

Signals and Communication Technology

Jonah Gamba

Radar Signal Processing for Autonomous Driving

 Springer

Signals and Communication Technology

Series Editors

Emre Celebi, Department of Computer Science, University of Central Arkansas,
Conway, AR, USA

Jingdong Chen, Northwestern Polytechnical University, Xi'an, China

E. S. Gopi, Department of Electronics and Communication Engineering, National
Institute of Technology, Tiruchirappalli, Tamil Nadu, India

Amy Neustein, Linguistic Technology Systems, Fort Lee, NJ, USA

H. Vincent Poor, Department of Electrical Engineering, Princeton University,
Princeton, NJ, USA

This series is devoted to fundamentals and applications of modern methods of signal processing and cutting-edge communication technologies. The main topics are information and signal theory, acoustical signal processing, image processing and multimedia systems, mobile and wireless communications, and computer and communication networks. Volumes in the series address researchers in academia and industrial R&D departments. The series is application-oriented. The level of presentation of each individual volume, however, depends on the subject and can range from practical to scientific.

“Signals and Communication Technology” is indexed by Scopus.

More information about this series at <http://www.springer.com/series/4748>

Jonah Gamba

Radar Signal Processing for Autonomous Driving

 Springer

Jonah Gamba
Tsukuba, Ibaraki, Japan

ISSN 1860-4862 ISSN 1860-4870 (electronic)
Signals and Communication Technology
ISBN 978-981-13-9192-7 ISBN 978-981-13-9193-4 (eBook)
<https://doi.org/10.1007/978-981-13-9193-4>

© Springer Nature Singapore Pte Ltd. 2020

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd. The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

To Megumi, Sekai, Mirai, and the family

Preface

This book is a result of the painful experience of learning from scratch with little reference material directly dealing in a collective manner with the application of radar processing principles to automotive applications. Radar processing as a topic is well matured and has been around for decades. The main focus has been on military applications such as missile tracking and navigation. However, the arrival of non-traditional applications in the form of advanced driver assistance systems exposed the inadequacy of volumes of material already available on the topic. Therefore, a holistic approach starting from the basics to detailed aspects is necessary. Consequently, this book is meant to be the groundbreaker for such an approach providing the big picture to the interested reader on full range of radar principles required for the autonomous driving applications.

The subject of this book is theory, principles, and methods used in radar algorithm development with a special focus on automotive radar signal processing. In the automotive industry, autonomous driving is currently a hot topic that has led to numerous applications for both safety and driving comfort. It is estimated that full autonomous driving will be realized in the next twenty to thirty years and one of the enabling technologies is radar sensing. However, there are very few books in the market covering essential tools and methods necessary for the successful implementation of the processing algorithms. There is also no single self-contained text describing the full depth and breadth of the required foundation for building these applications making it necessary for both professionals and newcomers to gather information from multiple texts scattered in the literature. Our main intention in this book is to address the problem by presenting both detection and tracking topics specifically for automotive radar processing. It is meant to be a one-stop solution to the abovementioned problem.

The content given in this book is a result of several years of research and development experience in the automotive industry. Based on this experience, this book provides illustrations, figures, and tables for the reader to quickly grasp the concepts and start working practical solutions. A complete and comprehensive coverage of all the essential methods and tools required to successfully implement and evaluate automotive radar processing algorithms is presented. In addition,

instructive references are given as a bridge to connect to the vast ideas that have been studied in the past. From the ideas, deep insights can be gained for tailoring particular methods to the autonomous driving applications. The chapters are organized as follows:

Chapter 1 Fundamentals of Radar Systems. This chapter presents the basics of radar systems. It also describes the importance of antennas to radar systems. In addition, it gives an overview of current automotive applications where radar systems are increasingly being incorporated. Automotive radar processing will be the main focus of this book.

Chapter 2 The Radar Equation. This chapter introduces the radar equation which is essential for understanding the effects of propagation on the transmitted radar signal. It outlines the effect of propagation losses on the maximum detectable range. This is illustrated by an example with typical automotive radar parameters.

Chapter 3 Signal Processing for Radar Systems. This chapter introduces key ideas behind signal processing of received radar signal. It deals with Fourier transforms and their applications to radar signal processing. It also gives a wide coverage of window functions that are typically used in processing radar signals. Extension of these concepts to two-dimensional processing is concisely presented.

Chapter 4 Radar Waveforms and Their Mathematical Models. This chapter deals with various radar waveforms derived from the frequencies that are used in target detection and localization. Waveform design is a critical part of automotive radar development. Depending on target range, velocity, and angle, the waveform can be chosen carefully to meet the requirements. Among the waveforms, FMCW and fast chirp ramp sequence are explained with illustrations. The essential mathematical relations that affect radar performance are given. The effects of parameter selection are also considered.

Chapter 5 Radar Target Detection. Target detection is one of the very first operations that has to be performed on received radar signal. The purpose of detection is normally to distinguish genuine target reflection from noise and clutter. This chapter discusses the main concepts and methods used for radar detection, especially with reference to Sterling target models. Topics related to constant false alarm rate detections are explained to give an insight to both new seasoned readers.

Chapter 6 Direction of Arrival (DOA) Estimation. DOA estimation forms the third component of the radar cube: range, velocity, and angle. In practice, DOA estimation is often complicated by the fact that there will be a multiple and unknown number of source signals impinging on the receiver array at the same time, with unknown amplitudes. This chapter gives a detailed presentation of DOA methods. The aim of this chapter is to give some of the actively and continuously researched methods in DOA estimation that are a key part of automotive radar algorithms. The strengths and the limitations of the most popular methods are given to allow informed decision of the best choice for a given target application.

Chapter 7 Target Filtering and Tracking. Based on range, velocity, and DOA, target tracking is a key part of automotive radar processing. Filtering enables the radar system to capture the targets' motion dynamics in real time. This chapter

introduces the key methods based on conventional Kalman filtering. Additionally, advanced recent approaches based on Bayesian theory are introduced. The idea is to give a good understanding of the tracking approaches and also indicate some of the points that present challenges in practical applications. Finally, we also touch on data association which cannot be avoided for multi-target processing.

Chapter 8 Target Recognition and Classification Techniques. This chapter introduces a hot topic in automotive radar processing. Recognition and classification are normally associated with image processing but the use of these techniques radar processing is increasing. This trend is driven by regulatory requirements which are making it necessary to distinguish objects in the driving environment. The demand for recognition techniques is even more important in the autonomous driving setting since human control has to be eventually eliminated. In this chapter, we cover machine learning concepts and also touch on deep learning algorithms, providing examples to drive the point home.

Chapter 9 Automotive Radar Applications. This chapter gives some of the key applications categorizing them into short-range radar (SRR) and long-range radar (LRR) areas. Radar requirements are given along to illustrate the challenges involved in the radar systems. This chapter ends with future directions although I should say that the changes in this field are so rapid that prediction is a risky business.

I hope that the material presented in this book will be valuable to all readers, both new and already working on the next-generation autonomous driving algorithms. The degree of success of this book can be judged by the level of reduction of the pain required to start building and evaluating radar processing algorithms for future vehicles.

I hope that this new perspective that I have attempted to elaborate will find reverberations in the automotive industry.

Tsukuba, Japan
2019

Jonah Gamba

Acknowledgements

I would like to thank all the people who have in some ways, either directly or indirectly, contributed to the preparation of this book.

First and foremost, I would like to thank my family members, Megumi, Sekai, and Mirai, for their patience during this very long process. It requires a lot of understanding to accommodate the time spent on putting material together and almost always glued to the computer. Your support is greatly appreciated. Special thanks to Dr. Courage Kamusoko and Prof. Hiromi Murakami, formerly Seikei University, for their continuous encouragement during the process of putting this book together.

I would also like to express my sincere gratitude to Honjo Scholarship Foundation for funding my research in signal processing during my doctoral degree program. Through their kindness, I was able to pursue my interest in signal processing, which finally led to the topic of this book.

Many thanks also to Divya Meiyazhagan, Ramesh Premnath, Chew Low and the production team of Springer for their very efficient and professional support in order to make this book publishable.

