t}|\boldsymbol{\theta},\mathbf{w};\nu\right)\prod_{i=1}^{M}p_{\theta_i}\left(\theta_i\right)p_{w_i|s_i}\left(w_i\mid s_i\right)p_{s_i}\left(s_i\right). \tag{4.37}$$

Since (4.36) is not tractable, we turn to variational inference. We specifically use the naive mean field approximation as described in subsection 4.2.1. We thus assume all frequencies to be a posteriori independent, Mutually and of other variables

$$q_{\boldsymbol{\theta},\mathbf{w},\mathbf{s}|\mathbf{t}}\left(\boldsymbol{\theta},\mathbf{w},\mathbf{s}|\mathbf{t};\boldsymbol{\beta}\right)=\prod_{i=1}^{M}q_{\theta_i|\mathbf{t}}\left(\theta_i|\mathbf{t}\right)q_{\mathbf{w}|\mathbf{t},\mathbf{w}}\left(\mathbf{w}|\mathbf{t},\mathbf{w}\right)q_{\mathbf{s}|\mathbf{t}}\left(\mathbf{s}|\mathbf{t}\right). \tag{4.38}$$

We then make the assumption that the factors for the support have all their probability mass at the estimator of the support $\hat{\mathbf{s}}$. I.e., $q_{\mathbf{s}|\mathbf{t}}\left(\mathbf{s}|\mathbf{t}\right)=\delta\left(\mathbf{s}-\hat{\mathbf{s}}\right)$. When using this factorisation, the estimate of the frequency θ_i is then given by

$$\hat{\theta}_i=\arg\left(\mathbb{E}_{q_{\theta_i,\mathbf{t}}}\left[\exp\left\{j\theta_i\right\}\right]\right) \tag{4.39}$$

as shown in [48]. The variational estimates for the mean and the covariance of the weights are

$$\hat{\mathbf{w}}=\mathbb{E}_{q_{\mathbf{w}|\mathbf{t}}}\left[\mathbf{w}\right] \tag{4.40}$$

and

$$\hat{\mathbf{C}}=\mathbb{E}_{q_{\mathbf{w}|\mathbf{t}}}\left[\mathbf{w}\mathbf{w}^H\right]-\hat{\mathbf{w}}\hat{\mathbf{w}}^H, \tag{4.41}$$

respectively.

We now show how to infer the frequencies in the VALSE model. As shown in subsection 4.2.1, the solution for the variational approximation with respect to the factor $q_{\theta_i|\mathbf{t}}(\theta_i|\mathbf{t})$ is given by (4.29). Filling in, we get

$$\begin{aligned}\ln q_{\theta_i|\mathbf{t}}(\theta_i|\mathbf{t})&=\mathbb{E}_{\sim\theta_i}\left[p_{\mathbf{t},\boldsymbol{\theta},\mathbf{w},\mathbf{s}}\left(\mathbf{t},\theta_i,\boldsymbol{\theta}_{\sim i},\mathbf{w},\mathbf{s};\boldsymbol{\beta}\right)\right]+\text{Const}\\&=\mathbb{E}_{\sim\theta_i}\left[p_{\mathbf{t}|\boldsymbol{\theta},\mathbf{w}}\left(\mathbf{t}\middle|\theta_i,\boldsymbol{\theta}_{\sim i},\mathbf{w};\boldsymbol{\beta}\right)\right]+\ln p_{\theta_i}\left(\theta_i\right)+\text{Const.}\end{aligned} \tag{4.42}$$

Filling in the model likelihood (4.35) gives

$$q_{\theta_i|\mathbf{t}}(\theta_i|\mathbf{t})\propto p_{\theta_i}\left(\theta_i\right)\exp\left\{\mathcal{R}\left\{\boldsymbol{\eta}_i^H\boldsymbol{\phi}\left(\theta_i\right)\right\}\right\}, \tag{4.43}$$

where the vector $\boldsymbol{\eta}_i$ is

$$\boldsymbol{\eta}_i=\frac{2}{\nu}\left(\mathbf{t}-\sum_{l\in\hat{\mathbf{s}}\setminus i}\hat{w}_l\boldsymbol{\phi}\left(\hat{\theta}_l\right)\right)\hat{w}_i^*-\frac{2}{\nu}\sum_{l\in\hat{\mathbf{s}}\setminus i}\hat{C}_{l,i}\boldsymbol{\phi}\left(\hat{\theta}_l\right). \tag{4.44}$$

From examining (4.43), we see that (4.43) is an m-fold wrapped von Mises distribution. Such a distribution is a strongly multimodal function and does not lead to an analytic expression for $E_{q_{\theta_i|\mathbf{t}}}[\boldsymbol{\phi}(\theta_i)]$. [24] provides a method to approximate such an m-fold wrapped von Mises distribution with a mixture of von Mises distributions by matching their characteristic functions. By using the mixture of von Mises approximation, we get a closed form expression for $\mathbb{E}_{q_{\theta_i|\mathbf{t}}}[\boldsymbol{\phi}(\theta_i)]$.

For the inference of the weights and support in the VALSE model, we solve the variational problem for $q_{\mathbf{w},\mathbf{s}|\mathbf{t}}(\mathbf{w},\mathbf{s}|\mathbf{t})$ when keeping $q_{\theta_i|\mathbf{t}}$ fixed. Since we restrict $q_{\mathbf{w},\mathbf{s}|\mathbf{t}}(\mathbf{w},\mathbf{s}|\mathbf{t})$ in (4.38) to get the marginal probability mass function $q_{\mathbf{s}|\mathbf{t}}(\mathbf{s}|\mathbf{t})=\delta(\mathbf{s}-\hat{\mathbf{s}})$, we cannot use the regular solution to the variational approximation problem. We therefore plug (4.38) directly into (4.28) to get

$$\mathcal{L}\left(q_{\mathbf{w}|\mathbf{t},\mathbf{s}},\hat{\mathbf{s}}\right)=\text{Const}-\mathbb{E}\left[\ln q_{\mathbf{w}|\mathbf{t},\mathbf{s}}\left(\mathbf{w}|\mathbf{t},\hat{\mathbf{s}}\right)-\mathbb{E}\left[\ln p_{\mathbf{t},\boldsymbol{\theta},\mathbf{w},\mathbf{s}}\left(\mathbf{t},\boldsymbol{\theta},\mathbf{w},\hat{\mathbf{s}};\boldsymbol{\beta}\right)\right]\right]. \tag{4.45}$$

Introducing the probability density function

$$g(\mathbf{w};\hat{\mathbf{s}})=\frac{1}{Z\left(\hat{\mathbf{s}}\right)}\exp\left\{\mathbb{E}\left[\ln p_{\mathbf{t},\boldsymbol{\theta},\mathbf{w},\mathbf{s}}\left(\mathbf{t},\boldsymbol{\theta},\mathbf{w},\hat{\mathbf{s}};\boldsymbol{\beta}\right)\right]\right\}, \tag{4.46}$$

where $p_{\mathbf{t},\boldsymbol{\theta},\mathbf{w},\mathbf{s}}$ is given in (4.37) and $Z(\hat{\mathbf{s}})$ is the normalising constant resulting from integrating over $\mathbf{w}$. We then write

$$\mathcal{L}\left(q_{\mathbf{w}|\mathbf{t},\mathbf{s}}, \hat{\mathbf{s}}\right) = -\mathrm{KL}\left(q_{\mathbf{w}|\mathbf{t},\mathbf{s}}, g\right) + \ln Z\left(\hat{\mathbf{s}}\right) + \mathrm{Const.} \tag{4.47}$$

For an arbitrary $\hat{\mathbf{s}}$, $\mathcal{L}\left(q_{\mathbf{w}|\mathbf{t},\mathbf{s}}, \hat{\mathbf{s}}\right)$ reaches its maximum when the Kullback–Leibler divergence reaches zero. $\mathcal{L}\left(q_{\mathbf{w}|\mathbf{t},\mathbf{s}}, \hat{\mathbf{s}}\right)$ is thus maximised when

$$q_{\mathbf{w}|\mathbf{t},\mathbf{s}}\left(\mathbf{w}|\mathbf{t},\hat{\mathbf{s}}\right) = g(\mathbf{w};\hat{\mathbf{s}}) \quad \text{and} \quad \hat{\mathbf{s}} = \underset{s}{\mathrm{argmax}} \ln Z\left(\mathbf{s}\right). \tag{4.48}$$

Working out the solution (4.48) by plugging in the joint likelihood (4.37) together with the model likelihood (4.35) and the Bernoulli-Gaussian process (4.32), we get an expression for $q_{\mathbf{w}|\mathbf{t},\mathbf{s}}\left(\mathbf{w}|\mathbf{t},\hat{\mathbf{s}}\right)$

$$q_{\mathbf{w}|\mathbf{t},\mathbf{s}}\left(\mathbf{w}|\mathbf{t},\hat{\mathbf{s}}\right) = \mathcal{CN}\left(\mathbf{w}_{\hat{\mathbf{s}}}\middle|\hat{\mathbf{w}}_{\hat{\mathbf{s}}}, \hat{\mathbf{C}}_{\hat{\mathbf{s}}}\right) \prod_{i\notin\hat{\mathbf{s}}} \delta\left(w_i\right), \tag{4.49}$$

where the mean and covariance matrix are given as

$$\hat{\mathbf{w}}_{\hat{\mathbf{s}}} = \nu^{-1}\hat{\mathbf{C}}_{\hat{\mathbf{s}}}\mathbf{h}_{\hat{\mathbf{s}}} \quad \text{and} \quad \hat{\mathbf{C}}_{\hat{\mathbf{s}}} = \nu\left(\mathbf{J}_{\hat{\mathbf{s}}} + \frac{\nu}{\tau}\mathbf{I}\right)^{-1}, \tag{4.50}$$

respectively. The vector $\mathbf{h}$ is defined as $\left[\boldsymbol{\phi}\left(\hat{\theta}_1\right)^H\mathbf{t}, ..., \boldsymbol{\phi}\left(\hat{\theta}_M\right)^H\mathbf{t}\right]^T$ and the matrix $\mathbf{J}$ is defined to have diagonal elements $J_{i,i} = N$ and off-diagonal elements $J_{i,j} = \boldsymbol{\phi}\left(\hat{\theta}_i\right)^H\boldsymbol{\phi}\left(\hat{\theta}_j\right)$, $i,j = 1,...,M$, $i \neq j$. Recalling that we have assumed that the factors for the support have all their probability mass at the estimator of the support $\hat{\mathbf{s}}$. I.e., $q_{\mathbf{s}|\mathbf{t}}\left(\mathbf{s}|\mathbf{t}\right) = \delta\left(\mathbf{s}-\hat{\mathbf{s}}\right)$.We can thus write the posterior on the weights as

$$q_{\mathbf{w}|\mathbf{t}}\left(\mathbf{w}|\mathbf{t}\right) = q_{\mathbf{w}|\mathbf{t},\mathbf{s}}\left(\mathbf{w}|\mathbf{t},\hat{\mathbf{s}}\right) = \mathcal{CN}\left(\mathbf{w}_{\hat{\mathbf{s}}}\middle|\hat{\mathbf{w}}_{\hat{\mathbf{s}}}, \hat{\mathbf{C}}_{\hat{\mathbf{s}}}\right) \prod_{i\notin\hat{\mathbf{s}}} \delta\left(w_i\right), \tag{4.51}$$

The expression for $\hat{\mathbf{w}}_{\hat{\mathbf{s}}}$ in (4.50) therefore denotes the estimate of the mean of the posterior distribution of the weights. To get the estimate for the support, we need to solve $\hat{\mathbf{s}} = \mathrm{argmax}_s \ln Z\left(\mathbf{s}\right)$. This problem is known to be NP-hard [49], in VALSE we therefore use a single most likely replacement heuristic to find an approximate solution. This heuristic is also proposed in [49].

Finally, we turn to the inference of the parameters in the parameter vector $\boldsymbol{\beta}$. The evidence lower bound of the variational Bayesian approximation is given by

$$\mathcal{L}\left(\boldsymbol{\beta}\right) = \mathbb{E}_{q_{\boldsymbol{\theta},\mathbf{w},\mathbf{s},|\mathbf{t}}}\left[\ln p_{\mathbf{t},\boldsymbol{\theta},\mathbf{w},\mathbf{s}}\left(\mathbf{t},\theta_i,\boldsymbol{\theta}_{\sim i},\mathbf{w},\mathbf{s};\boldsymbol{\beta}\right)\right] + \mathrm{Const.} \tag{4.52}$$

Filling in the joint pdf (4.37), the variational Bayesian factorisation (4.38) and using the model likelihood (4.35) together with the defined priors (4.32) and (4.34), we get

$$\begin{aligned}\mathcal{L}\left(\boldsymbol{\beta}\right) =& \frac{1}{\nu}\left[2\mathcal{R}\left\{\hat{\mathbf{w}}_{\hat{\mathbf{s}}}^H\mathbf{h}_{\hat{\mathbf{s}}}\right\} - \hat{\mathbf{w}}_{\hat{\mathbf{s}}}^H\mathbf{J}_{\hat{\mathbf{s}}}\hat{\mathbf{w}}_{\hat{\mathbf{s}}} - \mathbf{t}^H\mathbf{t} - \mathrm{tr}\left\{\mathbf{J}_{\hat{\mathbf{s}}}\hat{\mathbf{C}}_{\hat{\mathbf{s}}}\right\}\right] - N\ln\nu - \frac{1}{\tau}\left[\hat{\mathbf{w}}_{\hat{\mathbf{s}}}^H\hat{\mathbf{w}}_{\hat{\mathbf{s}}} + \mathrm{tr}\left\{\hat{\mathbf{C}}_{\hat{\mathbf{s}}}\right\}\right] \\ & - \|\hat{\mathbf{s}}\|_0\ln\tau + \|\hat{\mathbf{s}}\|_0\ln\rho + \left(M - \|\hat{\mathbf{s}}\|_0\right)\ln\left\{1-\rho\right\} + \mathrm{Const.}\end{aligned} \tag{4.53}$$

To get the estimates for the parameters, we take the derivative of $\mathcal{L}\left(\boldsymbol{\beta}\right)$ with respect to each individual parameter in the parameter vector and solve for zero. The estimates $\hat{\nu}$, $\hat{\rho}$, and $\hat{\tau}$ are then given by

$$\hat{\nu} = \frac{1}{N}\left\|\mathbf{t} - \sum_{i\in\hat{\mathbf{s}}}\hat{w}_i\boldsymbol{\phi}\left(\theta_i\right)\right\|_2^2 + \frac{1}{N}\mathrm{tr}\left\{\mathbf{J}_{\hat{\mathbf{s}}}\hat{\mathbf{C}}_{\hat{\mathbf{s}}}\right\} + \sum_{i\in\hat{\mathbf{s}}}\left|\hat{w}_i\right|^2\left(1 - \frac{\left\|\boldsymbol{\phi}\left(\hat{\theta}_i\right)\right\|}{N}\right), \tag{4.54}$$

$$\hat{\rho} = \frac{\|\hat{\mathbf{s}}\|_0}{M}, \quad \text{and} \quad \hat{\tau} = \frac{\hat{\mathbf{w}}_{\hat{\mathbf{s}}}^H\hat{\mathbf{w}}_{\hat{\mathbf{s}}} + \mathrm{tr}\left\{\hat{\mathbf{C}}_{\hat{\mathbf{s}}}\right\}}{\|\hat{\mathbf{s}}\|_0}, \tag{4.55}$$

respectively.

We now have all steps required for the iterative scheme of the VALSE algorithms. The VALSE algorithm iteratively updates all estimates. First, we update the estimates of the frequencies by using the heuristic of [24] to approximate (4.43). The heuristic matches the characteristic function of (4.43) with the characteristic function of a mixture of von Mises distributions. We then turn to the weights and the support of the model used in the VALSE algorithm and update the weights and the covariance matrix of the weights using (4.50). The support is updated with the single most likely replacement algorithm of [49]. Finally, we update the estimates of the parameters using (4.54) and (4.55). We keep iterating until some convergence criterion is met. E.g., we could stop iterating when the changes in the reconstructed signal are below some threshold.

5

Extension to ambiguity aware relevance vector machine and incorporation of prior information

After having addressed the off-grid problem, we will now build out the framework to the ambiguity aware relevance vector machine. for the extension to the ambiguity aware relevance vector machine, we first propose a frequentist test that tests whether a target returned by the relevance vector machine can be statistically significantly discerned from ambiguities or noise. We then take a Bayesian view and incorporate prior information, first by designing a Bayesian likelihood ratio test and then by computing a fully Bayesian posterior.

5.1. Frequentist test

To test whether a given target can be statistically significantly separated from ambiguities or noise, we design a generalised likelihood ratio test (GLRT). In a generalised likelihood ratio test, we decide for the alternative hypothesis, $\mathcal{H}_1$, if the likelihood ratio, $L(\mathbf{x})$, exceeds a threshold λ.

As shown in the Neyman-Pearson lemma [50], the likelihood ratio test is the most powerful test for a given false alarm probability. I.e., the test has the highest probability of detection for a given probability of a false alarm.

For our test, we look at the output of a relevance vector machine and for each component, we test whether we can reject the null hypothesis which states that the component under test is still ambiguous. As the relevance vector machine returns all relevance vectors in a matrix $\mathbf{\Phi}$, the corresponding relevance vector under test is denoted by $\boldsymbol{\phi}_t$.

$\mathbf{\Theta}_0$ is the vector space containing vectors that could possibly be ambiguities of the relevance vector under test.

$\mathbf{\Theta}_{tot}$ is then defined as the vector space that contains all possible ambiguities as well as the relevance vector under test. I.e., $\mathbf{\Theta}_{tot} = \mathbf{\Theta}_0 \cup \boldsymbol{\phi}_t$.

We write the likelihood ratio test as

$$L(\mathbf{t}) = \frac{\max_{\boldsymbol{\phi} \in \mathbf{\Theta}_{tot}} p\left(\mathbf{t} \mid \mathbf{w}, \sigma^2\right)}{\max_{\boldsymbol{\phi} \in \mathbf{\Theta}_0} p\left(\mathbf{t} \mid \mathbf{w}, \sigma^2\right)} > \lambda, \tag{5.1}$$

where $p\left(\mathbf{t} \mid \mathbf{w}, \sigma^2\right)$ is the total data likelihood given in (3.7), as used in the relevance vector machine.