Jonah Gamba

Contents

1	Fundamentals of Radar Systems	1
1.1	Introduction	1
1.2	Essential Functions of Radar	1
1.3	Radar System Fundamentals	2
1.4	Antennas for Radar Measurements	3
1.5	Antenna Arrays Basics	3
1.6	An Outline of Automotive Radar Applications	11
1.7	Challenges for Automotive Radar Developers	12
	References	13
2	The Radar Equation	15
2.1	Introduction	15
2.2	Radar Performance Requirements	15
2.3	The Radar Equation	15
2.4	Effect of Losses	16
2.5	Radar Equation for Automotive Applications	17
	References	20
3	Signal Processing for Radar Systems	23
3.1	Introduction	23
3.1.1	Definition of Fourier Transform	23
3.1.2	Fourier Transform Properties	24
3.1.3	Fourier Series	26
3.1.4	Sampling Theorem	27
3.1.5	Discrete Fourier Transforms (DFT)	27
3.1.6	Power Spectrum Estimation	28
3.1.7	Windowing Techniques	29

3.2	Multi-dimensional Fourier Transforms (Basis for Range Doppler Estimation)	33
3.3	Noise and Clutter Reduction in Radar Signals	35
	References	35
4	Radar Waveforms and Their Mathematical Models	37
4.1	Introduction	37
4.2	Waveform Types Overview	37
4.3	Continuous Waveform (CW)	38
4.4	Pulse Doppler Radar (PDR) Waveform	38
4.5	Frequency-Modulated CW (FMCW) and Their Variations	38
4.5.1	Linear Frequency-Modulated CW (LFMCW)	39
4.5.2	Stepped FMCW	40
4.5.3	Multi-frequency Shift Keying (MFSK)	43
4.5.4	Interrupted FMCW (FMICW)	44
4.6	Fast Chirp Ramp Sequence Waveform	45
	References	51
5	Radar Target Detection	53
5.1	Introduction	53
5.2	Target Models (Sterling 1–Sterling 4 Models)	53
5.3	Peak Detection	56
5.3.1	Fixed Threshold	57
5.3.2	Multi-cell Thresholding	57
5.3.3	Constant False Alarm Detection (CFAR)	57
	References	62
6	Direction of Arrival (DOA) Estimation	65
6.1	Introduction	65
6.2	Classification of DOA Estimation Methods	65
6.3	Approaches to DOA Estimation	66
6.3.1	Signal Model	66
6.3.2	DOA Estimation Methods	67
6.3.3	Spatial Smoothing	74
6.3.4	Other DOA Algorithms	77
6.3.5	Multi-dimensional DOA Algorithms	83
6.3.6	Recent Approaches to Estimation	84
	References	85
7	Target Filtering and Tracking	87
7.1	Introduction	87
7.2	Kalman Filter	87
7.3	Bayesian Filtering (Sequential Monte Carlo)	95

7.4 Data Association in Multi-target Tracking 99

7.5 Challenges in Target Filtering and Tracking 102

References 103

8 Target Recognition and Classification Techniques 105

8.1 Introduction 105

8.2 Methods of Target Recognition 106

8.3 Challenges and Trends in Target Recognition Based on Radar . . . 117

 8.3.1 Challenges 117

 8.3.2 Current Trends 118

References 120

9 Automotive Radar Applications 123

9.1 Introduction 123

9.2 Short-Range Radar (SRR) 125

9.3 Long-Range Radar (LRR) 131

9.4 Trends in Automotive Applications 135

 9.4.1 Future Roadmaps Automotive Applications 135

 9.4.2 Future Contributions of Automotive Applications 137

 9.4.3 Future Directions and Conclusion 141

References 141

Chapter 1

Fundamentals of Radar Systems



1.1 Introduction

Radar is a device that employs electromagnetic waves to determine the existence and location of objects by relying on the strength of received reflected waves. The term radar is an acronym for *radio detection and ranging* [1, 2].

On the electromagnetic spectrum, depending on the target application, radar stretches from the high-frequency (HF) band of 3–30 MHz for over-the-horizon (OTH) radar to mm (millimeter) band of 40–300 GHz for automotive, autonomous vehicle navigation, high-resolution meteorological observation, and imaging, among other applications [3]. Current automotive radars employ frequencies in the 24–100 GHz range, but the tendency to move frequencies above 100 GHz frequency is an active research and development topic. It is worth noting that at higher frequencies, signal attenuation increases, thereby limiting the detection range. According to [2, 4], attenuation by due to water vapor has maxima at 22.24 GHz and at about 184 GHz, while attenuation due to oxygen molecules peaks at 60 and 118 GHz. For oxygen molecules, the attenuation attains a value greater than 10 dB/km at 60 GHz, while for water vapor it approaches 1 dB/km at 60 GHz. Below 1 GHz, the effect of atmospheric attenuation can be considered negligible, and important above 10 GHz. Figure 1.1 illustrates the position of radar applications on the electromagnetic spectrum.

1.2 Essential Functions of Radar

The key functions of the radar are to detect, locate, and in most cases, track objects of interest. In automotive applications, the objects of interests are vehicles, pedestrians, bicycles, motorcycles, etc., including obstacles that are found on or along roads. In most recent cases under research, the radar is also tasked with recognition and classification of these objects [5–8].

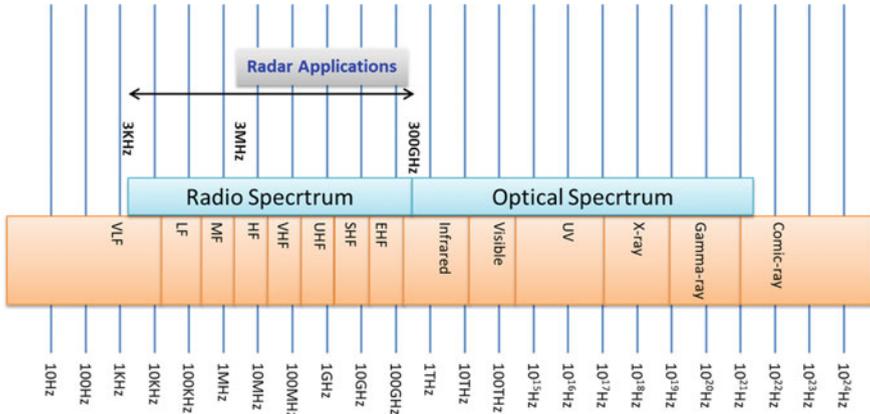


Fig. 1.1 Illustration of the position of radar signals on the electromagnetic spectrum. Radar spreads from 3 to 300 MHz in the radio spectrum

1.3 Radar System Fundamentals

Generally, radar systems consist of three major subsystems: the transmitter, the receiver, and signal processing subsystems as shown in Fig. 1.2. The antenna serves as the electromagnetic interface to the outside.

The transmitter subsystem functions as the source of the signal. The ranging capabilities of the radar are mainly determined by the transmitter design. Therefore, the power generated and the associated costs of the transmitter are important character-

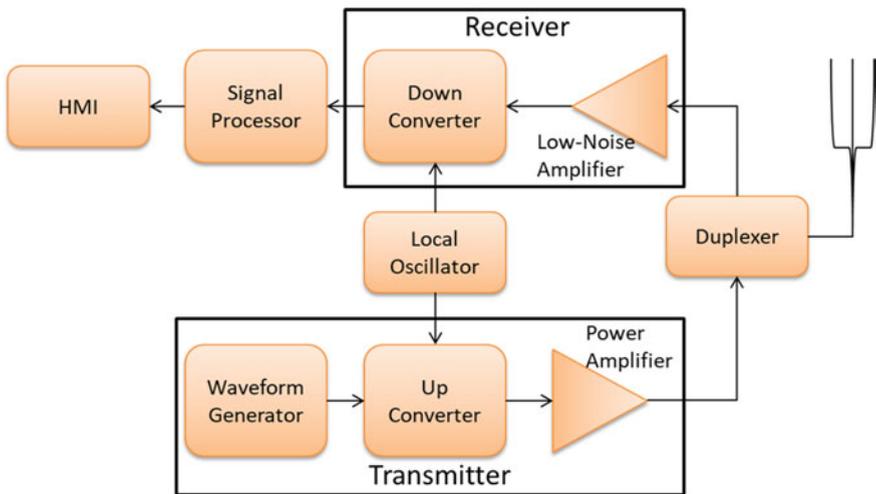


Fig. 1.2 Components of a simplified radar system

istics to consider. Moreover, for all radar applications, the power that is radiated by the antenna is regulated which further places constraints on the transmitter design. As will be shown in the following sections, the detectable maximum range is proportional to the fourth root of the transmitted power. This means that, in order to double the detectable range, the power must be increased by sixteen times, i.e., by 2^4 . The transmitter is made of the waveform generator, the up converter, and the power amplifier. The various waveforms that are used in radar systems will be discussed in the following chapters.