We essentially test whether there is a statistically significant difference in likelihoods when we include the relevance vector under test. I.e., the relevance vector under test can be distinguished from its possible ambiguities.

Considering the event that the relevance vector under test does not represent a target, but only noise. Then the test will test this spurious relevance vector against other realisations of the same noise. This implies that the test is still useful even if we are testing a spurious relevance vector.

To get a decision rule, we write out the likelihood ratio, $L(\mathbf{t})$, by filling in $p\left(\mathbf{t} \mid \mathbf{w}, \sigma^2\right)$ as

$$L(\mathbf{t}) = \frac{\max_{\boldsymbol{\phi} \in \boldsymbol{\Theta}_{tot}} \frac{1}{\pi^N |\sigma^2 \mathbf{I}|} \exp\left\{-\frac{1}{\sigma^2} \|\mathbf{t} - \boldsymbol{\Phi}\mathbf{w}\|_2^2\right\}}{\max_{\boldsymbol{\phi} \in \boldsymbol{\Theta}_0} \frac{1}{\pi^N |\sigma^2 \mathbf{I}|} \exp\left\{-\frac{1}{\sigma^2} \|\mathbf{t} - \boldsymbol{\Phi}\mathbf{w}\|_2^2\right\}} > \lambda. \tag{5.2}$$

We define $\hat{\boldsymbol{\Phi}}_{tot}$ and $\hat{\boldsymbol{\Phi}}_0$ as the matrix of relevance vectors that gives the maximum likelihood when optimised within $\boldsymbol{\Theta}_{tot}$ and $\boldsymbol{\Theta}_0$, respectively. The likelihood ratio can then be written as

$$\begin{aligned}
L(\mathbf{t}) &= \frac{\frac{1}{\pi^N |\sigma^2 \mathbf{I}|} \exp\left\{-\frac{1}{\sigma^2} \left\|\mathbf{t} - \hat{\boldsymbol{\Phi}}_{tot}\mathbf{w}\right\|_2^2\right\}}{\frac{1}{\pi^N |\sigma^2 \mathbf{I}|} \exp\left\{-\frac{1}{\sigma^2} \left\|\mathbf{t} - \hat{\boldsymbol{\Phi}}_0\mathbf{w}\right\|_2^2\right\}} > \lambda \\
\Rightarrow L(\mathbf{t}) &= \exp\left\{-\frac{1}{\sigma^2}\left(\left\|\mathbf{t} - \hat{\boldsymbol{\Phi}}_{tot}\mathbf{w}\right\|_2^2 - \left\|\mathbf{t} - \hat{\boldsymbol{\Phi}}_0\mathbf{w}\right\|_2^2\right)\right\} > \lambda \\
\Rightarrow L(\mathbf{t}) &= \exp\left\{-\frac{1}{\sigma^2}\left(\left(\mathbf{t} - \hat{\boldsymbol{\Phi}}_{tot}\mathbf{w}\right)^H \left(\mathbf{t} - \hat{\boldsymbol{\Phi}}_{tot}\mathbf{w}\right) - \left(\mathbf{t} - \hat{\boldsymbol{\Phi}}_0\mathbf{w}\right)^H \left(\mathbf{t} - \hat{\boldsymbol{\Phi}}_0\mathbf{w}\right)\right)\right\} > \lambda \\
\Rightarrow L(\mathbf{t}) &= \exp\left\{-\frac{1}{\sigma^2}\left(\|\mathbf{t}\|_2^2 + \left\|\hat{\boldsymbol{\Phi}}_{tot}\mathbf{w}\right\|_2^2 - 2\Re\left\{\mathbf{t}^H \hat{\boldsymbol{\Phi}}_{tot}\mathbf{w}\right\} - \|\mathbf{t}\|_2^2 - \left\|\hat{\boldsymbol{\Phi}}_0\mathbf{w}\right\|_2^2 + 2\Re\left\{\mathbf{t}^H \hat{\boldsymbol{\Phi}}_0\mathbf{w}\right\}\right)\right\} > \lambda \\
\Rightarrow L(\mathbf{t}) &= \exp\left\{-\frac{1}{\sigma^2}\left(\left\|\hat{\boldsymbol{\Phi}}_{tot}\mathbf{w}\right\|_2^2 - \left\|\hat{\boldsymbol{\Phi}}_0\mathbf{w}\right\|_2^2 - 2\Re\left\{\mathbf{t}^H \left(\hat{\boldsymbol{\Phi}}_{tot} - \hat{\boldsymbol{\Phi}}_0\right)\mathbf{w}\right\}\right)\right\} > \lambda.
\end{aligned} \tag{5.3}$$

We then take the logarithm of the likelihood ratio, which we can do as the logarithm is a monotonically increasing function.

$$\begin{aligned}
l(\mathbf{t}) = \log\{L(\mathbf{t})\} &= -\left\|\hat{\boldsymbol{\Phi}}_{tot}\mathbf{w}\right\|_2^2 + \left\|\hat{\boldsymbol{\Phi}}_0\mathbf{w}\right\|_2^2 + 2\Re\left\{\mathbf{t}^H \left(\hat{\boldsymbol{\Phi}}_{tot} - \hat{\boldsymbol{\Phi}}_0\right)\mathbf{w}\right\} > \sigma^2 \log\{\lambda\} \\
\Rightarrow l(\mathbf{t}) &= \Re\left\{\mathbf{t}^H \left(\hat{\boldsymbol{\Phi}}_{tot} - \hat{\boldsymbol{\Phi}}_0\right)\mathbf{w}\right\} > \frac{1}{2}\sigma^2 \log\{\lambda\} + \frac{1}{2}\left\|\hat{\boldsymbol{\Phi}}_{tot}\mathbf{w}\right\|_2^2 - \frac{1}{2}\left\|\hat{\boldsymbol{\Phi}}_0\mathbf{w}\right\|_2^2 \\
\Rightarrow l(\mathbf{t}) &= \Re\left\{\mathbf{t}^H \left(\hat{\boldsymbol{\Phi}}_{tot} - \hat{\boldsymbol{\Phi}}_0\right)\mathbf{w}\right\} > \lambda',
\end{aligned} \tag{5.4}$$

where we have defined $\lambda' = \frac{1}{2}\sigma^2 \log\{\lambda\} + \frac{1}{2}\left\|\hat{\boldsymbol{\Phi}}_{tot}\mathbf{w}\right\|_2^2 - \frac{1}{2}\left\|\hat{\boldsymbol{\Phi}}_0\mathbf{w}\right\|_2^2$.

As $\hat{\boldsymbol{\Phi}}_{tot}$ and $\hat{\boldsymbol{\Phi}}_0$ have identical columns except for the column under test, we can write

$$l(\mathbf{t}) = \Re\left\{\mathbf{t}\left(\hat{\boldsymbol{\phi}}_{tot} - \hat{\boldsymbol{\phi}}_0\right)\mathbf{w}_t\right\} > \lambda', \tag{5.5}$$

where the subscript t corresponds to the relevance vector that is under test and the vectors $\hat{\boldsymbol{\phi}}_{tot}$ and $\hat{\boldsymbol{\phi}}_0$ are the vectors that give the maximum likelihood when optimised within $\boldsymbol{\Theta}_{tot}$ and $\boldsymbol{\Theta}_0$, respectively.

To set the threshold for a given false alarm rate, we need to characterise the distribution of $l(\mathbf{t})$ under $\mathcal{H}_0$. We note that $l(\mathbf{t})$ is the real part of a complex Gaussian random variable, which is by definition a Gaussian random variable [39]. A Gaussian random variable is completely characterised by its mean and variance. Under $\mathcal{H}_0$, we express the expectation of $l(\mathbf{t})$ as

$$\begin{aligned}
\mathbb{E}\left[l(\mathbf{t}); \mathcal{H}_0\right] &= \mathbb{E}\left[\Re\left\{\mathbf{t}^H \left(\hat{\boldsymbol{\phi}}_{tot} - \hat{\boldsymbol{\phi}}_0\right)\mathbf{w}_t\right\}\right] \\
&= \mathbb{E}\left[\Re\left\{\sum_{j=1}^{N} \mathbf{t}_j^H \left(\hat{\boldsymbol{\phi}}_{tot} - \hat{\boldsymbol{\phi}}_0\right)_j \mathbf{w}_t\right\}\right].
\end{aligned} \tag{5.6}$$

We then use the fact that the expectation of the real part of a complex Gaussian random variable is equal to the real part of the expectation.

$$\begin{aligned}\mathbb{E}\left[l\left(\mathbf{t}\right);\mathcal{H}_0\right] &= \Re\left\{\mathbb{E}\left[\sum_{j=1}^{N}\mathbf{t}_j^H\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)_j\mathbf{w}_t\right]\right\}\\ &= \Re\left\{\sum_{j=1}^{N}\mathbb{E}\left[\mathbf{t}_j^H\right]\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)_j\mathbf{w}_t\right\}.\end{aligned} \tag{5.7}$$

Under $\mathcal{H}_0$, the expectation of $\mathbf{t}_j$ is the j'th row of $\hat{\boldsymbol{\Phi}}_0$ multiplied with $\mathbf{w}$. So,

$$\begin{aligned}\mathbb{E}\left[l\left(\mathbf{t}\right);\mathcal{H}_0\right] &= \Re\left\{\sum_{j=1}^{N}\left(\hat{\boldsymbol{\Phi}}_{0j}\mathbf{w}\right)^H\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)_j\mathbf{w}_t\right\}\\ &= \Re\left\{\left(\hat{\boldsymbol{\Phi}}_0\mathbf{w}\right)^H\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)\mathbf{w}_t\right\}.\end{aligned} \tag{5.8}$$

For the variance of $l\left(\mathbf{t}\right)$ under $\mathcal{H}_0$, we write

$$\mathrm{var}\left(l\left(\mathbf{t}\right);\mathcal{H}_0\right) = \mathrm{var}\left(\Re\left\{\mathbf{t}^H\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)\mathbf{w}_t\right\}\right). \tag{5.9}$$

Using that the variance of the real part of a complex normal random variable is half of the real part of the variance of the complex normal random variable [39], we write

$$\begin{aligned}\mathrm{var}\left(l\left(\mathbf{t}\right);\mathcal{H}_0\right) &= \frac{1}{2}\Re\left\{\mathrm{var}\left(\mathbf{t}^H\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)\mathbf{w}_t\right)\right\}\\ &= \frac{1}{2}\Re\left\{\mathrm{var}\left(\sum_{j=1}^{N}\mathbf{t}_j^H\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)_j\mathbf{w}_t\right)\right\}\\ &= \frac{1}{2}\Re\left\{\sum_{j=1}^{N}\mathrm{var}\left(\mathbf{t}_j^H\right)\left(\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)_j\mathbf{w}_t\right)^H\left(\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)_j\mathbf{w}_t\right)\right\}\\ &= \frac{1}{2}\Re\left\{\sum_{j=1}^{N}\sigma^2\left(\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)_j\mathbf{w}_t\right)^H\left(\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)_j\mathbf{w}_t\right)\right\}\\ &= \frac{1}{2}\Re\left\{\sigma^2\left(\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)\mathbf{w}_t\right)^H\left(\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)\mathbf{w}_t\right)\right\}.\end{aligned} \tag{5.10}$$

We have now fully characterised the distribution of $l\left(\mathbf{t}\right)$ under $\mathcal{H}_0$ as

$$l\left(\mathbf{t}\right) \overset{\mathcal{H}_0}{\sim} \mathcal{N}\left(\Re\left\{\left(\hat{\boldsymbol{\Phi}}_0\mathbf{w}\right)^H\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)\mathbf{w}_t\right\}, \frac{1}{2}\Re\left\{\sigma^2\left(\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)\mathbf{w}_t\right)^H\left(\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)\mathbf{w}_t\right)\right\}\right). \tag{5.11}$$

The probability of false alarms is then given by

$$P_{FA} = Pr\{l\left(\mathbf{t}\right) > \lambda'; \mathcal{H}_0\} = Q\left(\frac{\lambda' - \Re\left\{\left(\hat{\boldsymbol{\Phi}}_0\mathbf{w}\right)^H\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)\mathbf{w}_t\right\}}{\Re\left\{\sigma^2\left(\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)\mathbf{w}_t\right)^H\left(\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)\mathbf{w}_t\right)\right\}}\right), \tag{5.12}$$

where $Q(\cdot) = 1 - F(\cdot)$ and $F(x)$ is the CDF of a standard normal random variable. Rewriting to get an expression for our threshold, given a certain P_{FA} gives

$$\lambda' = \Re\left\{\sigma^2\left(\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)\mathbf{w}_t\right)^H\left(\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)\mathbf{w}_t\right)\right\}Q^{-1}\left(P_{FA}\right) + \Re\left\{\left(\hat{\boldsymbol{\Phi}}_0\mathbf{w}\right)^H\left(\hat{\boldsymbol{\phi}}_{tot}-\hat{\boldsymbol{\phi}}_0\right)\mathbf{w}_t\right\}. \tag{5.13}$$

5.2. Bayesian likelihood ratio

In the case that we have prior information on the velocity of the target, we would ideally be able to incorporate that information into our framework. This guides us towards a Bayesian approach to the classification problem. Extending the constant false alarm rate detector of section 5.1 to a Bayesian likelihood ratio is quite straightforward. Instead of maximising over the unknown variable as in the generalised likelihood ratio test, we integrate the unknown variable out of the expression. Given some prior on the velocities, denoted by $p(\theta)$, We can integrate out the velocity in a Bayesian likelihood ratio test as

$$\frac{p\left(\mathbf{t};\mathcal{H}_1\right)}{p\left(\mathbf{t};\mathcal{H}_0\right)} = \frac{\int_{\boldsymbol{\phi}\in\boldsymbol{\Theta}_{tot}} p\left(\mathbf{x} \mid \boldsymbol{\phi};\mathcal{H}_1\right) p\left(\boldsymbol{\phi}\right) \mathrm{d}\boldsymbol{\phi}}{\int_{\boldsymbol{\phi}\in\boldsymbol{\Theta}_0} p\left(\mathbf{x} \mid \boldsymbol{\phi};\mathcal{H}_0\right) p\left(\boldsymbol{\phi}\right) \mathrm{d}\boldsymbol{\phi}} > \lambda. \tag{5.14}$$

To make this applicable to our problem, we design a test similar to the test of section 5.1. Like in section 5.1, we look at the output of the relevance vector machine and set up a test whether we can reject the null hypothesis which states that the component under test is still ambiguous. As the relevance vector machine returns all relevance vectors in a matrix $\boldsymbol{\Phi}$, the corresponding relevance vector under test is denoted by $\boldsymbol{\phi}_t$.

$\boldsymbol{\Theta}_0$ is the vector space containing vectors that could possibly be ambiguities of the relevance vector under test.

$\boldsymbol{\Theta}_{tot}$ is then defined as the vector space containing all possible ambiguities as well as the relevance vector under test. I.e., $\boldsymbol{\Theta}_{tot} = \boldsymbol{\Theta}_0 \cup \boldsymbol{\phi}_t$.

We then generate the probability mass function $p_{\theta_0}(\boldsymbol{\phi})$ by plugging all possible ambiguities into $p(\theta_0)$ and normalising. Then, $p_{\theta_{tot}}(\boldsymbol{\phi})$ is generated by plugging all possible ambiguities as well as $\boldsymbol{\phi}_t$ into $p(\theta_0)$ and normalising.

The Bayesian likelihood ratio that can be applied to the outputs of a relevance vector machine is then given by

$$\frac{\sum_{\boldsymbol{\phi}\in\boldsymbol{\Theta}_{tot}} p\left(\mathbf{x} \mid \boldsymbol{\phi};\mathbf{w},\sigma^2\right) P_{\theta_{tot}}(\phi)}{\sum_{\boldsymbol{\phi}\in\boldsymbol{\Theta}_0} p\left(\mathbf{x} \mid \boldsymbol{\phi};\mathbf{w},\sigma^2\right) P_{\theta_0}(\phi)} > \lambda. \tag{5.15}$$

This Bayesian likelihood ratio is easy to implement and allows us to incorporate prior information. However, the result is not easily interpretable. We cannot find a clear distribution of the ratio. Jeffreys [51] provides a table to interpret such a ratio. This table is however based on experience/expert knowledge and is not rigorously derived.

5.3. Estimation of a posterior distribution

Since we are taking a Bayesian perspective to the problem, we will work towards a posterior on the relevance vector machine components. Let us consider the case where we have a collection of classes C. We then define a prior on the probability of each type of class $c \in C$, denoted by $p(c)$

We now define $\boldsymbol{\Theta}$ as the collection of relevance vectors that could possibly be ambiguities of the relevance vector under test, as well as the relevance vector itself. The probability of distribution for the velocity of a certain class is then modelled by $p(\boldsymbol{\phi}|c)$, where $\boldsymbol{\phi} \in \boldsymbol{\Theta}$.