The receiver subsystem functions as the receiver of the usually known reflected signal. Due to the presence of clutter and other unwanted signals, the receiver should ideally maximize the signal-to-noise ratio (SNR) of the desired signal by rejecting or suppressing unwanted signals. The main components of the receiver are the low-noise amplifier (LNA), and the down converter.

The function of the signal processing subsystem applies various algorithms on the received signal to extract useful information that can be used to determine the object position, for tracking and object identification. Since most radar systems operate in the presence of clutter and noise, powerful signal processing techniques are a necessity. On the other hand, due to limitation of hardware resources, installation space, and other constraints, there is a limit to the complexity of the algorithms that can be implemented, especially for real-time automotive and navigation applications.

1.4 Antennas for Radar Measurements

As outlined in Sect. 1.3, in order to transmit and receive EM signals, radar requires an antenna. There are various types of antennas that are used in automotive radars. These include planar, waveguide, lens, and reflector antennas [9, 10]. The planar antennas are gaining widespread use due to their low cost and simpler mounting requirements. The scanning mechanism can either be mechanical or electronic although the general trend is to utilize electronic scanning. Electronic scanning eliminates moving parts which could be problematic when it comes to maintenance.

1.5 Antenna Arrays Basics

Array processing is concerned with processing of signals acquired from antenna arrays. An antenna array is a set of multiple antennas from which the signals are combined or processed in order to achieve improved performance when compared to an individual antenna. The main purpose of antenna array is to increase signal strength, increase the directivity, reduce sidelobe power, increase signal-to-noise ratio (SNR), maximize signal-to-noise-plus-interference ratio (SNIR), and increase antenna gain [11].

The main task of the antenna designer is to minimize losses, reduce size, and reduce the cost of array antenna. Common array configurations include linear arrays, planar arrays rectangular arrays, and circular arrays, among other arrays. To understand the basic operations of the antenna, we derive some relations for the simplest and most common antenna configuration, which is the linear array.

Linear Antenna Arrays

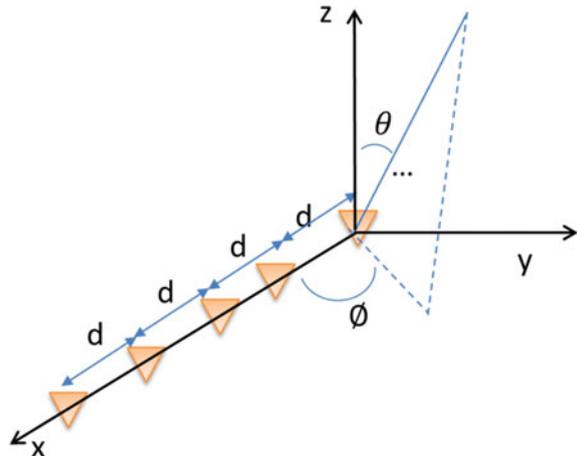
Consider a linear antenna array of N spatially separated antenna elements, where N is an integer greater than or equal to two. The performance of an antenna array increases with the number of antenna elements in the array at the expense of increased size, complexity, and cost. However, recent advances in antenna design and processing methods such as virtual array techniques are promising low-cost antennas at reduced size and complexity.

In the application of antennas arrays, array directivity, array steering, and array beamwidth are of great interest and play an important role in determining the expected behavior of the antenna. We use the uniform linear array antenna as an example to get a better understanding of these characteristics in relationship to the antenna geometry. The uniform linear array consists of uniformly spaced antenna elements. The uniform linear array is chosen because of its wide use and its simplicity.

The radiation properties of the array can be explored from the array factor. The array factor of linear array with equally spaced N isotropic radiating elements placed along horizontal axis x as a function of the angles θ and ϕ in a spherical coordinate system shown in Fig. 1.3 can be expressed by following equation

$$AF_{LIN}(\theta, \phi) = \sum_{n=1}^N I_n e^{j(\delta_n + kd_n \sin \theta)} \quad (1.1)$$

Fig. 1.3 Illustration of a five-element uniform linear array antenna along the x -axis



where I_n and δ_n ($n = 1, 2 \dots N$) are the amplitude and phase excitation of n th array element, d = distance between two adjacent elements, and wave number $k = \frac{2\pi}{\lambda}$, (λ = wavelength).

Values I_n and δ_n are determined by specific design of beam forming network. Typically, the array factor is expressed by an absolute value Eq. (1.1) normalized to its maximum and is plotted in dB scale.

For the uniform amplitude and equal phase distributions ($I_n = I$ and $\delta_n = \delta$, $n = 1, 2 \dots N$), normalized array factor is given by

$$|\text{AF}_{\text{LIN}}(\theta, \phi)| = \frac{1}{N} \left| \frac{\sin(N * kd * \frac{\sin\theta}{2})}{\sin(kd * \frac{\sin\theta}{2})} \right|. \quad (1.2)$$

It is seen that the linear array factor (1.2) is independent of ϕ values, and since it is of the form $f(x) = \sin(Nx)/(N * \sin(x))$, it has a maximum equal to 1 for the angle direction $\theta = 0$. It follows that array factor before normalization (i.e., before division by N) has a maximum value of N . As can be seen, the function (1.2) also has maximum value for the following angle directions (grating lobe angle directions)

$$\theta_r = \pm \sin^{-1}\left(\lambda * \frac{r}{d}\right), \quad r = 1, 2, \dots \quad (1.3)$$

The array factor, AF, can be considered as the spatial analog of a low-pass finite-impulse response averaging filter in discrete-time digital signal processing. It may also be viewed as a window-based narrow-beam design using a rectangular window.

If the distance between the adjacent elements is equal or less than the wavelength λ , linear antenna array has only one beam peak within the visible observation angle region (-90° to 90°). When $d > \lambda$, the unwanted beam peak (grating lobe) occurs in the real angle range of (-90° to 90°). Therefore, observation angle range dictates the value of the maximum element spacing to avoid the occurrence of the grating lobe. For example, if the observation angle occurs in the range -30° and 30° , array element spacing can be chosen as 2λ . As it follows from Eq. (1.2), the value of the maximum sidelobe (with respect to the main beam peak) for the array with uniform amplitude distribution is about -13.1 dB, and the angle direction of this lobe can be estimated from the following expression

$$\theta_{\text{maxlobe}} = \pm \sin^{-1}\left(3\lambda * \frac{1}{2Nd}\right). \quad (1.4)$$

As can be seen from Figs. 1.4, 1.5, 1.6, and 1.7, one-wavelength spacing generates grating lobes with an magnitude that is equal to the main lobe value. Values for the maximum sidelobes are around -13 dB which follows from the expression (1.2). The beamwidth of the main lobe between two adjacent nulls is about

$$\Delta\theta \cong \frac{2\lambda}{Nd}. \quad (1.5)$$

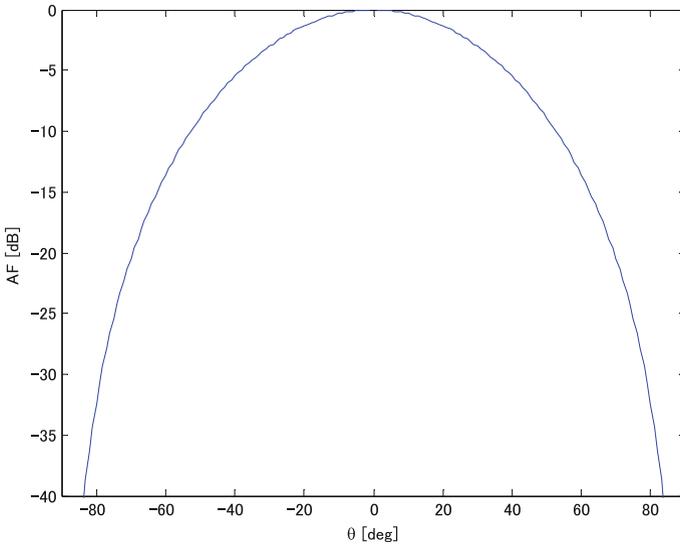


Fig. 1.4 Array factor for a two-element array, with array spacing $d = \lambda/2$

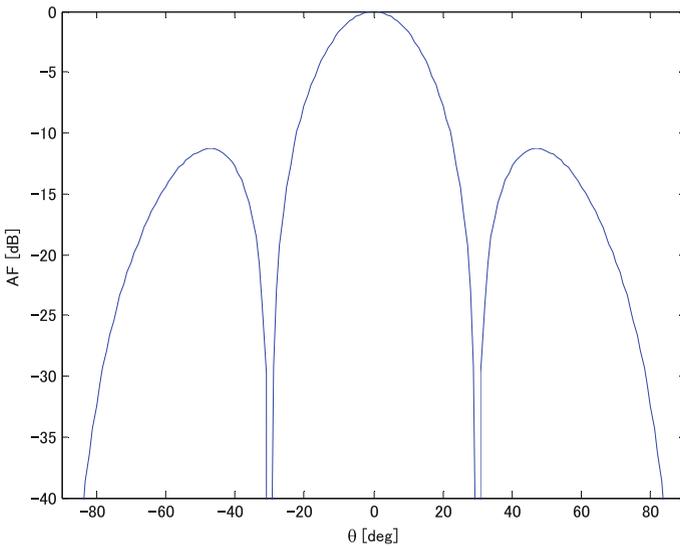


Fig. 1.5 Array factor for four-element array, with element spacing of $d = \lambda/2$. The increase in directivity is evident, but unwanted sidelobes on either side of the main beam arise

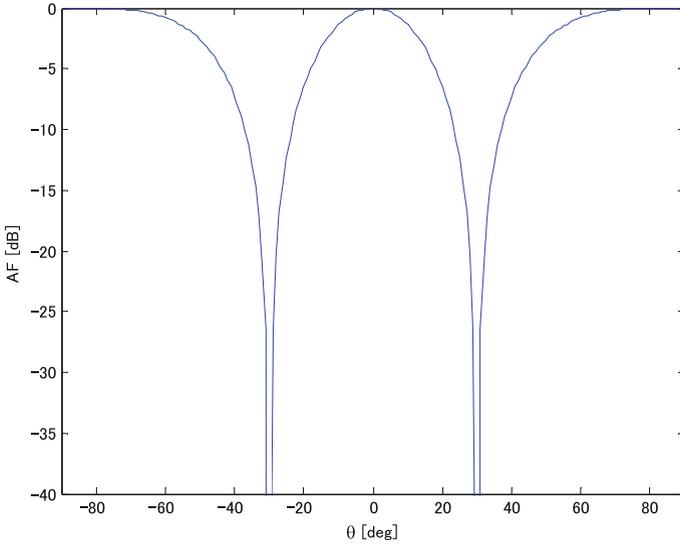


Fig. 1.6 Array factor for a two-element array with an element spacing of $d = \lambda$

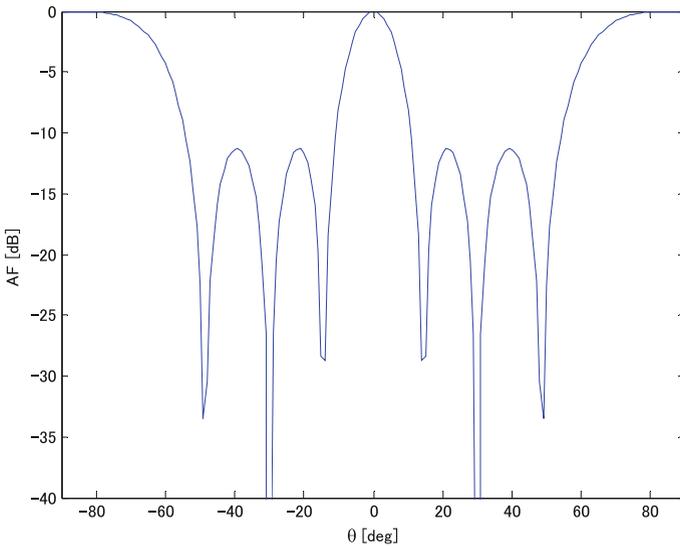


Fig. 1.7 Array factor for a four-element array with an element spacing of $d = \lambda$

The directivity of the broadside linear array factor toward the normal to the array with uniform amplitude and phase distribution is given by [12]

$$D_{\text{LIN}} = \frac{N^2}{N + 2 \sum_{n=1}^{N-1} (N - n) * \text{sinc}(nkd)}, \quad (1.6)$$

where $\text{sinc}(x) = \sin(x)/x$ is the cardinal function. From Eq. (1.6), it can be shown that the directivity increases as the number of antenna elements increases but drops when grating lobes are present.

Example Given an 11-element antenna with a spacing of 2.5λ , then the main lobe beamwidth can be estimated as

$$\Delta\theta \cong \frac{2\lambda}{Nd} = 2 * \frac{\lambda}{11 * 2.5\lambda} = 0.0727[\text{rad}] = 4.17[\text{deg}].$$

Consider a linear array of two antenna elements of spacing d meters as shown in Fig. 1.8. From basic mathematics, the phase difference between the two antennas is a result of the distance x . We can express this distance as $x = d \sin(\theta)$ and the corresponding time difference of arrival as Δt . Then, we can express the radian phase difference $\Delta\phi$ between antennas 1 and 2 two as

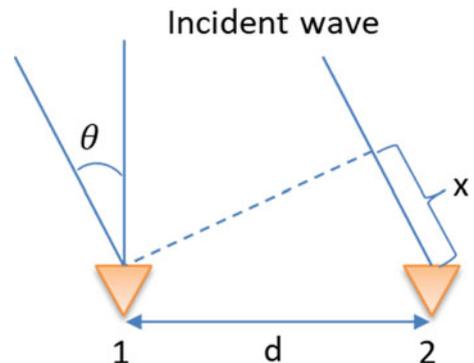
$$\Delta\phi = \omega * \Delta t = \omega * \frac{x}{c} = \omega * \frac{d * \sin(\theta)}{c}, \quad (1.7)$$

where c is the speed of light (electromagnetic waves) and ω is the angular velocity. The above expression (1.7) can be simplified to

$$\Delta\phi = 2\pi f * \frac{d * \sin(\theta)}{f\lambda} = \frac{2\pi d * \sin(\theta)}{\lambda} = \frac{2\pi d}{\lambda} \sin(\theta). \quad (1.8)$$

The factor $k_0 = \frac{2\pi d}{\lambda}$ is sometimes referred to as the wave number. The above two antenna concept can be extended to N -element linear array antenna so that the expression becomes

Fig. 1.8 Two-element linear array example



$$\Delta\phi = \frac{2\pi(N-1)d}{\lambda} \sin(\theta). \quad (1.9)$$

Theoretically, the value of d is conveniently set to $\frac{\lambda}{2}$ as this avoids grating lobes in the antenna's field of view (FOV). Therefore, with $d = \frac{\lambda}{2}$, the phase difference at the n -th antenna with reference to the antenna 1 is given by

$$\Delta\phi = \frac{2\pi(n-1) * \lambda/2}{\lambda} \sin(\theta) = \pi(n-1) \sin(\theta). \quad (1.10)$$

For the general case where $d = k\lambda$, and k is constant, the phase difference is given by

$$\Delta\phi = \frac{2\pi(n-1)*k\lambda}{\lambda} \sin(\theta) = 2k\pi(n-1) \sin(\theta). \quad (1.11)$$

It follows that the phase difference is a function of the direction of arrival for a given array configuration.

Examples (1) Phase difference for $n = 2$ and $d = \frac{\lambda}{2}$ is shown in Fig. 1.9.

(2) Phase difference for $n = 2$ and $d = \frac{\lambda}{4}$ is shown in Fig. 1.10.

(3) Phase difference for $n = 2$ and $d = \lambda$ is shown in Fig. 1.11.