By applying Bayes rule, we can rewrite the joint probability as

$$p\left(c,\boldsymbol{\phi},\mathbf{t},\mathbf{w},\sigma^2\right) = p\left(c,\boldsymbol{\phi}|\mathbf{t},\mathbf{w},\sigma^2\right) p\left(\mathbf{t},\mathbf{w},\sigma^2\right). \tag{5.16}$$

We then also write

$$p\left(c,\boldsymbol{\phi},\mathbf{t},\mathbf{w},\sigma^2\right) = p\left(\mathbf{t}|c,\boldsymbol{\phi},\mathbf{w},\sigma^2\right) p\left(c,\boldsymbol{\phi},\mathbf{w},\sigma^2\right). \tag{5.17}$$

By equating the right hand side of (5.16) to (5.17), we can write

$$p\left(c,\boldsymbol{\phi}|\mathbf{t},\mathbf{w},\sigma^2\right) = \frac{p\left(\mathbf{t}|c,\boldsymbol{\phi},\mathbf{w},\sigma^2\right) p\left(c,\boldsymbol{\phi},\mathbf{w},\sigma^2\right)}{p\left(\mathbf{t},\mathbf{w},\sigma^2\right)}. \tag{5.18}$$

In our current model, the data $\mathbf{t}$ conditioned on $\boldsymbol{\phi}$, $\mathbf{w}$ and σ^2 will not look different coming from a different class c. I.e., $p\left(\mathbf{t}|c,\boldsymbol{\phi},\mathbf{w},\sigma^2\right) = p\left(\mathbf{t}|\boldsymbol{\phi},\mathbf{w},\sigma^2\right)$. We thus write

$$\begin{aligned} p\left(c,\boldsymbol{\phi}|\mathbf{t},\mathbf{w},\sigma^2\right) &= \frac{p\left(\mathbf{t}|\boldsymbol{\phi},\mathbf{w},\sigma^2\right)p\left(c,\boldsymbol{\phi},\mathbf{w},\sigma^2\right)}{p\left(\mathbf{t},\mathbf{w},\sigma^2\right)} \\ &= \frac{p\left(\mathbf{t}|\boldsymbol{\phi},\mathbf{w},\sigma^2\right)p\left(\boldsymbol{\phi}|c,\mathbf{w},\sigma^2\right)p\left(c,\mathbf{w},\sigma^2\right)}{p\left(\mathbf{t}|\mathbf{w},\sigma^2\right)p\left(\mathbf{w},\sigma^2\right)} \\ &= \frac{p\left(\mathbf{t}|\boldsymbol{\phi},\mathbf{w},\sigma^2\right)p\left(\boldsymbol{\phi}|c,\mathbf{w},\sigma^2\right)p\left(c|\mathbf{w},\sigma^2\right)p\left(\mathbf{w},\sigma^2\right)}{p\left(\mathbf{t}|\mathbf{w},\sigma^2\right)p\left(\mathbf{w},\sigma^2\right)} \\ &= \frac{p\left(\mathbf{t}|\boldsymbol{\phi},\mathbf{w},\sigma^2\right)p\left(\boldsymbol{\phi}|c,\mathbf{w},\sigma^2\right)p\left(c|\mathbf{w},\sigma^2\right)}{p\left(\mathbf{t}|\mathbf{w},\sigma^2\right)}. \end{aligned} \tag{5.19}$$

Assuming that we have specified $p\left(\boldsymbol{\phi}|c\right)$ and $p\left(c\right)$ to both be independent of $\mathbf{w}$ and σ^2, we write

$$p\left(c,\boldsymbol{\phi}|\mathbf{t},\mathbf{w},\sigma^2\right) = \frac{p\left(\mathbf{t}|\boldsymbol{\phi},\mathbf{w},\sigma^2\right)p\left(\boldsymbol{\phi}|c\right)p\left(c\right)}{p\left(\mathbf{t}|\mathbf{w},\sigma^2\right)}. \tag{5.20}$$

If we also take into account the probability that the vector under test is no target at all and thus spuriously given by the relevance vector machine, we should include that probability in the prior as well. We can model this with a null vector, $\mathbf{0}$, for $\boldsymbol{\phi}$. If we define a Bernoulli random variable with probability ρ of a target being anywhere, we can then write a new $p\left(\boldsymbol{\phi}|c\right)$ as.

$$p_{new}\left(\boldsymbol{\phi}|c\right) = \begin{cases} \rho p\left(\boldsymbol{\phi}|c\right) & \text{for } \boldsymbol{\phi} \in \boldsymbol{\Theta} \\ \left(1-\rho\right) & \text{for } \boldsymbol{\phi} = \mathbf{0} \end{cases} \tag{5.21}$$

Leading to the new posterior

$$p\left(c,\boldsymbol{\phi}|\mathbf{t},\mathbf{w},\sigma^2\right) = \begin{cases} \frac{p\left(\mathbf{t}|\boldsymbol{\phi},\mathbf{w},\sigma^2\right)\rho p\left(\boldsymbol{\phi}|c\right)p(c)}{p\left(\mathbf{t}|\mathbf{w},\sigma^2\right)} & \text{for } \boldsymbol{\phi} \in \boldsymbol{\Theta} \\ \frac{p\left(\mathbf{t}|\boldsymbol{\phi},\mathbf{w},\sigma^2\right)\left(1-\rho\right)p(c)}{p\left(\mathbf{t}|\mathbf{w},\sigma^2\right)} & \text{for } \boldsymbol{\phi} = \mathbf{0} \end{cases} \tag{5.22}$$

In the case of no target, it does not make sense to consider the class of target. Therefore, we marginalise over the class for the case where $\boldsymbol{\phi} = \mathbf{0}$ to get the posterior probability of no target

$$p\left(\mathbf{0}|\mathbf{t},\mathbf{w},\sigma^2\right) = \sum_{c\in C}\frac{p\left(\mathbf{t}|\mathbf{0},\mathbf{w},\sigma^2\right)\left(1-\rho\right)p\left(c\right)}{p\left(\mathbf{t}|\mathbf{w},\sigma^2\right)} = \frac{p\left(\mathbf{t}|\mathbf{0},\mathbf{w},\sigma^2\right)\left(1-\rho\right)}{p\left(\mathbf{t}|\mathbf{w},\sigma^2\right)}. \tag{5.23}$$

Allowing us to express the posterior as

$$p\left(c,\boldsymbol{\phi}|\mathbf{t},\mathbf{w},\sigma^2\right) = \begin{cases} \frac{p\left(\mathbf{t}|\boldsymbol{\phi},\mathbf{w},\sigma^2\right)\rho p\left(\boldsymbol{\phi}|c\right)p(c)}{p\left(\mathbf{t}|\mathbf{w},\sigma^2\right)} & \text{for } \boldsymbol{\phi} \in \boldsymbol{\Theta}, c \in C \\ \frac{p\left(\mathbf{t}|\mathbf{0},\mathbf{w},\sigma^2\right)\left(1-\rho\right)}{p\left(\mathbf{t}|\mathbf{w},\sigma^2\right)} & \text{for } \boldsymbol{\phi} = \mathbf{0} \end{cases} \tag{5.24}$$

Giving us the posterior of the relevance vector under test that includes the probability of the target being spurious. Note that the outcome space of the posterior is quite unusual as it contains all combinations of $\boldsymbol{\phi} \in \boldsymbol{\Theta}$ and $c \in C$ as well as the class independent outcome of $\boldsymbol{\phi} = \mathbf{0}$. If we are only interested in the posterior of the velocity or the posterior of the class, we can marginalise over the other variable to get the posterior of the variable of interest.

6

Simulation study

We set up four simulation studies to evaluate the performance of the proposed framework. The first simulation study compares the general performance of the framework with two benchmark algorithms. In the second simulation study, we simulate particularly difficult cases to solve and compare the performance of the framework with the performance of the benchmarks. In the third simulation study, we evaluate the performance of the framework when prior information is available. Finally, in the fourth simulation study, we consider a simple single target case and solve the problem by using an MCMC sampler, then compare the MCMC output with the output of the framework.

6.1. Data generating process

For all simulations, we use the same data-generating process. The data-generating process models observation data containing the Doppler response of M targets for B bursts with N_p pulses. The observed data will thus result in a total amount of B time-series of length N_p, written as

$$t_b[i] = \sum_{m=1}^{M} \alpha_m \exp\left\{\frac{j2\pi 2T_b i\theta_m f_c}{c}\right\} + \varepsilon, \tag{6.1}$$

where i denotes the time index, $i \in \{1, 2, \dots, N_p\}$. The pulse repetition time corresponding to burst b is denoted by T_b. the amplitude of the received reflection and the velocity corresponding to the m'th target are denoted by α_m and θ_m, respectively. The carrier frequency of the radar is denoted by f_c and c denotes the speed of light. The noise term is denoted by ε.

6.2. Benchmarks

We introduce two methods in this section to compare the performance of the framework with existing methods.

6.2.1. Matched filter

We implement a coincidence-based algorithm based on matched filter detections. Such a coincidence-based algorithm is essentially what a lot of the hit-based methods implement in different techniques [3], [5], [7]. In these coincidence types of algorithms, we run a matched filter on each individual burst. Implementation details of a matched filter detector are found in, e.g., [52]. If there are any hits that occur in all matched filters for the individual bursts at the same Doppler velocity, we choose those hits as final hits.

6.2.2. Feedback N-signal Orthogonal Matching Pursuit

The Feedback N-signal Orthogonal Matching Pursuit (FN-OMP) algorithm of Aouchiche *et al.* [21] is an OMP algorithm that uses a Levenberg Marquardt algorithm in each iteration to refine the estimates and allow for non-coherent processing. The algorithm is given in pseudocode in Algorithm 1.

Algorithm 1 Feedback N-signal Orthogonal Matching Pursuit

Require:
Observed data for the B individual bursts $\mathbf{t_b}\ \forall b \in \{1, ..., B\}$
Dictionaries with relevance vectors $\boldsymbol{\Phi}_b = \left[\boldsymbol{\phi_b}\left(\theta_1\right), \boldsymbol{\phi_b}\left(\theta_2\right), ..., \boldsymbol{\phi_b}\left(\theta_M\right)\right]\ \forall b \in \{1, ..., B\}$
Threshold τ
Ensure:
Estimated Doppler velocities $\hat{\boldsymbol{\theta}}$
Estimated weights $\hat{\mathbf{w}}_b\ \forall b \in \{1, ..., B\}$
Initialise:
Residuals $\boldsymbol{\epsilon}_b = \mathbf{t}_B\ \forall b \in \{1, ..., B\}$
Set of Doppler velocities $\hat{\boldsymbol{\theta}} \leftarrow \emptyset$
Sets of weights $\hat{\mathbf{w}}_b \leftarrow \emptyset\ \forall b \in \{1, ..., B\}$
while $\max_{\theta_1 \le \theta \le \theta_M} \sum_{b=1}^{B} \frac{\left|\boldsymbol{\phi_b}(\theta)^H \boldsymbol{\epsilon}_b\right|}{\sqrt{\boldsymbol{\phi_b}(\theta)^H \boldsymbol{\phi_b}(\theta)}} \ge \tau$ **do**

1. Add new Doppler velocity to set $\hat{\boldsymbol{\theta}} \leftarrow \hat{\boldsymbol{\theta}} \cup \operatorname{argmax}_{\theta_1 \le \theta \le \theta_M} \sum_{b=1}^{B} \frac{\left|\boldsymbol{\phi_b}(\theta)^H \boldsymbol{\epsilon}_b\right|}{\sqrt{\boldsymbol{\phi_b}(\theta)^H \boldsymbol{\phi_b}(\theta)}}$
2. Initialise B new weights $\hat{\mathbf{w}} \leftarrow \hat{\mathbf{w}}_b \cup w_b^{new}\ \forall b \in \{1, ..., B\}$
3. Refine all estimates for the Doppler frequencies and weights using a Levenberg–Marquardt algorithm, using the old estimates as starting points $\left[\hat{\boldsymbol{\theta}}, \hat{\mathbf{w}}\right] = \operatorname{argmin}_{\boldsymbol{\theta}, \hat{\mathbf{w}}} \sum_{b=1}^{B} \left|\mathbf{t}_b - \boldsymbol{\Phi}_b\left(\hat{\boldsymbol{\theta}}\right)\hat{\mathbf{w}}_b\right|^2$
4. Calculate new residuals $\boldsymbol{\epsilon}_b = \boldsymbol{\Phi}_b\left(\hat{\boldsymbol{\theta}}\right)\hat{\mathbf{w}}_b\ \forall b \in \{1, ..., B\}$

end while

6.3. Simulation of general cases

To compare general performance, we perform a Monte Carlo simulation of general cases. In each run, we simulate two targets at random positions and give the data to six different algorithms. Four of the algorithms are formed by the framework and the other two are the benchmarks. We then generate a false alarm rate against probability of detection curve of each individual algorithm for different SNRs.

To generate the observed data, we simulate the Doppler responses of two targets for two individual bursts of length 12. This results in two time-series of observed data, as specified by our data generating process (6.1). We draw the amplitudes α_m according to a Swerling 2 fluctuation model. I.e., $\alpha_m \sim Rayleigh(\iota_m)$, where we choose $\iota_m = 0.6\ \forall m \in \{1, 2\}$. We set the carrier frequency $f_c = 3 \cdot 10^8$ Hz. The PRT of the first burst is $0.5 \cdot 10^{-3}$ s, resulting in a Doppler fold of 30 m/s according to (2.3). The PRT of the second burst is $0.4 \cdot 10^{-3}$ s, resulting in a Doppler fold of 37.5 m/s according to (2.3). Combining those folds, we should theoretically be able to achieve a new Doppler fold equal to the smallest common multiple of the two individual folds. I.e., we get a total Doppler fold of 150 m/s. The two targets are generated randomly according to a uniform distribution. I.e., $\theta_m \sim U(-75, 75)$. For the noise term, we generate i.i.d. complex Gaussian noise with a standard deviation, σ, of 1.04, 0.585, 0.329, or 0.185 to get a signal-to-noise ratio (SNR) of 15, 20, 25, or 30 dB, respectively.

The SNR of a single pulse is defined as the ratio between the expected value of the squared signal and the expected value of the noise squared. I.e.,

$$SNR = \frac{\mathbb{E}\left[\left\|\sum_{m=1}^{M} \alpha_m \exp\left\{\frac{j2\pi 2T_b i\theta_m f_c}{c}\right\}\right\|^2\right]}{\mathbb{E}\left[\varepsilon^2\right]}. \tag{6.2}$$

Due to the fact that the noise is zero mean Gaussian, the expected value squared is equal to the variance of the noise. I.e., $\mathbb{E}\left[\varepsilon^2\right] = \sigma^2$. The absolute value of the exponent is equal to 1, so for a single pulse we can thus write

$$SNR = \frac{\sum_{m=1}^{M} \mathbb{E}\left[\left\|\alpha_m^2\right\|\right]}{\sigma^2}. \tag{6.3}$$

As shown in Appendix D, the non-central second order moment of a Rayleigh distribution is equal to

$2\iota_m^2$. We can thus express the single pulse SNR in dB as

$$SNR = 10\log_{10}\left\{2\sum_{m=1}^{M}\iota_m^2\right\} - 10\log_{10}\left\{\sigma^2\right\}. \tag{6.4}$$

By taking into account the integration gain from the two bursts of twelve pulses, we get a total SNR in dB of

$$SNR_{tot} = 10\log_{10}\left\{2\sum_{m=1}^{M}\iota_m^2\right\} - 10\log_{10}\left\{\sigma^2\right\} + 10\log_{10}\{24\}. \tag{6.5}$$

We give the data to the root-based MRVM and the VALSE-based MRVM, as well as to the two benchmarks. For all algorithms, we define a grid from -75 to 75, with a resolution of one. The first benchmark is a regular matched filter with a coincidence algorithm, where we decide that there is a target when the same gridpoint exceeds the threshold in both bursts. The second benchmark is the FN-OMP algorithm of [21].

In the MRVM based methods, we prune the component from the model when the specific α becomes greater than 10^6. We stop iterating when the sum of the absolute difference between the alphas of the current iteration and the previous iteration is smaller than 10^{-10}, or when we exceed 2000 iterations.

We apply both the GLRT and Bayesian ambiguity awareness extension to the root-based MRVM and the VALSE-based MRVM methods, resulting in four different methods. For the GLRT ambiguity aware relevance vector machines, we set the threshold using (5.13). For the Bayesian ambiguity aware relevance vector machines, we decide that the relevance vector under test is not a target if the posterior probability of no target exceeds the threshold. If the probability of no target is below threshold, we choose the maximum a posteriori (MAP) estimate.

For the matched filter method, the thresholds are set by evaluating the inverse cumulative distribution of a standard normal random variable between 0 and 1 in 1000 steps. To set the thresholds for the FN-OMP method, we evaluate the quantiles of the empirical distribution of the maximum of 150 realisations of the sum of two standard normal distributions. We evaluate these quantiles from 0 to 1 in 1000 steps. When using the extension to GLRT ambiguity awareness, we set the thresholds based on (5.13) with λ ranging from 0 to 1 in 1000 steps. For the Bayesian ambiguity awareness, we again range the threshold from 0 to 1 in 1000 steps.

We run the experiment 1000 times, each time with independently generated data. We gather all the detections produced by the methods and compare them with the generated Doppler velocities. For each iteration, we look at every individual element in the set of generated Doppler velocities and search for a detection that lies within one resolution of the generated Doppler velocities. If we find such a detection, we remove both the Doppler velocity and the detection. Repeating for all generated Doppler velocities, the final elements left in the set of generated Doppler velocities is the collection of missed detections. Similarly, the elements left in the set of detections is the set of false alarms. Using the sets of missed detections and false alarms, we calculate the probability of detection, p_d, and the false alarm rate r_{fa}. The probability of detection is defined as the ratio between the detected targets and the total number of targets. The false alarm rate is the average number of false alarms in a simulation run. This procedure results in 1000 pairs of p_d and r_{fa} that form approximate receiver operating characteristic (ROC) curves. Note that this p_d is not exactly the same as the probability of detection used in most literature. Here, we specifically require the detections to be unambiguous.

Figure 6.1 shows the results of the experiment. For an SNR of 15, 20, 25 and 30 dB, we get six approximations of ROC curves. I.e., for the root-based MRVM methods with GLRT as well as Bayesian ambiguity awareness, the VALSE-based MRVM methods with GLRT and Bayesian ambiguity awareness, the FN-OMP method and, the matched filter method.

From a performance point of view, the VALSE-based MRVM methods tend to perform best. Especially when a high probability of detection is required, the VALSE-based MRVM methods tend to achieve higher probabilities of detection for the lowest false alarm rate.

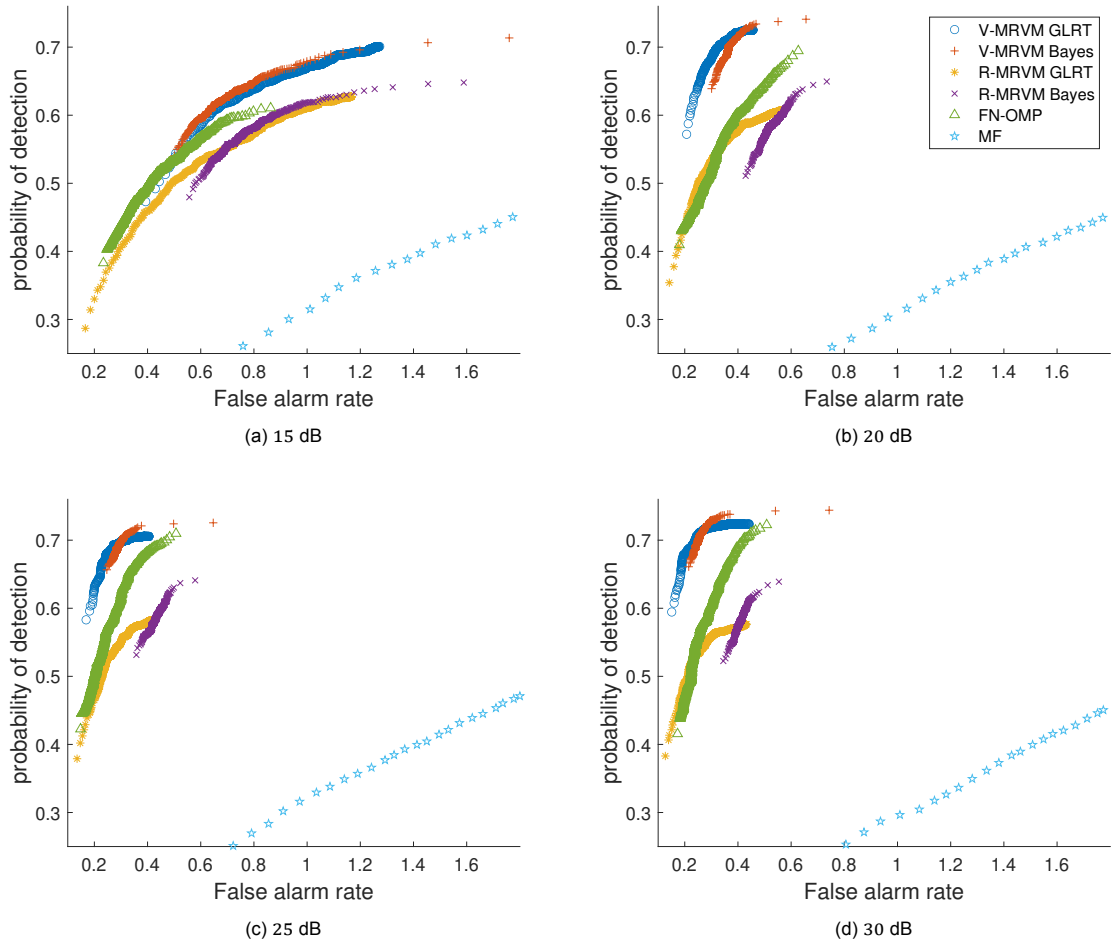


Figure 6.1: Plots of the estimated ROC curves for all methods that have a threshold. Each simulation run has two targets with a Doppler velocity uniformly generated between -75 and 75. The amplitudes of the reflection of the targets follow a Swerling 2 fluctuation model. The legend is shared across all subfigures. The scatterplots corresponds to the following SNR: (a) 15 dB, (b) 20 dB, (c) 25 dB, (d) 30 dB.

For the MRVM-based methods, we generally see that the GLRT ambiguity awareness is preferred when a low false alarm rate is desired. When a high probability of detection is required, the Bayesian ambiguity awareness seems to be more favourable. This effect is visible across all SNRs, but does not seem to hold for the VALSE-MRVM based at an SNR of 15 dB

The general performance of the FN-OMP method tends to be good, but the FN-OMP method is outperformed by the VALSE-MRVM based methods at most points in the ROC curve for the different SNRs. An exception of this trend in performance is in the case of an SNR of 15 dB when a low false alarm rate is required. The FN-OMP method can provide us with the highest probability of detection when such a low false alarm rate is required. We do, however, also see that the Root-based method with GLRT ambiguity awareness outperforms the FN-OMP method when we want to go even lower in terms of false alarm rate.

The matched filter method seems to perform worst in all scenarios. One advantage of the matched filter method is that it can reach higher probabilities of detection, albeit for extremely high false alarm rates. Zoomed-out plots of the estimated ROC curves of the matched filter method are depicted in Figure E.1 of the additional figures given in Appendix E.

When looking at Figure 6.1 from a sensitivity analysis point of view, we notice that there is an obvious limit For the MRVM-based methods. we cannot get a significantly greater probability of detection than the MRVM methods themselves, as the main task of the ambiguity awareness is to reject components.

The Bayesian ambiguity awareness is theoretically able to increase this probability of detection, but this happens only so often in practice. We show an example of this increase in the simulation experiment with prior information below. In the simulation of a general case, there is however no increase in probability of detection achieved across all SNRs, only a significant decrease in false alarm rate.

The FN-OMP does not have such an inherent limit, we do however see that the ROC curve seems to converge towards a similar asymptote in terms of probability of detection.

To compare the plain MRVM methods with the ambiguity aware methods, we repeat the simulation experiment for the plain MRVM methods. As there is no threshold like in the ambiguity awareness to be set in the plain MRVM methods, we only generate one point and not a complete curve.

Table 6.1: Average false alarm rate and probability of detection across SNRs for the plain MRVM methods. Each simulation run has two targets with a Doppler velocity uniformly generated between -75 and 75. The amplitudes of the reflection of the targets follow a Swerling 2 fluctuation model. The simulation is repeated 1000 times.

Model	R-MRVM		V-MRVM	
SNR	r_{fa}	p_d	r_{fa}	p_d
15 dB	1.5457	0.6554	1.6463	0.7281
20 dB	0.8180	0.6120	0.7500	0.7285
25 dB	0.696	0.6295	0.664	0.73
30 dB	0.686	0.62	0.839	0.7385

Table 6.1 shows the results for the two plain MRVM methods. The table reveals the inherent limit in terms of probability of detection for the ambiguity aware methods. Another notable result is that in the VALSE-based plain MRVM, we clearly see a jump in false alarm rate when going from 25 dB to 30 dB. Upon closer inspection of the results, we see that for higher SNRs, the VALSE-based plain MRVM tends to return multiple detections in a single place when only one target is present. We also see that a lot of these double detections get rejected when using an ambiguity aware relevance vector machine.

6.4. Simulation of a difficult case

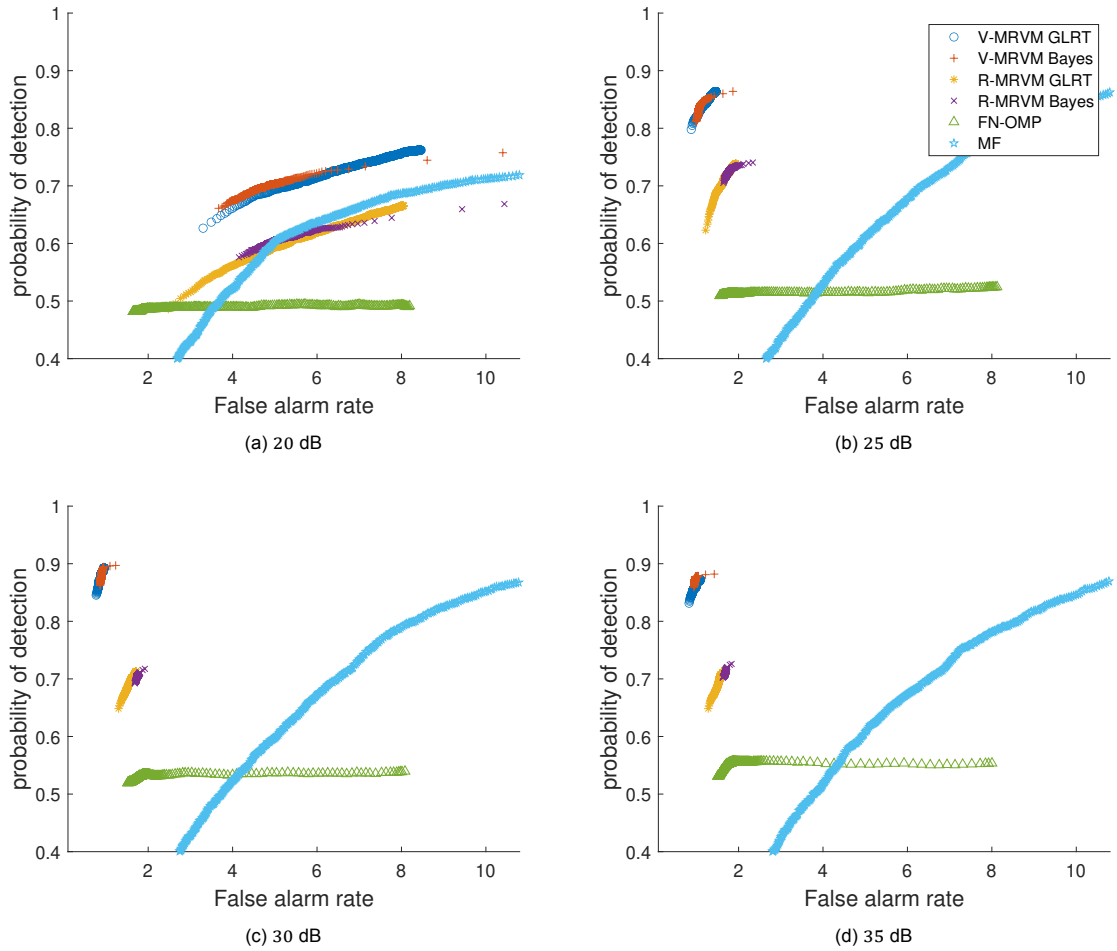
We perform a Monte Carlo simulation of a particularly difficult case to assess the performance of the methods in difficult situations and thus assess the robustness of the methods when the methods have to deal with difficult scenarios. This simulation study also highlights the drawback of greedy algorithms. The difficulty in the simulated scenario is caused by simulating two targets that both have a specific ambiguity that coincides. This coincidence causes a sharp peak in the matched filter response, making that specific Doppler velocity a strong local optimum.

To generate the observed data, we simulate data similar to the data of the simulation of a general case. We again simulate the Doppler responses of two targets for two individual bursts of length 12 from the data generating process (6.1). We draw the two amplitudes from a Swerling 2 fluctuation model. I.e., $\alpha_m \sim Rayleigh(\iota_m)$, where we now choose $\iota_m = 1.075\ \forall m \in \{1, 2\}$. We choose a higher ι_m in this study to increase the SNR by 5 dB compared to the general case in order to make the difficult problem more manageable for the methods used.

The two targets are now generated at fixed points. For each simulation run, we choose $\theta_1 = 40$ m/s and $\theta_2 = -27.5$ m/s. These Doppler velocities together with the ambiguity folds of 30 m/s and 37.5 m/s will cause two ambiguities to overlap at a Doppler velocity of 10 m/s.

For the noise term, we generate i.i.d. complex Gaussian noise with a standard deviation, σ, of 1.04, 0.585, 0.329, or 0.185 to get a signal-to-noise ratio of 20, 25, 30, or 35 dB, respectively. The SNR is calculated according to (6.5).

All thresholds are varied in the same way as in the simulation study for the general case. We then run the experiment 1000 times, each time with independently generated data and determine the average probability of detection p_d and false alarm rate r_{fa}. This procedure again results in 1000 pairs of p_d and r_{fa}, forming approximate ROC curves.



(a) 20 dB (b) 25 dB

(c) 30 dB (d) 35 dB

Figure 6.2: Plots of the estimated ROC curves for all methods that have a threshold. Each simulation run has two targets with a Doppler velocity of 40 and -27.5, respectively. The amplitudes of the reflection of the targets follow a Swerling 2 fluctuation model. The legend is shared across all subfigures. The scatterplots corresponds to the following SNR: (a) 20 dB, (b) 25 dB, (c) 30 dB, (d) 35 dB.

Figure 6.2 shows the results of the experiment. For an SNR of 20, 25, 30 and 35 dB, we get six approximations of ROC curves. I.e., for the root-based MRVM methods with GLRT as well as Bayesian ambiguity awareness, the VALSE-based MRVM methods with GLRT and Bayesian ambiguity awareness, the FN-OMP method, and the matched filter method.

Where in the first simulation, the FN-OMP method came close to the framework, it now clearly shows a shortcoming in performance. Every method that is extended to have ambiguity awareness tends to outperform the FN-OMP method across all SNRs. This lack of performance of FN-OMP is caused by the greedy nature of the FN-OMP algorithm. The greedy nature of the FN-OMP method causes the method to quickly select the spurious peak in the matched filter response and does not allow for updating that component later on. The MRVM-based methods also have a tendency to select such a strong peak, but are more flexible and allow for updates of that component later on.

Comparing the matched filter method to the other methods shows that the matched filter method is able to achieve a good probability of detection, but suffers from high false alarm rates.

6.5. Incorporation of prior information

We now set up a simulation study to show the impact on detection performance of the methods that have Bayesian ambiguity awareness when we have a priori information on the Doppler velocity of the targets.

Apart from the generation of the Doppler velocities, the setup is exactly the same as in the simulation study for general cases. We now generate the Doppler velocities according to a Gaussian mixture distribution instead of a uniform distribution. The Gaussian mixture consists of two Gaussians, both having a probability of 0.5 to be drawn from. The first Gaussian has a mean of -8 and a standard deviation of 10, the second Gaussian has a mean of 40 and a standard deviation of 5. The pdf of the resulting Gaussian mixture distribution is shown in Figure 6.3.

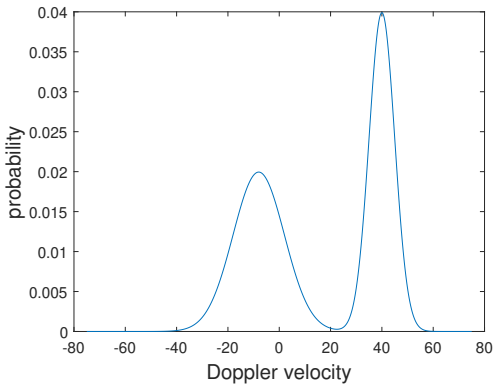


Figure 6.3: Distribution of the Doppler velocities used in the simulation study with prior information.

All thresholds are varied in the same way as in the simulation study for the general case. We then run the experiment 1000 times, each time with independently generated data and determine the average probability of detection p_d and false alarm rate r_{fa}. This procedure again results in 1000 pairs of p_d and r_{fa}, forming approximate ROC curves.

Figure 6.4 shows the results of the experiment. For an SNR of 15, 20, 25 and 30 dB, we get eight approximations of ROC curves. I.e., for the root-based MRVM methods with GLRT as well as Bayesian ambiguity awareness, the VALSE-based MRVM methods with GLRT and Bayesian ambiguity awareness, the FN-OMP method, and the matched filter method. For completeness, the figure also depicts the results for the methods that are extended with Bayesian ambiguity awareness when using a flat prior.

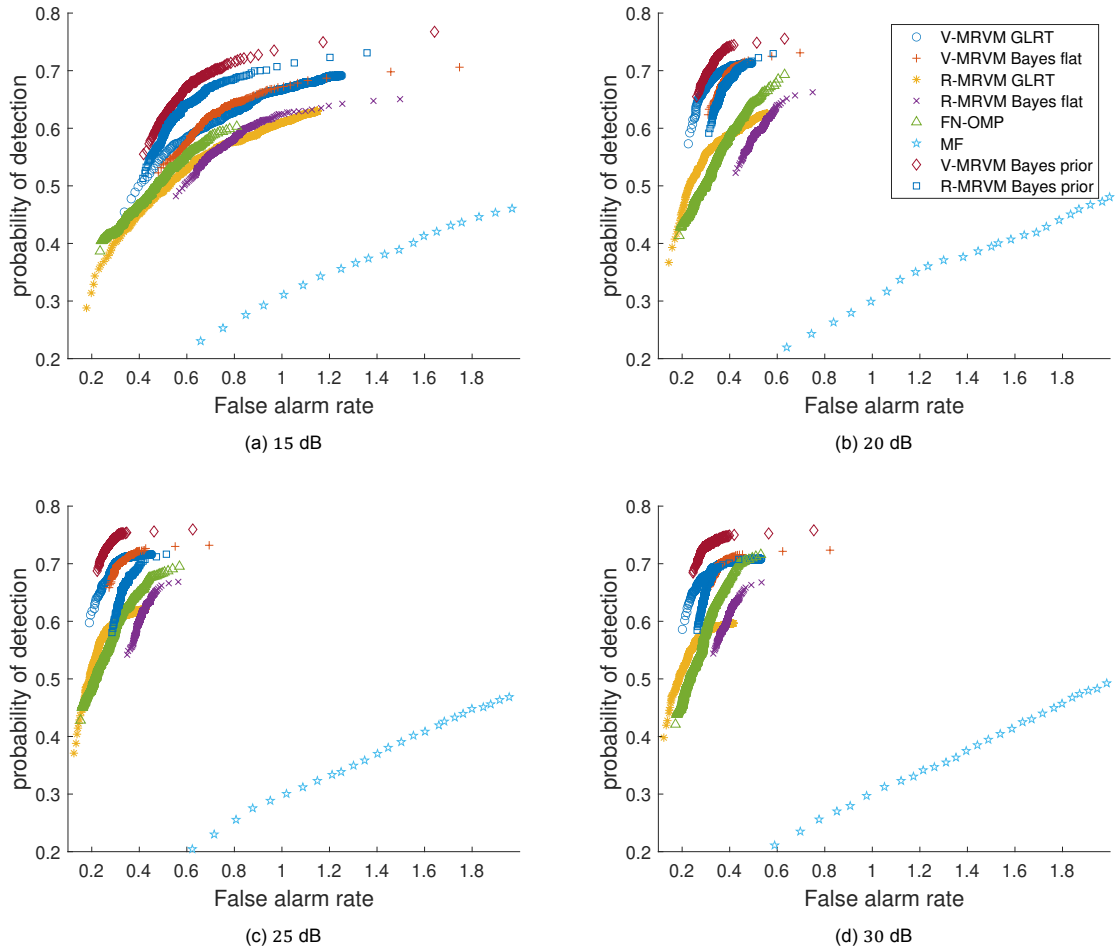
The results are similar to the simulation of a general case. As could be expected due to the fact that the simulation set-up is very similar. The only difference is that the target velocities are now generated according to a Gaussian mixture instead of a uniform distribution.