In practice, phase measurements are restricted to the range $[-\pi, \pi]$. This is due the fact phase angles which are shifted by multiples of $\pm 2\pi$ outside the interval $[-\pi, \pi]$ cannot be distinguished from angles within the interval $[-\pi, \pi]$. Specifically, angles

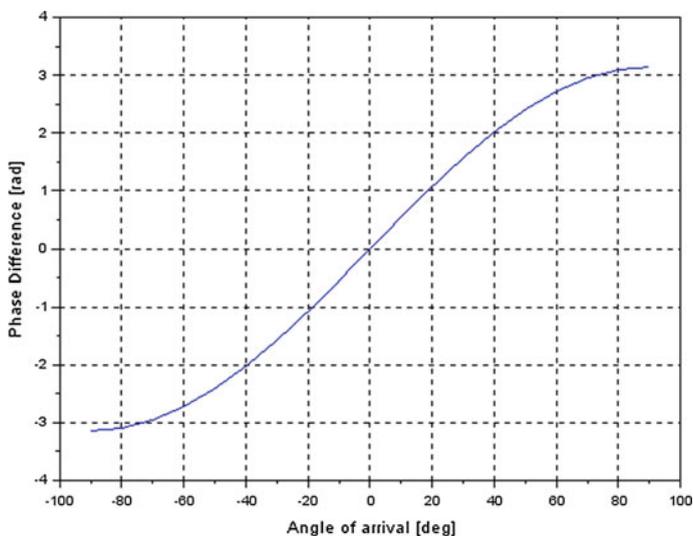


Fig. 1.9 Phase change for $n = 2$ and $d = \frac{\lambda}{4}$ for θ in the range $[-90, 90]$ degrees

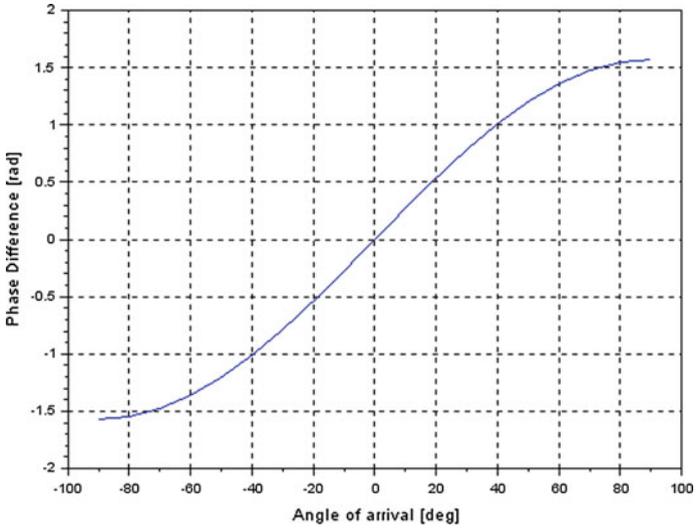


Fig. 1.10 Phase change for $n = 2$ and $d = 0.25\lambda$ for θ in the range $[-90, 90]$ degrees

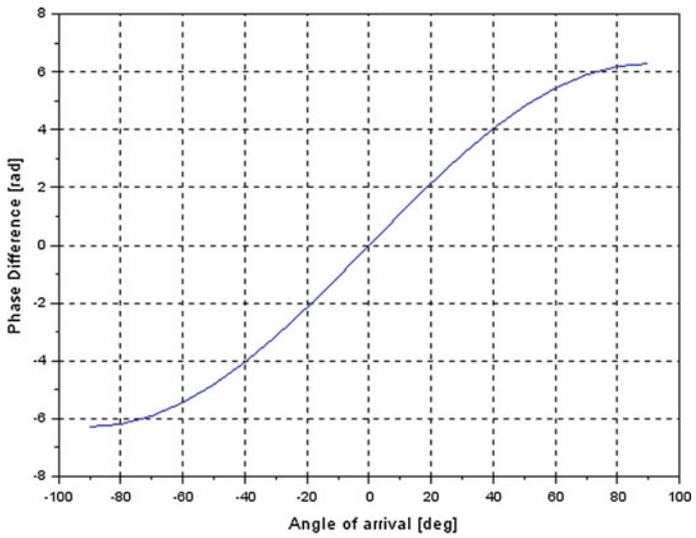


Fig. 1.11 Phase change for $n = 2$ and $d = \lambda$ for θ in the range $[-90, 90]$ degrees

in $|\Delta\phi| \leq \pi$ are indistinguishable from those in the interval $\Delta\phi \pm (2m + 1)\pi$, where m is a nonzero integer. This is referred to as phase wrapping. For the n -array antenna, the so-called phase-wrap angle can be computed as

$$2k\pi(n - 1) \sin(\theta) = \pi \quad (1.12)$$

$$\sin(\theta) = \frac{1}{2k(n - 1)} \quad (1.13)$$

$$\theta = \sin^{-1}\left(\frac{1}{2k(n - 1)}\right) \quad (1.14)$$

Theoretically, one approach to bring the phase into the desired range can be accomplished by addition of or subtraction of 2π , depending on the measured angle. When the measured angle is less than $-\pi$, add 2π to bring the angle into the $[-\pi, \pi]$ range. On the other hand, when the measured angle is greater than $+\pi$, subtract 2π , so the angle falls into the $[-\pi, \pi]$ range.

For automotive radar, phase estimation is important in the measurement of direction of arrival, which will be explained in detail in Chap. 6.

If the target lies in the $[-\pi, \pi]$ interval, then correct detection is possible, but if phase wrapping occurs, false detection of target happens. One solution to this problem is sensor fusion, in which case false angles can be eliminated by comparison of corresponding range obtained for target position based on either the image sensor or LIDAR.

1.6 An Outline of Automotive Radar Applications

With the rapid drive toward driverless or autonomous driving, the number of automotive applications is ever increasing, with focus on safety and comfort. Signal processing can be handily used in many of these applications. Besides radar, other sensors such as the ultrasonic sensor, camera, LIDAR, and GPS can be used to generate the input signals to the signal processor in a sensor fusion setup. Audio, image, and array processing algorithms can be applied to the acquired signals in order to extract information for automatic control of vehicle dynamics or alert the driver of impending dangers around the vehicle. We will mostly give examples related to active safety as it is directly related to recent advances in driving technology such as driverless or autonomous driving.

Automotive radar applications can be classified by the level autonomy [13, 14]. Here, we briefly describe current and future applications based on this classification. The Society of Automotive Engineers (SAE) [13] defines six levels of automation that are seen as the best way to advance autonomous technology.

Level 0 (No Automation): Human driver in control all the time. That means the driver performs steering, braking, and acceleration. There are no systems that automatically assist except for some warnings.

Level 1 (Driver Assistance): Collision mitigation braking that automatically brakes if a collision is imminent is possible. However, the driver is still in control of most of functions.

Level 2 (Partial Automation): Automatic acceleration/deceleration, braking, and steering assistance. It is at this level that some meaningful level of automation begins. However, the driver is required to continuously monitor the vehicle and the traffic during the all the time. Additionally, the driver must be in a position to resume control of the vehicle immediately if necessary.

Level 3 (Conditional Automation): The driving system executes steering and acceleration/deceleration operation including monitoring of the driving environment. At this level, the burden on the driver is reduced so that continuous monitoring is no longer necessary. However, the driver must be able to take control when requested to do so by the system.

Level 4 (High Automation): Extends the autonomous capabilities to be able to handle all driving responsibilities in specific scenarios, even if a human driver fails to respond appropriately to request for intervention. These scenarios are derived from the type of road, the vehicle speed, and the environmental conditions.

Level 5 (Full Automation): Levels 1–4 apply to some driving modes, but Level 5 takes full control of the vehicle for all driving modes. A human driver is not required. The vehicle is capable of completely and independently performing the driving task on all types of roads, at all speeds and under all environmental conditions. This level of autonomy is still under research and development and could take some time to achieve. However, great progress has been made up to Level 4 which brings Level 5 within reachable range.

For each level of autonomy, a radar system with desired characteristics to achieve the expected performance is required. For driver assistance systems at Level 1, the radar sensor mainly performs the task of distance calculation, while the radar acts as integral part of sensors needed to replace the driver at Level 5 automation.

1.7 Challenges for Automotive Radar Developers

Development of the automotive radar sensor faces many challenges from both hardware and software aspects. The software component is dominated by signal processing. Without efficient signal processing algorithms, the utility of the radar sensors becomes very limited.