There is a clear advantage when using prior information. The methods with Bayesian ambiguity awareness perform significantly better when using the correct prior, compared to when these methods use a flat prior. Across all SNRs, the methods that use the correct prior achieve a lower false alarm rate as well as a higher probability of detection.

To compare the plain MRVM methods with the ambiguity aware methods, we repeat the simulation experiment for the plain MRVM methods. As there is no threshold like in the ambiguity awareness to be set in the plain MRVM methods, we only generate one point and not a complete curve.

Table 6.2: Average false alarm rate and probability of detection across SNRs for plain MRVM-based methods. Each simulation run has two targets with a Doppler velocity generated according to the Gaussian mixture as depicted in Figure 6.3. The amplitudes of the reflection of the targets follow a Swerling 2 fluctuation model. The simulation is repeated 1000 times.

Model	R-MRVM		V-MRVM	
SNR	r_{fa}	p_d	r_{fa}	p_d
15 dB	1.4510	0.6905	1.5920	0.7285
20 dB	0.7930	0.6320	0.6970	0.7290
25 dB	0.6390	0.6340	0.7200	0.7255
30 dB	0.6860	0.6230	0.8480	0.7340



(a) 15 dB (b) 20 dB

(c) 25 dB (d) 30 dB

Figure 6.4: Plots of the estimated ROC curves for all methods that have a threshold. Each simulation run has two targets with a Doppler velocity generated according to the Gaussian mixture as depicted in Figure 6.3. The amplitudes of the reflection of the targets follow a Swerling 2 fluctuation model. The legend is shared across all subfigures. The scatterplots corresponds to the following SNR: (a) 15 dB, (b) 20 dB, (c) 25 dB, (d) 30 dB.

Table 6.2 shows the results for the two plain MRVM methods. A notable result is that the Bayesian ambiguity awareness with correct prior is actually able to increase the probability of detection compared with the plain MRVM-based methods. This is an especially interesting result as the simulation of general cases revealed that, when no prior information is available, the plain MRVM-based methods indicated a limit in terms of probability of detection for the ambiguity aware methods. This increase can be attributed to the fact that we choose the MAP estimate when using the Bayesian ambiguity awareness. When the MRVM returns an incorrect relevance vector that is actually an ambiguity of the actual target, the Bayesian ambiguity awareness is sometimes able to see that the posterior probability of the target selected by the MRVM is lower than the posterior probability of another position. The Bayesian ambiguity awareness will then select the Doppler velocity with the highest posterior probability.

6.6. Comparison with MCMC

To evaluate the estimated posterior of the framework with Bayesian ambiguity awareness, we design an MCMC filter. We specifically use a Metropolis-Hastings algorithm to search the sample space and estimate the true posterior. This comparison is of a qualitative nature as we cannot easily give measures to the differences of outputs and will therefore mostly rely on visual inspection.

We simulate a simple case for the comparison. We simulate the Doppler response of a single target with a Doppler velocity of 45 m/s three times with varying amplitudes. Each time, we simulate two individual bursts of length 12. We simulate the data according to the data generating process (6.1).

We consider three cases. In all cases we get a strong reflection in the first burst. In the first case, we get no reflection in the second burst. In the second case, we get a weak reflection in the second burst. In the third case, we also get a strong reflection in the second burst. We thus set the amplitude of the target from the first burst fixed at $\alpha_1 = 1$. We use a varied second amplitude to examine the effect on the posterior distribution. So, for the second burst, we vary the amplitude $\alpha_1 \in \{0.0, 0.3, 0.7\}$. We set the carrier frequency $f_c = 3 \cdot 10^8$ Hz. The PRT of the first burst is $0.5 \cdot 10^{-3}$ s, resulting in a Doppler fold of 30 m/s according to (2.3). The PRT of the second burst is $0.4 \cdot 10^{-3}$ s, resulting in a Doppler fold of 37.5 m/s according to (2.3). For the noise term, we generate i.i.d. complex Gaussian noise with a standard deviation of 1.

We give the simulated data to the VALSE based MRVM with Bayesian ambiguity awareness as well as a Metropolis–Hastings algorithm.

We design the Metropolis–Hastings algorithm by using the likelihood that results from the data generating process (6.1). We search the parameter space for four variables: the Doppler velocity of the target, the amplitude of the target in the first burst, the amplitude of the target in the second burst and the standard deviation of the noise. We use uniform priors for all these variables. We fix the model order at one, as we are mainly interested in the posterior distribution of the Doppler velocity.

Note that the design of a Metropolis–Hastings algorithm is far from trivial for a sample space that forms from a problem with ambiguities. As there exists a large amount of strong local optima, a Metropolis–Hastings algorithm has the tendency to get stuck in one of the local optima and not search the complete sample space. This is especially the case when generating candidate states for the velocity according to a distribution with low kurtosis, such as a normal distribution. Therefore, we opt for a Student's-t distribution with two degrees of freedom for the candidate states of the velocity. A heavy-tailed distribution, such as the Student's-t distribution with two degrees of freedom, has a chance to generate a candidate state according to the tails of the distribution. Such a candidate state has the possibility to "escape" a local optimum in which candidate states generated according to a normal distribution are likely to get stuck.

For the Metropolis-Hastings algorithm, we initialise 128 independent chains. Each chain produces $100,000$ proposals, bringing the total amount of samples to $12,800,000$. We initialise the weights at 0. The velocity is initialised according to a uniform distribution on the interval from -75 to 75. The variance is initialised at the sample variance of the generated data. We generate the proposals of the weights by generating zero-mean complex Gaussian updates with a standard deviation of 0.05. The proposals for the variance are also generating zero-mean complex Gaussian updates with a standard deviation of 0.05. The update proposals for the Doppler velocity are, as described above, generated by a complex Student's-t distribution with two degrees of freedom. We multiply the proposed innovation by 3.5 to allow the proposals to jump between local optima more easily. To reduce finite sample bias, we define a burn-in period of the first 10000 samples of each MCMC chain.

For the VALSE-based MRVM with Bayesian ambiguity awareness, we use the same setup as in the earlier simulation studies. I.e., we define a grid from -75 to 75, with a resolution of one. We prune the component from the model when the specific α becomes greater than 10^6. We stop iterating when the summed absolute difference between the alphas of the current iteration and the previous iteration is smaller than 10^{-10}, or when we exceed 2000 iterations. In the ambiguity awareness, we decide that the relevance vector under test is not a target if the posterior probability of no target exceeds 0.05. If the probability of no target is below 0.05, we choose the (MAP) estimate.

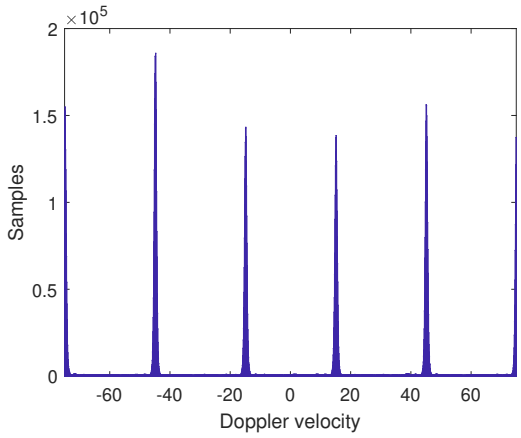
Figure 6.5 shows the collection of samples for the posterior of the Doppler velocity of the Metropolis-Hastings algorithm together with the output of the VALSE-based MRVM with Bayesian ambiguity awareness for the three cases.

Comparing the results of the VALSE-based MRVM with Bayesian ambiguity awareness to the Metropolis-Hastings algorithm for the case of a non-existent second reflection, we see that both outputs are similar. The peaks are in both outputs almost uniformly distributed over the possible ambiguities, as we would expect in the case where we only have a reflection in one burst.

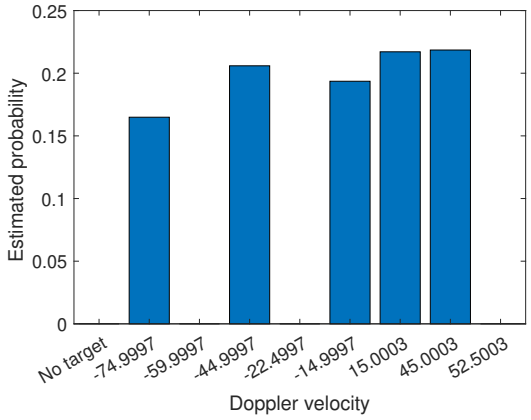
In the case of a weak reflection in the second burst, we would expect one high peak at 45 m/s and small, but significant, peaks at the places of the possible ambiguities. We see this pattern in both outputs.

We also see that the difference between the peak at 45 m/s and the possible ambiguities is greater in the output of the Metropolis-Hastings algorithm compared to the output of the VALSE-based MRVM with Bayesian ambiguity awareness. This observation leads to the conclusion that the VALSE-based MRVM with Bayesian ambiguity awareness is slightly less capable of accessing all useful information received in the second burst.

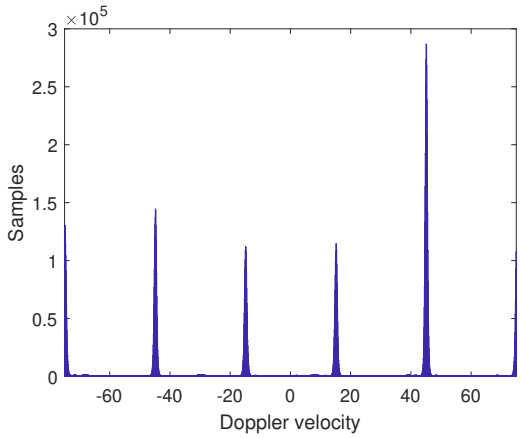
Comparing the VALSE-based MRVM with Bayesian ambiguity awareness with the Metropolis-Hastings algorithm for the case of a strong second reflection, we again see similar results. We do however see that the Metropolis-Hastings algorithm shows small peaks at the possible ambiguities. The VALSE-based MRVM with Bayesian ambiguity awareness estimates the probability that there is a target at any of the ambiguities to be zero, while the Metropolis-Hastings algorithm indicates that there still is a low probability.



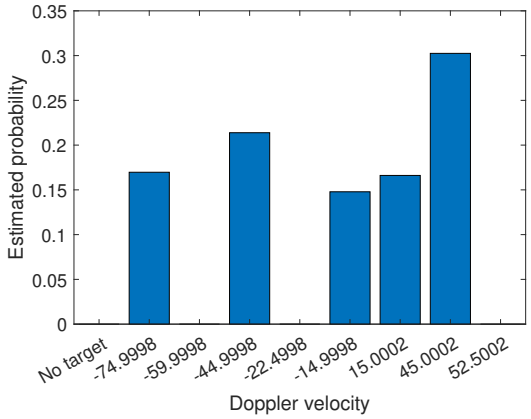
(a) MCMC, strong reflection in the first burst, no reflection in the second burst



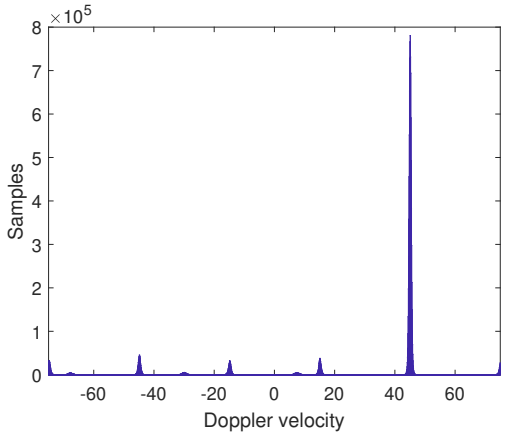
(b) V-MRVM Bayes, strong reflection in the first burst, no reflection in the second burst



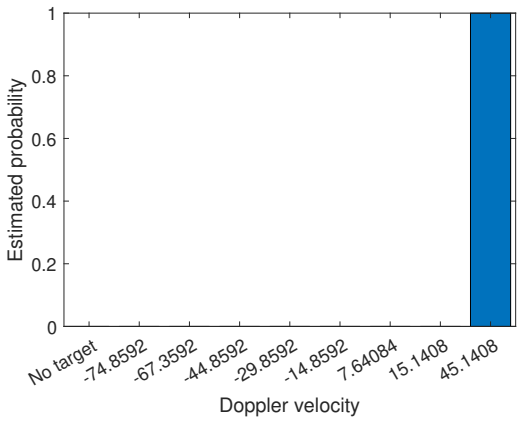
(c) MCMC, strong reflection in the first burst, weak reflection in the second burst



(d) V-MRVM Bayes, strong reflection in the first burst, weak reflection in the second burst



(e) MCMC, strong reflection in the first burst, strong reflection in the second burst



(f) V-MRVM Bayes, strong reflection in the first burst, strong reflection in the second burst

Figure 6.5: Estimated posterior distributions for the Metropolis-Hastings algorithm in the left column and the VALSE-based MRVM with Bayesian ambiguity awareness in the right column. The simulations are for a simple case with two bursts of a scenario of only one target with a Doppler velocity of 45 m/s. The reflection in the first burst is strong in every simulation. The reflection of the target in the second burst is non-existent in (a) and (b), weak in (c) and (d), and strong in (e) and (f).

7

Conclusion and recommendations

7.1. Conclusion

This thesis has worked towards a framework that can be used to resolve ambiguities in a multi-target environment. The framework works on a video integration level, is able to take prior information into account, can quantify the uncertainty of the estimates, and has a clear convergence criterion. Thus meeting the goal set in the research scope described in section 1.3. The conclusion is given through the answering of the research questions.

How can we formulate a framework that improves the state-of-the-art when it comes to resolving ambiguities in a multi-target environment when processing on a video level?
This thesis proposes a novel framework to improve the state-of-the-art when it comes to resolving ambiguities in a multi-target environment when processing on a video level. The framework is essentially formulated in two steps. The first step uses a variant of the multitask relevance vector machine adjusted to work off-grid on the complete domain. In the second step, the framework takes possible ambiguities into account. In the ambiguity awareness step, the framework has two options. The framework either performs a frequentist test to test whether a relevance vector can statistically significantly be discerned from ambiguities, or the framework estimates a posterior distribution of the individual relevance vectors.

What model or statistical technique should drive the framework?
The framework is driven by a relevance vector machine adjusted to work off-grid. The relevance vector machine is highly suitable to drive the framework. Specifically due to the fact that the relevance vector machine, when adjusted for complex numbered problems, assumes a complex Gaussian distribution as a prior on the weights. This complex Gaussian coincides with cases one and two of the Swerling fluctuation models. The relevance vector machine is also extendable to multiple measurement vectors under both the assumptions for coherent processing, as well as the assumptions needed for non-coherent video-level processing. The relevance vector machine has some desirable statistical properties. I.e., the global optimum is the maximally sparse solution and the relevance vector machine suffers from fewer local optima compared with competing models. Additionally, the relevance vector machine has a clear converge criterion.

How can we incorporate a priori information?
By taking a Bayesian view in the extension towards an ambiguity aware relevance vector machine, we are able to take prior information into account. In a Bayesian treatment of the ambiguity awareness, we take the total data likelihood of the relevance vector machine model specification as conditional probability. The framework then estimates a posterior distribution for each relevance vector by using Bayes' theorem, taking prior distributions into account.

How can we quantify the uncertainty in the estimates of the framework?
The framework quantifies the uncertainty in the estimates in two possible ways. Both of the techniques

are executed in the extension towards ambiguity awareness. The first procedure is a frequentist test. This frequentist test is used to test whether we can statistically significantly discern the estimate of the relevance vector machine from its possible ambiguities. This test provides a test statistic with the corresponding distribution. The combination of test statistics and corresponding distribution allows for p-values to be calculated. p-values represent the probability of obtaining test results at least as extreme as the observed result, under the assumption that the null hypothesis is correct. p-values, therefore, provide a measure of confidence. The second way is via the Bayesian ambiguiy awareness. the Bayesian ambiguity awareness provides a posterior distribution of the returned relevance vectors. This posterior distribution makes it possible to perform a qualitative assessment of the confidence of the estimate. There is a low amount of confidence in the estimate if the posterior distribution has a lot of probability mass on the possible ambiguities or on the 'no target' outcome. There is high confidence in an estimate when all probability mass is located at the returned estimate.

7.2. Future work

There are three main restrictions to the approach in this thesis as defined in the research scope. The first three candidates for future work directly result from lifting the restrictions defined in the research scope. Besides lifting the restrictions, one additional suggestion for future research is made below.

The first restriction concerns the noise term. The framework assumes white Gaussian noise. The white Gaussian noise assumption is realistic in some situations, but is frequently violated in real scenarios. Especially when working with radars, clutter can easily cause the noise to be coloured, non-Gaussian, or both. The first suggestion for future research is therefore to drop this assumption and write out the framework for other types of noise terms. E.g., allowing for heterogeneity or serial correlation in the covariance matrix. Or even opting for a completely different distribution, such as a Student's t-distribution. When adjusting the noise terms, the derivations will become more challenging. Some of the steps in the current derivations are only allowed when working with complex Gaussian distributions. When these steps are not applicable, the derivations will become more challenging.

The second restriction is that the framework is only tested on simulated data. Future work is thus to take real data to the framework. When using real data there are a lot of caveats, such as coloured noise and clutter types that are not accounted for. It would therefore be interesting to see how the framework would fare when handling real data.

The third restriction is the restriction of applying the framework to Doppler processing. Future work is to apply the framework to other types of ambiguities as well. Candidate problems with ambiguities are estimating range or DOA in radar systems. The framework should be adjusted when applying the framework to other types of ambiguities. When applying the framework to DOA estimation, the modifications are small. In DOA estimation, the same structure can be exploited as in Doppler processing. The main difference will be the choice of design matrices. When applying the framework to range estimation, the assumptions made to take the relevance vector machine off-grid no longer hold. In that case we should turn towards other off-grid methods. Required methods will then lie more in the class of optimisation methods such as the Taylor expansion method of Yang *et al.* [41].

The final suggestion for future work is to extend the framework beyond the first two Swerling fluctuation models. The framework currently has strong parallels specifically to Swerling cases one and two. An abstract goal is to work towards one model that encompasses all four Swerling cases. Ideally, the model then even encompasses every case in between the specific Swerling cases as well. To extend the framework beyond these Swerling cases, the prior on the weights of the relevance vector machine needs to be adjusted. The adjusted distribution can no longer be a complex Gaussian distribution. This distribution no longer being Gaussian will cause the derivations to be more difficult due to the same reason as with lifting the restriction of the noise term being complex Gaussian. A couple of steps in the derivations can no longer be performed, giving rise to the need of an alternative derivation.

Bibliography

[1] M. A. Richards, J. A. Scheer, J. Scheer, and W. A. Holm, *Principles of Modern Radar: Basic Principles, Volume 1*, ser. Electromagnetics and Radar. Institution of Engineering and Technology, 2010.

[2] M. A. Richards, W. L. Melvin, J. A. Scheer, and J. Scheer, *Principles of Modern Radar: Advanced Techniques, Volume 2*, ser. EBSCO ebook academic collection. Institution of Engineering and Technology, 2012.

[3] M. I. Skolnik, *Radar Handbook, Third Edition*, ser. Electronics electrical engineering. McGraw-Hill Education, 2008.

[4] A. Manikas and C. Proukakis, "Modeling and estimation of ambiguities in linear arrays," *IEEE Transactions on Signal Processing*, vol. 46, no. 8, pp. 2166–2179, 1998.

[5] S. A. Hovanessian, "An algorithm for calculation of range in a multiple prf radar," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-12, no. 2, pp. 287–290, 1976.

[6] M. I. Skolnik, *Radar handbook*. New York, NY: McGraw-Hill, 1970.

[7] N. Reddy and M. Swamy, "Resolution of range and doppler ambiguities in medium prf radars in multiple-target environment," in *ICASSP '84. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 9, 1984, pp. 514–517.

[8] G. V. Trunk and M. W. Kim, "Ambiguity resolution of multiple targets using pulse-doppler waveforms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 30, no. 4, pp. 1130–1137, 1994.

[9] F. Cai, H. Fan, Z. Lu, and Q. Fu, "Bernoulli filter for range and doppler ambiguous radar," *IET Signal Processing*, vol. 9, no. 9, pp. 647–654, 2015.

[10] M. Bocquel, J. N. Driessen, and A. Bagchi, "Multitarget particle filter addressing ambiguous radar data in tbd," in *2012 IEEE Radar Conference*, 2012, pp. 0575–0580.

[11] S. Bidon, J.-Y. Tourneret, L. Savy, and F. Le Chevalier, "Bayesian sparse estimation of migrating targets for wideband radar," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 2, pp. 871–886, 2014.

[12] S. Bidon, O. Besson, J.-Y. Tourneret, and F. Le Chevalier, "Bayesian sparse estimation of migrating targets in autoregressive noise for wideband radar," in *2014 IEEE Radar Conference*, 2014, pp. 0579–0584.

[13] M. Lasserre, S. Bidon, and F. Le Chevalier, "Velocity ambiguity mitigation of off-grid range migrating targets via bayesian sparse recovery," in *2016 IEEE Statistical Signal Processing Workshop (SSP)*, 2016, pp. 1–5.

[14] ——, "An unambiguous radar mode with a single prf wideband waveform," in *2017 IEEE Radar Conference (RadarConf)*, 2017, pp. 0404–0409.

[15] S. Bidon, M. Lasserre, and F. Le Chevalier, "Unambiguous sparse recovery of migrating targets with a robustified bayesian model," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 55, pp. 108–123, 2019.

[16] S. Bidon, A. Tamalet, and J.-Y. Tourneret, "Variational bayesian inference for sparse representation of migrating targets in wideband radar," in *2013 IEEE Radar Conference (RadarCon13)*, 2013, pp. 1–5.

[17] R. A. S. S. Branco and S. Bidon, “A variational bayes sparse recovery of migrating targets in ar noise,” in *2018 IEEE Radar Conference (RadarConf18)*, 2018, pp. 0240–0245.

[18] F. Shaban and M. A. Richards, “Application of l1 reconstruction of sparse signals to ambiguity resolution in radar,” in *2013 IEEE Radar Conference (RadarCon13)*, 2013, pp. 1–6.

[19] S. S. Chen, D. L. Donoho, and M. A. Saunders, “Atomic Decomposition by Basis Pursuit,” *SIAM Review*, vol. 43, no. 1, pp. 129–159, Jan. 2001.

[20] R. Tibshirani, “Regression shrinkage and selection via the lasso,” *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[21] L. Aouchiche, G. Desodt, C. Adnet, and L. Ferro-Famil, “Enhanced omp algorithm for the detection and estimation of closely spaced moving objects in the presence of doppler ambiguities,” in *2015 IEEE Radar Conference*, 2015, pp. 260–265.

[22] P. van Genderen and W. Meijer, “Non coherent integration in a medium prf radar,” in *Proceedings International Radar Conference*, 1995, pp. 91–94.

[23] J. Dai, X. Bao, W. Xu, and C. Chang, “Root sparse bayesian learning for off-grid doa estimation,” *IEEE Signal Processing Letters*, vol. 24, no. 1, pp. 46–50, 2017.

[24] M.-A. Badiu, T. L. Hansen, and B. H. Fleury, “Variational bayesian inference of line spectra,” *IEEE Transactions on Signal Processing*, vol. 65, no. 9, pp. 2247–2261, 2017.

[25] S. F. Cotter, B. D. Rao, K. Engan, and K. Kreutz-Delgado, “Sparse solutions to linear inverse problems with multiple measurement vectors,” *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2477–2488, 2005.

[26] Z. Zhang and B. D. Rao, “Sparse signal recovery with temporally correlated source vectors using sparse bayesian learning,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 5, pp. 912–926, 2011.

[27] J. Zhu, Q. Zhang, P. Gerstoft, M.-A. Badiu, and Z. Xu, “Grid-less variational bayesian line spectral estimation with multiple measurement vectors,” *Signal Processing*, vol. 161, pp. 155–164, 2019.

[28] J. W. Crispin and A. L. Maffett, “Radar cross-section estimation for simple shapes,” *Proceedings of the IEEE*, vol. 53, no. 8, pp. 833–848, 1965.

[29] ——, “Radar cross-section estimation for complex shapes,” *Proceedings of the IEEE*, vol. 53, no. 8, pp. 972–982, 1965.

[30] M. E. Tipping, “The relevance vector machine,” in *Advances in Neural Information Processing Systems*, S. Solla, T. Leen, and K. Müller, Eds., vol. 12. MIT Press, 1999.

[31] D. J. C. MacKay, “Bayesian non-linear modelling for the prediction competition,” *ASHRAE Transactions*, vol. 4, pp. 1053–1062, 1994.

[32] R. M. Neal, *Bayesian Learning for Neural Networks*. Berlin: Springer, 1996.

[33] M. E. Tipping, “Sparse bayesian learning and relevance vector machine,” *J. Mach. Learn. Res.*, vol. 1, pp. 211–244, 01 2001.

[34] D. P. Wipf and B. D. Rao, “Sparse bayesian learning for basis selection,” *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2153–2164, 2004.

[35] E. J. Candes, J. Romberg, and T. Tao, “Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information,” *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[36] I. F. Gorodnitsky and B. D. Rao, “Sparse signal reconstruction from limited data using focuss: a re-weighted minimum norm algorithm,” *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 600–616, 1997.

[37] B. D. Rao and K. Kreutz-Delgado, “An affine scaling methodology for best basis selection,” *IEEE Transactions on Signal Processing*, vol. 47, no. 1, pp. 187–200, 1999.

[38] E. J. Candès, M. B. Wakin, and S. P. Boyd, “Enhancing sparsity by reweighted ℓ1 minimization,” *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 877–905, Dec 2008.

[39] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. USA: Prentice-Hall, Inc., 1993.

[40] S. Ji, D. Dunson, and L. Carin, “Multitask compressive sensing,” *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 92–106, 2009.

[41] Z. Yang, L. Xie, and C. Zhang, “Off-grid direction of arrival estimation using sparse bayesian inference,” *IEEE Transactions on Signal Processing*, vol. 61, no. 1, pp. 38–43, 2013.

[42] H. Zhu, G. Leus, and G. B. Giannakis, “Sparsity-cognizant total least-squares for perturbed compressive sampling,” *IEEE Transactions on Signal Processing*, vol. 59, no. 5, pp. 2002–2016, 2011.

[43] H. Sagan, *Introduction to the Calculus of Variations*, ser. Dover books on advanced mathematics. Dover Publications, 1992.

[44] M. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul, “An introduction to variational methods for graphical models,” *Machine Learning*, vol. 37, pp. 183–233, 01 1999.

[45] T. M. Cover and J. A. Thomas, *Elements of Information Theory 2nd Edition (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, July 2006.

[46] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006.

[47] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[48] P. Jupp and K. Mardia, *Directional Statistics*, ser. Wiley Series in Probability and Statistics. Wiley, 2009.

[49] J. Kormylo and J. Mendel, “Maximum likelihood detection and estimation of bernoulli - gaussian processes,” *IEEE Transactions on Information Theory*, vol. 28, no. 3, pp. 482–488, 1982.

[50] J. Neyman, E. S. Pearson, and K. Pearson, “Ix. on the problem of the most efficient tests of statistical hypotheses,” *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, vol. 231, no. 694-706, pp. 289–337, 1933.

[51] H. Jeffreys, *Theory of Probability*, ser. International series of monographs on physics. Clarendon Press, 1998.

[52] S. M. Kay, *Fundamentals of Statistical Signal Processing: Detection theory*. USA: Prentice-Hall, Inc., 1998.

[53] D. A. Harville, *Matrix algebra from a statistician’s perspective*. New York ;: Springer, 1997.

[54] L. Guttman, “Enlargement Methods for Computing the Inverse Matrix,” *The Annals of Mathematical Statistics*, vol. 17, no. 3, pp. 336 – 343, 1946.

[55] J. R. Magnus and H. Neudecker, *Matrix Differential Calculus with Applications in Statistics and Econometrics*, ser. Wiley Series in Probability and Statistics. Wiley, 2019.

[56] P. Swerling, “Probability of detection for fluctuating targets,” *IRE Transactions on Information Theory*, vol. 6, no. 2, pp. 269–308, 1960.

A

Derivation of the complex-valued relevance vector machine

Repeating the model specification of the complex-valued relevance vector machine, we have the following canonical form

$$\mathbf{t} = \mathbf{\Phi}\mathbf{w} + \boldsymbol{\varepsilon}, \tag{A.1}$$

where $\mathbf{t} \in \mathbb{C}^N$ $\mathbf{\Phi} \in \mathbb{C}^{N\times M}$ and $\mathbf{w} \in \mathbb{C}^M$. The noise term $\boldsymbol{\varepsilon}$ is also modelled as an independent zero-mean complex Gaussian random variable. I.e., $\boldsymbol{\varepsilon} \sim \mathcal{CN}(\mathbf{0}, \sigma^2\mathbf{I})$. Due to the complex Gaussian noise term, we have

$$p(t_n|y(\mathbf{x}_n;\mathbf{w}),\sigma^2) = \mathcal{CN}(t_n|y(\mathbf{x}_n;\mathbf{w}),\sigma^2) \tag{A.2}$$

Allowing us to write the complete data likelihood as

$$p\left(\mathbf{t} \mid \mathbf{w},\sigma^2\right) = \mathcal{CN}\left(\mathbf{t}|\mathbf{\Phi}\mathbf{w},\sigma^2\mathbf{I}\right) = \frac{1}{\pi^N\left|\sigma^2\mathbf{I}\right|}\exp\left\{-\frac{1}{\sigma^2}\|\mathbf{t}-\mathbf{\Phi}\mathbf{w}\|_2^2\right\}. \tag{A.3}$$

We define a zero-mean complex Gaussian prior on the weights $\mathbf{w}$. The prior is given as

$$p\left(\mathbf{w}|\boldsymbol{\alpha}\right) = \prod_{i=0}^{M-1}\mathcal{CN}(w_i|0,\alpha_i^{-1}), \tag{A.4}$$

where $\boldsymbol{\alpha}$ is an M-length vector with hyperparameters. By defining the matrix $\mathbf{A} = \mathrm{diag}(\alpha_0,\ldots,a_{M-1})$, we can rewrite the prior distribution on the weights as

$$p\left(\mathbf{w}|\boldsymbol{\alpha}\right) = \mathcal{CN}\left(\mathbf{w}|\mathbf{0},\mathbf{A}^{-1}\right) = \frac{1}{\pi^M\left|\mathbf{A}^{-1}\right|}\exp\left\{-\mathbf{w}^H\mathbf{A}\mathbf{w}\right\}. \tag{A.5}$$

We first write out the joint distribution of $\mathbf{t}$ and $\mathbf{w}$. The logarithm of the joint distribution is given by

$$\ln p(\mathbf{t},\mathbf{w}|\boldsymbol{\alpha},\sigma^2) = \ln p(\mathbf{w}|\boldsymbol{\alpha}) + \ln p(\mathbf{t}|\mathbf{w};\sigma^2). \tag{A.6}$$

We write out the logarithm of the joint distribution as

$$\ln p(\mathbf{t},\mathbf{w}|\boldsymbol{\alpha},\sigma^2) = -\mathbf{w}^H\mathbf{A}\mathbf{w} - \left(\mathbf{t}-\mathbf{\Phi}\mathbf{w}\right)^H\mathbf{B}^{-1}\left(\mathbf{t}-\mathbf{\Phi}\mathbf{w}\right) + \mathrm{Const}, \tag{A.7}$$

where $\mathbf{B}$ is defined as $\sigma^2\mathbf{I}$ and Const denotes the terms that are independent from both $\mathbf{t}$ and $\mathbf{w}$. We rewrite the expression of $\ln p(\mathbf{t},\mathbf{w}|\boldsymbol{\alpha},\sigma^2)$ using a commonly used technique called completing the square as described in, e.g., [46, Ch. 2]. When completing the square, we look at the terms in the exponent of a (complex) Gaussian distribution and determine the mean and covariance characterising that distribution. Considering a general complex Gaussian $p(\mathbf{x}|\boldsymbol{\mu},\mathbf{\Sigma})$, we write the exponent as

$$-\left(\mathbf{x}-\boldsymbol{\mu}\right)^H\mathbf{\Sigma}^{-1}\left(\mathbf{x}-\boldsymbol{\mu}\right) = -\mathbf{x}^H\mathbf{\Sigma}^{-1}\mathbf{x} + 2\mathbf{x}^H\mathbf{\Sigma}^{-1}\boldsymbol{\mu} + \mathrm{Const}, \tag{A.8}$$

where Const is again the term independent of $\mathbf{x}$. By writing any exponent of a (complex) Gaussian in this form, we can immediately equate the second order terms in $\mathbf{x}$ to $\boldsymbol{\Sigma}^{-1}$ and the linear terms in $\mathbf{x}$ to $\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}$, allowing us to find $\boldsymbol{\mu}$.

Turning back to (A.7), we collect the quadratic terms, resulting in

$$\begin{aligned} &-\mathbf{w}^H\left(\mathbf{A}+\boldsymbol{\Phi}^H\mathbf{B}^{-1}\boldsymbol{\Phi}\right)\mathbf{w}-\mathbf{t}^H\mathbf{B}^{-1}\mathbf{t}+\mathbf{t}^H\mathbf{B}^{-1}\boldsymbol{\Phi}\mathbf{w}+\mathbf{w}^H\boldsymbol{\Phi}^H\mathbf{B}^{-1}\mathbf{t} \\ &=\begin{bmatrix}\mathbf{w}\\ \mathbf{t}\end{bmatrix}^H \underbrace{\begin{bmatrix}\mathbf{A}+\boldsymbol{\Phi}^H\mathbf{B}^{-1}\boldsymbol{\Phi} & -\boldsymbol{\Phi}^H\mathbf{B}^{-1}\\ -\mathbf{B}^{-1}\boldsymbol{\Phi} & \mathbf{B}^{-1}\end{bmatrix}}_{\boldsymbol{\Sigma}^{-1}_{\mathbf{w},\mathbf{t}}}\begin{bmatrix}\mathbf{w}\\ \mathbf{t}\end{bmatrix}. \end{aligned} \tag{A.9}$$

So, the precision matrix of the joint distribution of $\mathbf{w}$ and $\mathbf{t}$ is given by

$$\boldsymbol{\Lambda}_{\mathbf{w},\mathbf{t}}=\boldsymbol{\Sigma}^{-1}_{\mathbf{w},\mathbf{t}}=\begin{bmatrix}\mathbf{A}+\boldsymbol{\Phi}^H\mathbf{B}^{-1}\boldsymbol{\Phi} & -\boldsymbol{\Phi}^H\mathbf{B}^{-1}\\ -\mathbf{B}^{-1}\boldsymbol{\Phi} & \mathbf{B}^{-1}\end{bmatrix}. \tag{A.10}$$

We will now write out the covariance of the joint distribution of $\mathbf{w}$ and $\mathbf{t}$ by using the general identity that the inverse of a partitioned matrix is given by

$$\begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{C} & \mathbf{D}\end{bmatrix}^{-1}=\begin{bmatrix}\mathbf{M} & -\mathbf{M}\mathbf{B}\mathbf{D}^{-1}\\ -\mathbf{D}^{-1}\mathbf{C}\mathbf{M} & \mathbf{D}^{-1}+\mathbf{D}^{-1}\mathbf{C}\mathbf{M}\mathbf{B}\mathbf{D}^{-1}\end{bmatrix}, \tag{A.11}$$

where

$$\mathbf{M}=(\mathbf{A}-\mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1}. \tag{A.12}$$

$\mathbf{M}^{-1}$ is the so called Schur complement. So, we rewrite (A.10) by first writing out the inverse of the Schur complement as

$$\mathbf{M}=\left(\mathbf{A}+\boldsymbol{\Phi}^H\mathbf{B}^{-1}\boldsymbol{\Phi}-\mathbf{B}^{-1}\boldsymbol{\Phi}\left(\mathbf{B}^{-1}\right)^{-1}\mathbf{B}^{-1}\boldsymbol{\Phi}\right)^{-1}=\mathbf{A}^{-1}. \tag{A.13}$$