Starting with hardware aspects, size and weight are key factors that determine whether OEMs will accept the radar sensor as an option for the vehicle model under consideration. Currently, it is not mandatory to incorporate the radar sensors into vehicles and therefore minimization of sensor cost and vehicle integration effort

are of paramount importance. One of the known and most cited advantages of radar over other sensors is the ability to install it behind the bumper fascia. Immediately, size becomes constrained. The beauty of having the sensor behind the bumper fascia is that vehicle aesthetics are not affected by the addition of a safety-related sensor. Although radar sensing obviously enhances safety, it is desirable that it should not change the appearance of the vehicle as this would require extra effort in the design of the vehicle body. Therefore, the radar sensor must fit into the limited space behind the bumper. For passenger cars, the width of the bumper is limited to 40–50 cm. To complicate matters, not the full length of the bumper is usable. Most vehicles target the edges of the bumper near the headlights. The radar designer must aim to make the sensor less than the bumper width. Even though weight does not limit the radar very much, light weight is desirable in order to reduce installation cost. The current trend is to reduce the weight to below 200 g. A topic related to radar dimension is calibration which can be simplified depending on size of radar.

The next challenge is radar waveform design and signal processing. The radar applications are broadly divided into short range, mid-range, and long range [15]. This gives rise to short-range radar (SRR), mid-range radar (MRR), and long-range radar (LRR), respectively. After deciding on the radar application, the next natural step would be to define limits and resolution for range, velocity, and angle. This is a task for waveform design and will ultimately affect the radar antenna design. The waveform design decides the type modulation, the frequency bandwidth, and signal transmission and reception methods. A detailed look at this topic is presented in Chap. 6. During waveform design, an iterative choice of radar waveform, antenna configuration, and signal processing algorithm has to be performed until a combination that meets the target application is decided. This process is very time-consuming since it can only be stopped after key performance indicators are achieved. It also requires intensive collaboration between hardware developers, software developers, and system testers.

Since the target of radar sensing always includes object tracking, choosing a tracking algorithm is a non-trivial task. A balance has to be made between available radar device resources and tracking performance. The options available for this task include simple one-step prediction filters and complex nonlinear filters like the extended Kalman filters and particle filters. These options are treated in detail in Chap. 7.

References

1. Skolnik, M.I.: Radar Handbook, 3rd edn. McGraw-Hill, New York (2008)
2. Skolnik, M.I.: Introduction to Radar Systems, 2nd edn. McGraw-Hill, Inc., New York (1981)
3. Mahafza, B.R.: Radar Systems Analysis and Design Using MATLAB, 2nd edn. Chapman & Hall/CRC (2005)
4. Chen, C.C.: Attenuation of Electromagnetic Radiation by Haze, Fog, Clouds, and Rain. The Rand Corporation (1975)
5. Bartsch, A., Fitzek, F., Rasshofer, R.H.: Pedestrian recognition using automotive radar sensors. *Adv. Radio Sci.* **10**, 45–55 (2012)

6. Heuel, S., Rohling, H.: Pedestrian classification in automotive radar systems.. In: 13th International Radar Symposium, Warsaw, Poland, pp. 39–44 (May 2012)
7. Schubert, E., Meinel, F., Kunert, M., Menzel, W.: Clustering of high resolution automotive radar detections and subsequent feature extraction for classification of road users. In: 16th International Radar Symposium (IRS), Dresden, Germany, pp. 174–179 (June 2015)
8. Grimm, C., Farhoud, R., Fei, T., Warsitz, E., Umbach, R.H.: Detection of moving targets in automotive radar with distorted ego-velocity information. In: IEEE Microwaves, Radar and Remote Sensing Symposium (MRRS), Kiev, Ukraine, pp. 111–116 (Oct 2017)
9. Van Trees, H.: Optimum Array Processing Part. IV, Wiley Interscience (2002)
10. Kraus, J.D., Marhefka, R.J.: Antennas, 3rd edn. McGraw-Hill Education, New York (2001)
11. Rabinovich, V., Alexandrov, N.: Antenna Arrays and Automotive Applications. Springer, NY (2013)
12. Hansen, R.C.: Phased Array Antennas, 2nd edn. Wiley, New York (2009)
13. SAE: Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems. SAE (2014)
14. VDA: From Driver Assistance Systems to Automated Driving. VDA Magazine—Automation (Sept 2015)
15. Schneider, M.: Automotive Radar—Status and Trends. In: Proceedings of the German Microwave Conference (GeMIC) (2005)

Chapter 2

The Radar Equation



2.1 Introduction

This chapter introduces the radar equation which is essential for understanding the effects of propagation on the transmitted radar signal.

2.2 Radar Performance Requirements

In most applications, the radar should be designed to meet specific performance requirements. These requirements include the maximum range, range resolution, maximum velocity, velocity resolution, covered field of view in the angular space, and many other additional demands. The same requirements are also placed on the radar in an automotive applications. In order to meet these system requirements, the fundamental performance characteristics are determined by what is referred to as the radar equation. In this chapter, we take some effort to introduce the radar equation and what it means to the final target application.

2.3 The Radar Equation

As stated in the previous chapter, the antenna acts as the interface between the radar system and the transmission medium through which the electromagnetic waves propagate. The radar equation gives the relationship between the transmitted signal power, received signal power, the range to reflecting object, the characteristic of the reflecting objects, and antenna properties. It can be expressed as follows.

$$P_r = \frac{P_t G_t G_r \lambda^2 \sigma_s}{(4\pi)^3 R^4} = \frac{P_t G_t A_e \sigma_s}{(4\pi)^2 R^4}. \quad (2.1)$$

In Eq. (2.1), P_r denotes the received signal power, and P_t represents the transmitted signal power. The antenna properties are represented by the transmit gain G_t and the receive gain G_r . The parameter $A_e = \frac{G_r \lambda^2}{4\pi}$ corresponds to the effective aperture of the receiving antenna. The characteristics of the reflecting object at range R are represented by σ_s , which is referred to as the radar cross section (RCS). The RCS is a measure of a target's ability to reflect radar signals in the direction of the radar receiver [1] and is very difficult to estimate. The λ represents the transmitted electromagnetic signal wavelength and can be considered constant for any given radar system.

The above radar equation can be expressed in terms of range R as follows.

$$R = \sqrt[4]{\frac{P_t G_t A_e \sigma_s}{P_r (4\pi)^2}} \quad (2.2)$$

It is therefore clear from the above equation that the range is proportional to the fourth root of the transmitted power. By taking into consideration losses associated with the signal reception, the maximum detectable range can be estimated. In order to expand the range that can be covered by the radar system, high receiver sensitivity is required since transmit power cannot be arbitrarily increased. As further simplification of Eq. (2.1), the transmit gain and receive gain can be considered to be equal and constant. Since λ is constant, the dominant variables affecting received power become the transmitted power, the range, and the radar cross section.

Furthermore, if the minimum detectable power by the receiver, S_{\min} , is used to define the radar equation, Eq. (2.2) can be expressed as [1]

$$R = \sqrt[4]{\frac{P_t G_t A_e \sigma_s}{S_{\min} (4\pi)^2}} \quad (2.3)$$

2.4 Effect of Losses

The effect of propagation losses can be incorporated into the radar equation so that it gives a closer resemblance to expectations in practical situations.

The minimum received power can be expressed in terms of SNR as follows:

$$S_{\min} = kT_0 B F_n * \text{SNR}, \quad (2.4)$$

where $kT_0 B$ is referred to as the thermal noise from an ideal ohmic conductor and k is the Boltzmann constant ($1.38064852(79) \times 10^{-23}$ J/K), T_0 is the standard temperature/absolute temperature (290 K), and B is the receiver bandwidth (Hz) or effective noise bandwidth. F_n is the noise figure, and it accounts for nonlinearities introduced by the non-ideal receiver circuitry. The noise figure is a dimensionless value that is defined as the ratio of input to output SNR of the receiver. Additionally, system

losses denoted by L can be incorporated into the radar equation resulting in

$$R = \sqrt[4]{\frac{P_t G_t A_e \sigma_s}{k T_0 B F_n * \text{SNR} (4\pi)^2 L}}. \quad (2.5)$$

Furthermore, factors accounting for pulse integration and effects of propagation of the radar signal can also be added to the radar equation but for our purposes, Eq. (2.5) is sufficient to illustrate the effect of losses on the maximum detectable range. Depending on the radar requirements, the radar equation can also be fine-tuned to be application-specific such as for surveillance radar and radar jammers.

In most applications, the key radar parameters such as antenna gain, wavelength, and noise figure remain unchanged. Considering a target of interest like a vehicle at a fixed distance, we can easily determine the demands placed on the transmit power in order to double the detectable range. Doubling the detectable range gives both the system and/or the driver enough time to react in collision avoidance scenarios. The range is proportional to the fourth root of the transmit power, i.e., $R \propto \sqrt[4]{P_t}$. In order to double the detection range, this would translate to $2R \propto 2\sqrt[4]{P_t} = \sqrt[4]{16P_t}$. Keeping everything else constant, doubling the detectable distance would require increasing the transmit power by 16 times. For the above reason, simply boosting the transmit power places extreme limits system requirements, and hence, innovative radar designs incorporating complex signal processing techniques are used to extend the ranging capabilities of the radar sensor.

2.5 Radar Equation for Automotive Applications

In automotive applications, the radar equation is used to determine the maximum detectable range, and one challenge is the estimation of the RCS [2]. Due to the differences in shape, size, and texture of automotive targets, the RCS cannot be accurately estimated [3].

According to ITU [4], for automotive radars operating in the frequency band 77.5–78 GHz, the detection distance in meters can be expressed as:

$$R = \sqrt[4]{\frac{P_t G_A^2 \lambda^2 \sigma_{\text{tg}}}{S_{\text{min}} (4\pi)^3}} \quad (2.6)$$

where P_t is the transmitter power (W), G_A the antenna gain, σ_{tg} the effective target area, equal to 1 m^2 , λ the wavelength, equal to $3.859 \times 10^{-3} \text{ m}$ at 77.75 GHz, and S_{min} is receiver sensitivity, W. It is assumed that the transmit gain and receive are equal.

By taking attenuation, L_{atm} , of radio waves in the earth atmosphere into account, target detection distance, R_{atm} , will be given by the following equation:

$$R_{\text{atm}} = \sqrt[4]{\frac{P_t G_A^2 \lambda^2 \sigma}{S_{\text{min}} (4\pi)^3 L_{\text{atm}}}}. \quad (2.7)$$

Attenuation of radio waves in the earth atmosphere generally consists of attenuation in atmospheric gases (oxygen and water vapor) and attenuation in fog or rain. The analysis shows that the greatest contribution to the attenuation of radio waves is due to the atmospheric gases and rain. Therefore, to estimate detection distance, attenuation of radio waves in the atmosphere, L_{atm} (dB) needs to be determined. This attenuation is defined by the formula:

$$L_{\text{atm}} = R_{\text{atm}} * (\gamma_g + \gamma_R) \quad (2.8)$$

where γ_g is the specific attenuation due to atmospheric gases, dB/km, γ_R the specific attenuation due to rain, dB/km, and R_{atm} the detection distance considering attenuation of radio waves in the earth's atmosphere.

Atmospheric losses are normally determined by typical atmospheric conditions of pressure ($P_{\text{atm}} = 1013$ hPa), temperature ($T_{\text{atm}} = 15$ °C), water vapor density ($\rho = 7.5$ g/m³), and rain intensity ($R = 5, 10, 15, 20, 25$ and 30 mm/h).

Under these conditions, the specific attenuation due to oxygen at 77.75 GHz is estimated to be 0.088 dB/km and that due to water vapor is 0.286 dB/km. This gives a total specific attenuation ($\gamma_g + \gamma_R$) of 0.374 dB/km. However, in real-operating situations, conditions are far from ideal and deviations should be expected.

Although the above analysis is valid, it is more common to consider the overall system losses L_s which include polarization losses, atmospheric propagation losses, and antenna pattern losses among others [5]. In this case, the maximum range can be expressed as

$$R_{\text{max}} = \sqrt[4]{\frac{P_t G_A^2 \lambda^2 \sigma_s}{S_{\text{min}} (4\pi)^3 L_s}}. \quad (2.9)$$

Alternatively, we can rewrite Eq. (2.9) from Eq. (2.5) as

$$R_{\text{max}} = \sqrt[4]{\frac{P_t G_A^2 \lambda^2 \sigma_s}{k T_0 B F_n * \text{SNR}_{\text{min}} (4\pi)^3 L_s}}. \quad (2.10)$$

If the system bandwidth is defined in terms of coherent processing time T_{cpi} , then we can write Eq. (2.10) as

$$R_{\text{max}} = \sqrt[4]{\frac{P_t G_A^2 \sigma_s \lambda^2 T_{\text{cpi}}}{k T_0 F_n * \text{SNR}_{\text{min}} (4\pi)^3 L_s}}, \quad (2.11)$$

where SNR_{\min} is the minimum SNR corresponding to the minimum receive power. For most automotive applications, the typical radar parameters in the frequency band 76–81 GHz are shown in Table 2.1 [6–8].

As an additional note about the discussion, in practice, it is more convenient to work with decibels than physical units since all operations can be performed by simple addition and subtraction instead of multiplication. For this purpose, Eq. (2.7) can be re-arranged and written as

$$10 * \log_{10}(S_{\min}) = 10 * [(P_t) + \log_{10}(G_A^2) + \log_{10}(\sigma_s) + \log_{10}(\lambda^2) \\ - \log_{10}(L_s) - \log_{10}((4\pi)^3) - \log_{10}(R_{\max}^4)]$$

$$S_{\min}[\text{dB}] = P_t[\text{dB}] + G_A[\text{dB}] + \sigma_s[\text{dB}] + \lambda[\text{dB}] \\ - L_s[\text{dB}] - \text{Cnst}[\text{dB}] - R[\text{dB}]$$

where $G_A[\text{dB}] = 10 * \log_{10}(G_A^2)$ is the gain term, $\lambda[\text{dB}] = 10 * \log_{10}(\lambda^2)$ is the wavelength term, $\text{Cnst}[\text{dB}] = 10 * \log_{10}((4\pi)^3)$, $R[\text{dB}] = 10 * \log_{10}(R_{\max}^4)$ is the range- and wavelength-dependent term. The same can be done for other forms of the radar equation. Figure 2.1 illustrates the range detection principle.

Example Let's consider an automotive radar sensor in the 76–81 GHz band with λ the wavelength equal to 3.859×10^{-3} m, transmit power P_t of 12 dBm, the

Table 2.1 Typical automotive radar parameters

Parameter	Typical range	Notes
Operating range (m)	0–250	Depends on the intended application, e.g., SRR, MRR, and LRR
Transmit power (dBm)	10–13	Depends on regulations.
TX/RX antenna gain (dBi)	10–25	Depends on the azimuth and elevation of the field of view (FOV)
Receiver noise figure (dB)	10–20	Depends on hardware implementation.
Radar cross section (RCS) (dBsm)	(–10)–20	Pedestrians to trucks/buses
Receiver sensitivity (dBm)	(–120)–(–115)	Depends on hardware implementation
Minimum detection SNR (dB)	10–20	Depends on hardware implementation

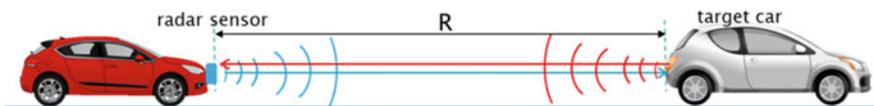


Fig. 2.1 An illustration of range detection by radar. The reflected signal from the target car is used to detect the range R

transmitter/receiver antenna gain of 12 dBi, minimum SNR of -20 dB, system loss of 3 dB, noise Figure F_n of 16 dB, and a bandwidth of 1 GHz. Assuming that the target being tracked is a passenger vehicle with a radar cross section of 10 m^2 , the maximum range without and with losses can be estimated as follows.