So, the covariance matrix of the joint distribution of $\mathbf{w}$ and $\mathbf{t}$ is

$$\boldsymbol{\Sigma}_{\mathbf{w},\mathbf{t}}=\begin{bmatrix}\mathbf{A}^{-1} & \mathbf{A}^{-1}\boldsymbol{\Phi}\mathbf{B}^{-1}\mathbf{B}\\ \mathbf{B}\mathbf{B}^{-1}\boldsymbol{\Phi}\mathbf{A}^{-1} & \mathbf{B}+\boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^H\end{bmatrix}=\begin{bmatrix}\mathbf{A}^{-1} & \mathbf{A}^{-1}\boldsymbol{\Phi}\\ \boldsymbol{\Phi}\mathbf{A}^{-1} & \mathbf{B}+\boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^H\end{bmatrix}. \tag{A.14}$$

By noting that there are no linear terms in either $\mathbf{w}$ or $\mathbf{t}$ in (A.7), we conclude that the mean of the joint distribution of $\mathbf{w}$ and $\mathbf{t}$ is equal to zero, i.e.,

$$\boldsymbol{\mu}_{\mathbf{w},\mathbf{t}}=\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\end{bmatrix}. \tag{A.15}$$

Having both the mean and covariance matrix, we have fully characterised the joint distribution of $\mathbf{w}$ and $\mathbf{t}$ as

$$p\left(\mathbf{t},\mathbf{w}|\boldsymbol{\alpha},\sigma^2\right)=\mathcal{CN}\left(\begin{bmatrix}\mathbf{w}\\ \mathbf{t}\end{bmatrix}\middle|\middle|\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\end{bmatrix},\begin{bmatrix}\mathbf{A}^{-1} & \mathbf{A}^{-1}\boldsymbol{\Phi}\\ \boldsymbol{\Phi}\mathbf{A}^{-1} & \mathbf{B}+\boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^H\end{bmatrix}\right). \tag{A.16}$$

From the joint distribution, we express the marginal distribution of $\mathbf{t}$ as

$$p\left(\mathbf{t}|\boldsymbol{\alpha},\sigma^2\right)=\mathcal{CN}\left(\mathbf{t}|\mathbf{0},\mathbf{B}+\boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^H\right). \tag{A.17}$$

We now turn to the posterior on the weights, $p\left(\mathbf{w}|\mathbf{t},\boldsymbol{\alpha},\sigma^2\right)$. We first consider a general case, where we have a joint complex Gaussian distribution on $\mathbf{x}$ with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. We assume that we can make the following partition

$$\begin{aligned} \mathbf{x}&=\begin{bmatrix}\mathbf{x}_a\\ \mathbf{x}_b\end{bmatrix},\\ \boldsymbol{\mu}&=\begin{bmatrix}\boldsymbol{\mu}_a\\ \boldsymbol{\mu}_b\end{bmatrix},\\ \boldsymbol{\Sigma}&=\begin{bmatrix}\boldsymbol{\Sigma}_{aa} & \boldsymbol{\Sigma}_{ab}\\ \boldsymbol{\Sigma}_{ba} & \boldsymbol{\Sigma}_{bb}\end{bmatrix},\\ \boldsymbol{\Lambda}=\boldsymbol{\Sigma}^{-1}&=\begin{bmatrix}\boldsymbol{\Lambda}_{aa} & \boldsymbol{\Lambda}_{ab}\\ \boldsymbol{\Lambda}_{ba} & \boldsymbol{\Lambda}_{bb}\end{bmatrix}. \end{aligned} \tag{A.18}$$

We then write out the exponent of the complex Gaussian distribution as

$$\begin{aligned} & -\left(\mathbf{x}-\boldsymbol{\mu}\right)^{H}\boldsymbol{\Sigma}^{-1}\left(\mathbf{x}-\boldsymbol{\mu}\right) \\ =& -\left(\mathbf{x}_a-\boldsymbol{\mu}_a\right)^{H}\boldsymbol{\Lambda}_{aa}\left(\mathbf{x}_a-\boldsymbol{\mu}_a\right)-\left(\mathbf{x}_a-\boldsymbol{\mu}_a\right)^{H}\boldsymbol{\Lambda}_{ab}\left(\mathbf{x}_b-\boldsymbol{\mu}_b\right) \\ & -\left(\mathbf{x}_b-\boldsymbol{\mu}_b\right)^{H}\boldsymbol{\Lambda}_{ba}\left(\mathbf{x}_a-\boldsymbol{\mu}_a\right)-\left(\mathbf{x}_b-\boldsymbol{\mu}_b\right)^{H}\boldsymbol{\Lambda}_{bb}\left(\mathbf{x}_b-\boldsymbol{\mu}_b\right). \end{aligned} \tag{A.19}$$

To get the conditional distribution, we assume $\mathbf{x}_b$ to be given, i.e., we treat it as a constant. We then complete the square over $\mathbf{x}_a$. Collecting the terms quadratic in $\mathbf{x}_a$ gives

$$-\mathbf{x}_a^{H}\boldsymbol{\lambda}_{aa}\mathbf{x}_a. \tag{A.20}$$

By completing the square, we can write the inverse of the conditional precision matrix as

$$\boldsymbol{\Sigma}_{a|b}^{-1}=\boldsymbol{\Lambda}_{aa}. \tag{A.21}$$

So, the conditional covariance matrix is given as

$$\boldsymbol{\Sigma}_{a|b}=\boldsymbol{\Lambda}_{aa}^{-1}. \tag{A.22}$$

collecting linear terms in $\mathbf{x}_a$ gives us

$$\mathbf{x}_a^{H}\left(\boldsymbol{\Lambda}_{aa}\boldsymbol{\mu}_a-\boldsymbol{\Lambda}_{ab}\left(\mathbf{x}_b-\boldsymbol{\mu}_b\right)\right). \tag{A.23}$$

From completing the square, we know that

$$\boldsymbol{\Sigma}_{a|b}^{-1}\boldsymbol{\mu}_{a|b}=\left(\boldsymbol{\Lambda}_{aa}\boldsymbol{\mu}_a-\boldsymbol{\Lambda}_{ab}\left(\mathbf{x}_b-\boldsymbol{\mu}_b\right)\right). \tag{A.24}$$

We now have an expression for the mean of the conditional distribution

$$\begin{aligned} \boldsymbol{\mu}_{a|b} &= \boldsymbol{\Sigma}_{a|b}\left(\boldsymbol{\Lambda}_{aa}\boldsymbol{\mu}_a-\boldsymbol{\Lambda}_{ab}\left(\mathbf{x}_b-\boldsymbol{\mu}_b\right)\right) \\ &= \boldsymbol{\mu}_a-\boldsymbol{\Sigma}_{a|b}\boldsymbol{\Lambda}_{ab}\left(\mathbf{x}_b-\boldsymbol{\mu}_b\right), \end{aligned} \tag{A.25}$$

where we use (A.22). Using (A.22) and (A.25) and filling in the values from (A.10) and (A.15), we can fully characterise the posterior on the weights as

$$p\left(\mathbf{w}|\mathbf{t},\boldsymbol{\alpha},\sigma^2\right)=\mathcal{CN}\left(\mathbf{w}|\boldsymbol{\Sigma}\boldsymbol{\Phi}^{H}\mathbf{B}^{-1}\mathbf{t},\boldsymbol{\Sigma}\right), \tag{A.26}$$

where $\boldsymbol{\Sigma}=\left(\mathbf{A}+\boldsymbol{\Phi}^{H}\mathbf{B}^{-1}\boldsymbol{\Phi}\right)^{-1}$.

We now turn to the Bayesian update equations of $\boldsymbol{\alpha}$ and σ. We derive the update equations by maximising the marginal distribution of $\mathbf{t}$, as expressed in Equation A.17. Since maximising the logarithm of the marginal distribution of $\mathbf{t}$ is equivalent to maximising the marginal distribution of $\mathbf{t}$, we write the logarithm of the marginal distribution of $\mathbf{t}$ as

$$\mathcal{L}\left(\mathbf{t}|\boldsymbol{\alpha},\sigma\right)=-\ln\left|\mathbf{B}+\boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{H}\right|-\mathbf{t}^{H}\left(\mathbf{B}+\boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{H}\right)^{-1}\mathbf{t}, \tag{A.27}$$

where we have only kept the terms depending on $\boldsymbol{\alpha}$ and σ. We rewrite the first term as

$$\begin{aligned} & -\ln\left|\mathbf{B}+\boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^{H}\right| \\ =& -\ln\left|\mathbf{A}^{-1}\right|-\ln\left|\mathbf{B}\right|-\ln\left|\mathbf{A}-\boldsymbol{\Phi}^{H}\mathbf{B}\boldsymbol{\Phi}\right| \\ =& \sum_{i=0}^{M-1}\ln\alpha_i-N\ln\sigma^2+\ln\left|\boldsymbol{\Sigma}\right|, \end{aligned} \tag{A.28}$$

where we use the matrix determinant lemma found in, e.g., Harville [53] stating

$$\left|\mathbf{A}+\mathbf{U}\mathbf{W}\mathbf{V}^{H}\right|=\left|\mathbf{W}\right|\left|\mathbf{A}\right|\left|\mathbf{W}^{-1}+\mathbf{V}^{H}\mathbf{A}^{-1}\mathbf{U}\right|. \tag{A.29}$$

Using the matrix inversion lemma [54], we rewrite the second term of (A.27) as

$$\begin{aligned}
&-\mathbf{t}^H\left(\mathbf{B}+\mathbf{\Phi}\mathbf{A}^{-1}\mathbf{\Phi}^H\right)^{-1}\mathbf{t}\\
=&-\mathbf{t}^H\left(\mathbf{B}^{-1}-\mathbf{B}^{-1}\mathbf{\Phi}\left(\mathbf{A}+\mathbf{\Phi}^H\mathbf{B}^{-1}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^H\mathbf{B}^{-1}\right)\mathbf{t}\\
=&-\sigma^{-2}\left(\mathbf{t}^H\mathbf{t}-\sigma^{-2}\mathbf{t}^H\mathbf{\Phi}\mathbf{\Sigma}\mathbf{\Phi}^H\mathbf{t}\right)\\
=&-\sigma^{-2}\left(\mathbf{t}^H\mathbf{t}-\mathbf{t}^H\mathbf{\Phi}\boldsymbol{\mu}\right)\\
=&-\sigma^{-2}\left(\left\|\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}\right\|_2^2+\mathbf{t}^H\mathbf{\Phi}\boldsymbol{\mu}-\boldsymbol{\mu}^H\mathbf{\Phi}^H\mathbf{\Phi}\boldsymbol{\mu}\right)\\
=&-\sigma^{-2}\left\|\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}\right\|_2^2-\sigma^{-2}\mathbf{t}^H\mathbf{\Phi}\boldsymbol{\mu}+\sigma^{-2}\boldsymbol{\mu}^H\mathbf{\Phi}^H\mathbf{\Phi}\boldsymbol{\mu}\\
=&-\sigma^{-2}\left\|\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}\right\|_2^2-\boldsymbol{\mu}^H\mathbf{\Sigma}^{-1}\boldsymbol{\mu}+\sigma^2\boldsymbol{\mu}^H\mathbf{\Phi}^H\mathbf{\Phi}\boldsymbol{\mu}\\
=&-\sigma^{-2}\left\|\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}\right\|_2^2-\boldsymbol{\mu}^H\mathbf{A}\boldsymbol{\mu},
\end{aligned} \tag{A.30}$$

where $\boldsymbol{\mu}$ is defined as the expectation of the posterior distribution on the weights, i.e., $\boldsymbol{\mu}=\mathbf{\Sigma}\mathbf{\Phi}^H\mathbf{B}^{-1}\mathbf{t}$. Combining the terms allows us to write the log marginal distribution of $\mathbf{t}$ as

$$\mathcal{L}\left(\mathbf{t}|\boldsymbol{\alpha},\sigma\right)=\sum_{i=0}^{M-1}\ln\alpha_i-N\ln\sigma^2+\ln|\mathbf{\Sigma}|-\sigma^{-2}\left\|\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}\right\|_2^2-\boldsymbol{\mu}^H\mathbf{A}\boldsymbol{\mu}. \tag{A.31}$$

To get the update for α_i, we take the derivative of the log marginal distribution with respect to α_i and set it equal to zero.

$$\frac{\partial\mathcal{L}\left(\mathbf{t}|\boldsymbol{\alpha},\sigma\right)}{\partial\alpha_i}=\frac{1}{\alpha_i}-\mathbf{\Sigma}_{ii}-\boldsymbol{\mu}_i^2=0. \tag{A.32}$$

We define

$$\begin{aligned}
\gamma_i=&1-\alpha_i\mathbf{\Sigma}_{ii}\\
\Rightarrow\mathbf{\Sigma}_{ii}=&\frac{1-\gamma_i}{\alpha_i}
\end{aligned} \tag{A.33}$$

and write (A.32) as

$$\begin{aligned}
0=&\frac{1}{\alpha_i}-\mathbf{\Sigma}_{ii}-\boldsymbol{\mu}_i^2\\
0=&\frac{1}{\alpha_i}-\frac{1-\gamma_i}{\alpha_i}-\boldsymbol{\mu}_i^2\\
0=&\gamma_i-\alpha_i\boldsymbol{\mu}_i^2\\
\alpha_i=&\frac{\gamma_i}{\boldsymbol{\mu}_i^2},
\end{aligned} \tag{A.34}$$

which gives the update equation for α_i. For the update of σ^2 we take the derivative of the log marginal distribution of $\mathbf{t}$ with respect to σ^{-2} and again set it equal to zero

$$\begin{aligned}
\frac{\partial\mathcal{L}\left(\mathbf{t}|\boldsymbol{\alpha},\sigma\right)}{\partial\sigma^{-2}}=&\frac{\sigma^2}{N}-\mathrm{tr}\left\{\mathbf{\Sigma}\frac{\partial\mathbf{\Sigma}^{-1}}{\partial\sigma^{-2}}\right\}-\left\|\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}\right\|_2^2\\
=&\sigma^2N-\mathrm{tr}\left\{\mathbf{\Sigma}\mathbf{\Phi}^H\mathbf{\Phi}\right\}-\left\|\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}\right\|_2^2\\
=&\sigma^2N-\mathrm{tr}\left\{\mathbf{\Sigma}\mathbf{\Phi}^H\mathbf{\Phi}+\sigma^2\mathbf{\Sigma}\mathbf{A}-\sigma^2\mathbf{\Sigma}\mathbf{A}\right\}-\left\|\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}\right\|_2^2\\
=&\sigma^2N-\mathrm{tr}\left\{\mathbf{\Sigma}\left(\mathbf{\Phi}^H\mathbf{\Phi}\sigma^{-2}+\mathbf{A}\right)\sigma^2-\sigma^2\mathbf{\Sigma}\mathbf{A}\right\}-\left\|\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}\right\|_2^2\\
=&\sigma^2N-\mathrm{tr}\left\{\left(\mathbf{A}+\sigma^{-2}\mathbf{\Phi}^H\mathbf{\Phi}\right)^{-1}\left(\mathbf{\Phi}^H\mathbf{\Phi}\sigma^{-2}+\mathbf{A}\right)\sigma^2-\sigma^2\mathbf{\Sigma}\mathbf{A}\right\}-\left\|\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}\right\|_2^2\\
=&\sigma^2N-\mathrm{tr}\left\{\left(\mathbf{I}-\mathbf{A}\mathbf{\Sigma}\right)\sigma^2\right\}-\left\|\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}\right\|_2^2\\
=&\sigma^2N-\sigma^2\sum_{i=0}^{M-1}\gamma_i-\left\|\mathbf{t}-\mathbf{\Phi}\boldsymbol{\mu}\right\|_2^2=0,
\end{aligned} \tag{A.35}$$

where we use Jacobi's formula for the derivative of the determinant of a matrix in the first step [55]. We can now write the update equation for σ^2 as

$$\sigma^2 = \frac{\left\|\mathbf{t} - \boldsymbol{\Phi}\boldsymbol{\mu}\right\|_2^2}{N - \sum_{i=0}^{M-1} \gamma_i}. \tag{A.36}$$

B

Derrivation of the complex multitask relevance vector machine

Repeating the model specification, we have $\mathbf{\Phi}_i \neq \mathbf{\Phi}_j$ and $\mathbf{w}_i \neq \mathbf{w}_j \quad \forall i,j \in \{1,2,\dots,B\}$. Allowing the joint measurement vectors equation to be written as

$$\begin{bmatrix} \mathbf{t}_1 \\ \mathbf{t}_2 \\ \vdots \\ \mathbf{t}_B \end{bmatrix} = \begin{bmatrix} \mathbf{\Phi}_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{\Phi}_2 & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{\Phi}_B \end{bmatrix} \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \vdots \\ \mathbf{w}_B \end{bmatrix} + \boldsymbol{\varepsilon}, \tag{B.1}$$

where we define a joint prior on the weights of each individual measurement vector.

$$\begin{aligned} \mathbf{w}_1 &\sim \mathcal{CN}\left(\mathbf{0}, \mathbf{A}^{-1}\right) \\ \mathbf{w}_2 &\sim \mathcal{CN}\left(\mathbf{0}, \mathbf{A}^{-1}\right) \\ &\vdots \\ \mathbf{w}_B &\sim \mathcal{CN}\left(\mathbf{0}, \mathbf{A}^{-1}\right), \end{aligned} \tag{B.2}$$

where $\mathbf{A}$ is a diagonal matrix, like in the regular relevance vector machine, defined as $\mathbf{A} = \mathrm{diag}(\alpha_0, \dots, \alpha_{M-1})$. This is the formulation of the multitask relevance vector machine of Ji *et al.* [40]. In the multitask relevance vector machine, we see the processing of the B bursts as individual tasks that are connected via the prior on the weights. We can update the estimates for weights $\boldsymbol{\mu}_b$, $b \in \{1,2,\dots,B\}$ and the covariance matrix $\mathbf{\Sigma}_b$, $b \in \{1,2,\dots,B\}$ for each individual bursts with the regular update equations

$$\mathbf{w}_b = \mathbf{\Sigma}_b \mathbf{\Phi}_b^H \mathbf{B}^{-1} \mathbf{t}_b \quad \text{and} \quad \mathbf{\Sigma}_b = \left(\mathbf{A} + \mathbf{\Phi}_b^H \mathbf{B}^{-1} \mathbf{\Phi}_b\right)^{-1}, \tag{B.3}$$

respectively. $\mathbf{B}$ is still defined as $\mathbf{B} = \sigma^2 \mathbf{I}$. If we then define

$$\begin{aligned} \boldsymbol{\mu} &= \begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \\ \vdots \\ \boldsymbol{\mu}_B \end{bmatrix}, \qquad \mathbf{\Sigma} = \begin{bmatrix} \mathbf{\Sigma}_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_2 & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{\Sigma}_B \end{bmatrix}, \quad \mathbf{\Phi} = \begin{bmatrix} \mathbf{\Phi}_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{\Phi}_2 & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{\Phi}_B \end{bmatrix}, \\ \mathbf{t} &= \begin{bmatrix} \mathbf{t}_1 \\ \mathbf{t}_2 \\ \vdots \\ \mathbf{t}_B \end{bmatrix}, \qquad \mathbf{A}_B = \begin{bmatrix} \mathbf{A} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{A} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{A} \end{bmatrix}, \quad \text{and} \quad \mathbf{B}_B = \begin{bmatrix} \mathbf{B} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{B} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{B} \end{bmatrix}, \end{aligned} \tag{B.4}$$

we can follow the same manipulations of the logarithm of the marginal distribution of $\mathbf{t}$ as in Appendix A. Resulting in

$$\begin{aligned}\mathcal{L}\left(\mathbf{t}|\boldsymbol{\alpha},\sigma\right) &= \sum_{b=1}^{B}\sum_{i=0}^{M-1}\ln\alpha_i - \ln\left|\mathbf{B}_B\right| + \ln|\boldsymbol{\Sigma}| - \sigma^{-2}\left\|\mathbf{t}-\boldsymbol{\Phi}\boldsymbol{\mu}\right\|_2^2 - \boldsymbol{\mu}^H\mathbf{A}_B\boldsymbol{\mu}. \\ &= B\sum_{i=0}^{M-1}\ln\alpha_i - \sum_{b=1}^{B}N\ln\sigma^2 + \ln|\boldsymbol{\Sigma}| - \sigma^{-2}\left\|\mathbf{t}-\boldsymbol{\Phi}\boldsymbol{\mu}\right\|_2^2 - \boldsymbol{\mu}^H\mathbf{A}_B\boldsymbol{\mu}.\end{aligned} \tag{B.5}$$

To then get the update for α_i, we take the derivative of the log marginal distribution of $\mathbf{t}$ with respect to α_i and set it equal to zero.

$$\frac{\partial\mathcal{L}\left(\mathbf{t}|\boldsymbol{\alpha},\sigma\right)}{\partial\alpha_i} = \frac{B}{\alpha_i} - \sum_{b=1}^{B}\boldsymbol{\Sigma}_{b;i,i} - \sum_{b=1}^{B}\boldsymbol{\mu}_{b;i}^2 = 0, \tag{B.6}$$

where $\boldsymbol{\Sigma}_{b;i,i}$ and $\boldsymbol{\mu}_{b,i}$ denote the (i,i)'th and i'th element of $\boldsymbol{\Sigma}_b$ and $\boldsymbol{\mu}_b$, respectively. By defining

$$\gamma_i = B - \alpha_i\sum_{b=1}^{B}\boldsymbol{\Sigma}_{b;i,i}, \tag{B.7}$$

we can write

$$\sum_{b=1}^{B}\boldsymbol{\Sigma}_{b;i,i} = \frac{B-\gamma_i}{\alpha_i} \tag{B.8}$$

and rewrite (B.6) as

$$\begin{aligned}0 &= \frac{B}{\alpha_i} - \sum_{b=1}^{B}\boldsymbol{\Sigma}_{b;i,i} - \sum_{b=1}^{B}\boldsymbol{\mu}_{b;i}^2 \\ 0 &= \frac{B}{\alpha_i} - \frac{B-\gamma_i}{\alpha_i} - \sum_{b=1}^{B}\boldsymbol{\mu}_{b;i}^2 \\ 0 &= \gamma_i - \alpha_i\sum_{b=1}^{B}\boldsymbol{\mu}_{b;i}^2 \\ \alpha_i &= \frac{\gamma_i}{\sum_{b=1}^{B}\boldsymbol{\mu}_{b;i}^2}.\end{aligned} \tag{B.9}$$

leading to the update equation for α_i. Similarly, for the update of σ^2, we take the derivative of the log marginal distribution of $\mathbf{t}$ with respect to σ^{-2} and again set it equal to zero.

$$\begin{aligned}
\frac{\partial \mathcal{L}\left(\mathbf{t}|\boldsymbol{\alpha},\sigma\right)}{\partial \sigma^{-2}} &= \sum_{b=1}^{B} \frac{\sigma^2}{N} - \mathrm{tr}\left\{\boldsymbol{\Sigma}\frac{\partial \boldsymbol{\Sigma}^{-1}}{\partial \sigma^{-2}}\right\} - \left\|\mathbf{t}-\boldsymbol{\Phi}\boldsymbol{\mu}\right\|_2^2 \\
&= \sum_{b=1}^{B} \sigma^2 N - \mathrm{tr}\left\{\boldsymbol{\Sigma}\boldsymbol{\Phi}^H\boldsymbol{\Phi}\right\} - \left\|\mathbf{t}-\boldsymbol{\Phi}\boldsymbol{\mu}\right\|_2^2 \\
&= \sum_{b=1}^{B} \sigma^2 N - \mathrm{tr}\left\{\boldsymbol{\Sigma}\boldsymbol{\Phi}^H\boldsymbol{\Phi} + \sigma^2\boldsymbol{\Sigma}\mathbf{A} - \sigma^2\boldsymbol{\Sigma}\mathbf{A}\right\} - \left\|\mathbf{t}-\boldsymbol{\Phi}\boldsymbol{\mu}\right\|_2^2 \\
&= \sum_{b=1}^{B} \sigma^2 N - \mathrm{tr}\left\{\boldsymbol{\Sigma}\left(\boldsymbol{\Phi}^H\boldsymbol{\Phi}\sigma^{-2} + \mathbf{A}\right)\sigma^2 - \sigma^2\boldsymbol{\Sigma}\mathbf{A}\right\} - \left\|\mathbf{t}-\boldsymbol{\Phi}\boldsymbol{\mu}\right\|_2^2 \\
&= \sum_{b=1}^{B} \sigma^2 N - \mathrm{tr}\left\{\left(\mathbf{A} + \sigma^{-2}\boldsymbol{\Phi}^H\boldsymbol{\Phi}\right)^{-1}\left(\boldsymbol{\Phi}^H\boldsymbol{\Phi}\sigma^{-2} + \mathbf{A}\right)\sigma^2 - \sigma^2\boldsymbol{\Sigma}\mathbf{A}\right\} - \left\|\mathbf{t}-\boldsymbol{\Phi}\boldsymbol{\mu}\right\|_2^2 \\
&= \sum_{b=1}^{B} \sigma^2 N - \mathrm{tr}\left\{\left(\mathbf{I} - \mathbf{A}\boldsymbol{\Sigma}\right)\sigma^2\right\} - \left\|\mathbf{t}-\boldsymbol{\Phi}\boldsymbol{\mu}\right\|_2^2 \\
&= \sum_{b=1}^{B} \sigma^2 N - \sigma^2 \sum_{b=1}^{B}\sum_{i=0}^{M-1}\left(1 - \alpha_i\boldsymbol{\Sigma}_{b;i,i}\right) - \left\|\mathbf{t}-\boldsymbol{\Phi}\boldsymbol{\mu}\right\|_2^2, \\
&= \sum_{b=1}^{B}\left(\sigma^2 N - \sigma^2 M + \sigma^2\sum_{i=0}^{M-1}\boldsymbol{\Sigma}_{b;i,i}\right) - \left\|\mathbf{t}-\boldsymbol{\Phi}\boldsymbol{\mu}\right\|_2^2 \\
&= \sigma^2\sum_{b=1}^{B}\left(N - M + \sum_{i=0}^{M-1}\boldsymbol{\Sigma}_{b;i,i}\right) - \sum_{b=1}^{B}\left\|\mathbf{t}_b - \boldsymbol{\Phi}_b\boldsymbol{\mu}_b\right\|_2^2 = 0,
\end{aligned} \tag{B.10}$$

where we again use Jacobi's formula for the derivative of the determinant of a matrix in the first step [55]. We can now write the update equation for σ^2as

$$\sigma^2 = \frac{\sum_{b=1}^{B}\left\|\mathbf{t}_b - \boldsymbol{\Phi}_b\boldsymbol{\mu}_b\right\|_2^2}{\sum_{b=1}^{B}\left(N - M + \sum_{i=0}^{M-1}\boldsymbol{\Sigma}_{b;i,i}\right)}. \tag{B.11}$$

C

Roots of a polynomial

Here, we describe a method to get the roots of a polynomial of order N. Let $f(x)$ be a polynomial of order N, given by

$$f(x) = a_N x^N + a_{N-1} x^{N-1} + \dots a_1 x + a_0. \tag{C.1}$$

We then normalise the polynomial by dividing by a_N, to get

$$f_n(x) = x^N + b_{N-1} x^{N-1} + \dots b_1 x + b_0, \tag{C.2}$$

where b_i is then given as

$$b_i = \frac{a_i}{a_N} \quad \forall i \in \{1, 2, \dots, N-1\}. \tag{C.3}$$

We then write the $N \times N$ Frobenius companion matrix, $\mathbf{C}$, of the polynomial $f(x)$ as

$$\mathbf{C} = \begin{bmatrix} b_{N-1} & b_{N-2} & \dots & b_2 & b_1 & b_0 \\ 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 & 0 \\ 0 & 0 & \dots & 0 & 1 & 0 \end{bmatrix}. \tag{C.4}$$

The roots of the polynomial $f(x)$ are then given by the eigenvalues of the Frobenius companion matrix $\mathbf{C}$.

D

Swerling fluctuation and the Rayleigh distribution

D.1. Swerling cases

The four Swerling cases, introduced by Swerling [56], describe how the returned signal power per pulse fluctuates. Swerling describes these fluctuations in four cases.

The first two cases represent targets that can be modelled by several independently fluctuating reflectors of approximately equal scattering area. This fluctuation in received signal power is modelled by drawing from a chi-squared distribution with two degrees of freedom.

The third and fourth cases represent targets that can be modelled by one large reflector with other small reflectors, or as a single large reflector, subject to relatively small changes in its orientation. Swerling models this type of fluctuation in received signal power with a chi-squared distribution with four degrees of freedom

In cases one and three, amplitudes are assumed to vary from scan to scan, but the amplitudes stay constant from burst to burst. In cases two and four, amplitudes are assumed to vary from burst to burst.

D.2. Link to the Rayleigh and complex Gaussian distributions

A Rayleigh distribution is defined by a random variable that follows the distribution of the square root of two independent zero-mean normal random variables with equal variance. I.e., if

$$R = \sqrt{X^2 + Y^2}, \tag{D.1}$$

where

$$X \sim N(0, \iota), Y \sim N(0, \iota). \tag{D.2}$$

Then

$$R \sim Rayleigh(\iota), \tag{D.3}$$

where ι is the scale parameter of the distribution. The pdf of the Rayleigh distribution is given by

$$f(x, \iota) = \frac{x}{\iota^2} \exp\left\{-\frac{x^2}{2\iota^2}\right\}, \quad x \in [0, \infty) \tag{D.4}$$

In the case of Swerling one or two fluctuation, the received signal power follows a chi-squared distribution with two degrees of freedom. A chi-squared distribution with two degrees of freedom is distributed according to the sum of squares of two independent standard normal distributions. I.e., if

$$Q = \sum_{i=1}^{2} Z_i^2, \tag{D.5}$$

where Z_i are i.i.d. standard normal random variables. Then,

$$Q \sim \chi_2^2. \tag{D.6}$$

In the case of Swerling one or two fluctuation, the absolute value of the amplitude of the received signal will be distributed according to a distribution that follows from taking the square root of a random variable that follows a chi-squared distribution with two degrees of freedom. By comparing the definitions of a chi-squared distribution with two degrees of freedom in (D.5) and the Rayleigh distribution in (D.1), we can see that the square root of a random variable that is distributed according to a chi-squared distribution with two degrees of freedom follows a Rayleigh distribution with a scale parameter equal to one.

Additionally, the absolute value of complex normally disturbed random variables also follows a Rayleigh distribution, as both the real and imaginary parts of a complex normally distributed random variable follow a normal random variable themselves.

D.3. Second moment of the Rayleigh distribution

The second moment of the Rayleigh distribution is given by

$$\mathbb{E}\left[x^2\right] = \int_0^\infty \frac{x^3}{\iota^2} \exp\left\{-\frac{x^2}{2\iota^2}\right\} \mathrm{d}x. \tag{D.7}$$

Performing a change of variables on $t = \frac{x^2}{2\iota^2}$ results in

$$\mathbb{E}\left[x^2\right] = \iota^2 \int_0^\infty t \exp\{-t\} \,\mathrm{d}t. \tag{D.8}$$

By using integration by parts, we write

$$\iota^2 \int_0^\infty t \exp\{-t\} \,\mathrm{d}t = \iota^2 \int_0^\infty u \mathrm{d}v = \iota^2 \left([uv]_0^\infty - \int_0^\infty v \mathrm{d}u \right), \tag{D.9}$$

where

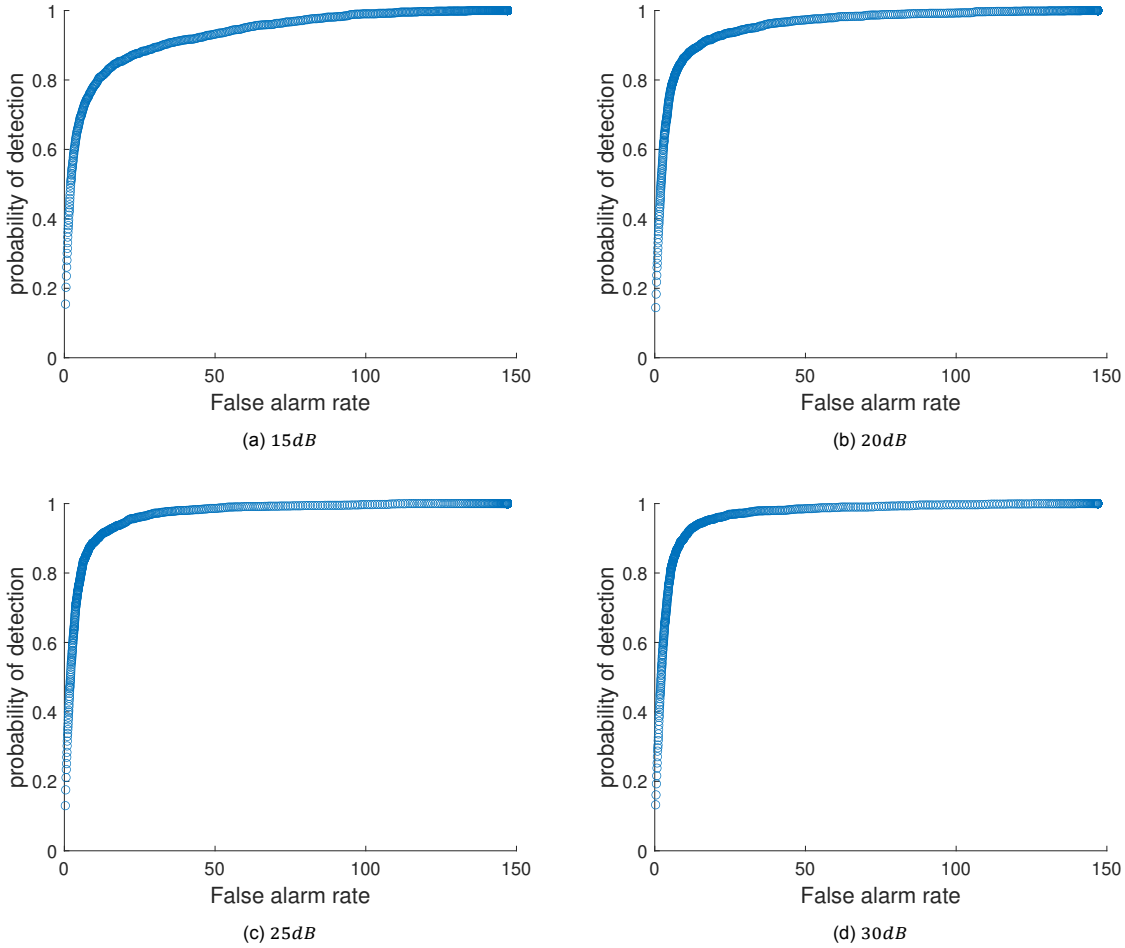
$$u = t, \quad \mathrm{d}u = 1, \quad \mathrm{d}v = \exp\{-t\}, \quad v = -\exp\{-t\}. \tag{D.10}$$

So,

$$\begin{aligned} \mathbb{E}\left[x^2\right] &= \iota^2 \left(\left[-\exp\{-t\}\right]_0^\infty + \int_0^\infty \exp\{-t\} \,\mathrm{d}t \right) \\ &= \iota^2 \left(\left[-\exp\{-t\}\right]_0^\infty + \left[-\exp\{-t\}\right]_0^\infty \right) \\ &= 2\iota^2. \end{aligned} \tag{D.11}$$

E

Additional figures



(a) $15dB$ (b) $20dB$

(c) $25dB$ (d) $30dB$

Figure E.1: Plots of the estimated ROC curves for the matched filter method. The simulation setup is as specified in the simulation of a general case. Each scatterplot corresponds to the following SNR: (a) 15 dB, (b) 20 dB, (c) 25 dB, (d) 30 dB.