Without losses taken into consideration,

$$R[\text{dB}] = P_t[\text{dB}] + G_A[\text{dB}] + \sigma_{\text{tg}}[\text{dB}] + \lambda[\text{dB}] \\ - k\text{TB}[\text{dB}] - \text{Cnst}[\text{dB}] - F_n[\text{dB}] - S_{\text{min}}[\text{dB}]$$

$$R_\lambda[\text{dB}] = (12 - 30)[\text{dB}] + 2 * 12[\text{dB}] + 10[\text{dB}] + (-48.272[\text{dB}]) \\ - (-143.975)[\text{dB}] - 32.976[\text{dB}] - (16[\text{dB}]) - (-20)[\text{dB}] \\ = 82.727[\text{dB}]$$

$$R_{\text{max}}^4 = 10^{82.727/10} = 187,383,347.2$$

$$R_{\text{max}} = 116.999[\text{m}].$$

Taking losses into consideration,

$$R[\text{dB}] = P_t[\text{dB}] + G_A[\text{dB}] + \sigma_{\text{tg}}[\text{dB}] + \lambda[\text{dB}] - L_s[\text{dB}] \\ - k\text{TB}[\text{dB}] - \text{Cnst}[\text{dB}] - F_n[\text{dB}] - S_{\text{min}}[\text{dB}]$$

$$R_\lambda[\text{dB}] = (12 - 30)[\text{dB}] + 2 * 12[\text{dB}] + 10[\text{dB}] + (-48.272[\text{dB}]) - 3[\text{dB}] \\ - (-143.975)[\text{dB}] - 32.976[\text{dB}] - (16[\text{dB}]) - (-20)[\text{dB}] \\ = 79.727[\text{dB}]$$

$$R_{\text{max}}^4 = 10^{79.727/10} = 93,914,141.4$$

$$R_{\text{max}} = 98.443[\text{m}]$$

As expected, the effect of losses is to reduce the detectable maximum range. Radar parameters play a crucial role in the range coverage of the radar sensor, and extra care needs to be taken in their design.

References

1. Skolnik, M.I.: Introduction to Radar Systems, 3rd edn. McGraw-Hill, New York, NY (2001)
2. Rabinovich, V., Alexandrov, N.: Antenna Arrays and Automotive Applications. Springer, New York (2013)

3. Mahafza, B.R.: Radar Systems Analysis and Design Using MATLAB, 2nd edn. Chapman & Hall/CRC (2005)
4. ITU: Systems characteristics and compatibility of automotive radars operating in the frequency band 77.5–78 GHz for sharing studies (2012)
5. Issakov, V.: Microwave Circuits for 24 GHz Automotive Radar in Silicon-Based Technologies. Springer, Berlin, Heidelberg (2010)
6. ITU-R: Systems characteristics of automotive radars operating in the frequency band 76–81 GHz for intelligent transport systems applications (2018)
7. Ramasubramanian, K.: mmWave Radar for Automotive and Industrial Applications, Texas Instruments (2017)
8. CEPT ECC Report 262, Studies related to surveillance radar equipment operating in the 76 to 77 GHz range for fixed transport infrastructure (Jan 2017)

Chapter 3

Signal Processing for Radar Systems



3.1 Introduction

This chapter introduces key ideas behind signal processing of received radar signal. We will mostly be interested in Fourier transforms and their applications to radar signal processing.

3.1.1 Definition of Fourier Transform

The purpose of the Fourier transform is to transform a time-domain signal into the frequency-domain signal. For periodic signals, compact representation of the signal becomes possible. That means a series of time-domain signals can be represented by a single number, i.e., the frequency of the signal. Signal compression for transmission is one popular application of this concept [1, 2].

There are many a good books and online materials that give various levels of details about the Fourier transforms and their mathematical interpretation [3–5]. That kind of analysis is beyond the scope of this chapter, but our intention is to give the necessary background since Fourier transform is a basic tool used in radar signal processing.

The Fourier transform (FT) of a signal $x(t)$ is defined as

$$X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt, \omega \in (-\infty, \infty) \tag{3.1}$$

or

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft} dt, f \in (-\infty, \infty). \tag{3.2}$$

The inverse Fourier transform (IFT) is then given by

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) e^{j\omega t} d\omega, \tag{3.3}$$

or

$$x(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi f t} df. \tag{3.4}$$

The above definitions provide basic relationships required to transform signals from the time domain to the frequency domain and vice versa.

3.1.2 Fourier Transform Properties

This section outlines some of the commonly used Fourier transform properties, and where appropriate, some short comments of the utility are given.

Linearity

Input signal	Fourier transform	Conditions
$ax(t) + by(t)$	$aX(\omega) + bY(\omega)$	a and b are constants

Symmetry

Input signal	Fourier transform	Conditions
$2\pi X(-\omega)$	$\int_{-\infty}^{\infty} X(t) e^{-j\omega t} dt$	$X(\omega)$ exists

Time shift

Input signal	Fourier transform	Conditions
$x(t \pm t_0)$	$e^{\pm j\omega t_0} X(\omega)$	t_0 is real

Time shift results in phase shift in the Fourier domain.

Time and Frequency Scaling

Input signal	Fourier transform	Conditions
$x(at)$	$\frac{1}{ a } X\left(\frac{\omega}{a}\right)$	$X(\omega)$ exists

The scaling property explains what is often referred to as time dilation and frequency compression. When $a < 1$, $x(at)$ is dilated or expanded and $X(\omega/a)$ is compressed. On the other hand, when $a > 1$, $x(at)$ is compressed and $X(\omega/a)$ is expanded.

Central ordinate

Input signal	Fourier transform	Conditions
$X(0)$	$\int_{-\infty}^{\infty} x(t)dt$	
$x(0)$	$\frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega)d\omega$	

Frequency shift

Input signal	Fourier transform	Conditions
$e^{\pm\omega_0 t} \int_{-\infty}^{\infty} x(t)dt$	$X(\omega \mp \omega_0)$	$X(\omega)$ exists

Modulation

Input signal	Fourier transform	Conditions
$x(t)\cos(\omega_0 t)$	$0.5 * [(X(\omega + \omega_0) + (X(\omega - \omega_0))]$	$X(\omega)$ exists
$x(t)\sin(\omega_0 t)$	$0.5 \frac{1}{j} * [(X(\omega - \omega_0) - (X(\omega + \omega_0))]$	$X(\omega)$ exists

Derivatives

Input signal	Fourier transform	Conditions
$\frac{d^n}{dt^n} x(t)$	$(j\omega)^n X(\omega)$	Derivatives exists

Time convolution

Input signal	Fourier transform	Conditions
$\int_{-\infty}^{\infty} x(\tau)h(t - \tau)d\tau$	$X(\omega)H(\omega)$	$X(\omega), H(\omega)$ exists

Frequency convolution

Input signal	Fourier transform	Conditions
$x(t)h(t)$	$\frac{1}{2\pi} \int_{-\infty}^{\infty} X(\tau)H(\omega - \tau)d\tau$	

Autocorrelation

Input signal	Fourier transform	Conditions
$x(\tau)x^*(\tau - t)d\tau$	$X(\omega)X^*(\omega) = X(\omega) ^2$	

Parseval's theorem

Input signal	Fourier transform	Conditions
$\int_{-\infty}^{\infty} x(t) ^2 dt$	$\int_{-\infty}^{\infty} X(\omega) ^2 d\omega$	

Moments

Input signal	Fourier transform	Conditions
$m_n = \int_{-\infty}^{\infty} t^n x(t) dt$	$\frac{d^n}{d\omega^n} X(\omega) _{\omega=0}$	m_n is the n th moment

3.1.3 Fourier Series

Fourier series is the decomposition of a periodic signal into the sum of sinusoidal functions [3]. Mathematically, the decomposition can be performed as follows:

$$s_N(t) = \frac{A_0}{2} + \sum_{n=1}^N A_n \sin\left(\frac{2\pi nt}{T} + \varphi_N\right), \quad (3.5)$$

where T denotes the period, N is the number sinusoids, A_n are Fourier coefficients, and φ_N is an arbitrary phase term.

The complex equivalent of the Fourier series expansion can be given as

$$s_N(t) = \frac{a_0}{2} + \sum_{n=1}^N a_n \cos\left(\frac{2\pi nt}{T}\right) + b_n \sin\left(\frac{2\pi nt}{T}\right) = \sum_{n=-N}^N c_n e^{-j\frac{2\pi nt}{T}}, \quad (3.6)$$

where the Fourier coefficients are given by

$$a_n = \frac{2}{T} \int_{t_0}^{t_0+T} s(t) \cos\left(\frac{2\pi nt}{T}\right) dt$$

$$b_n = \frac{2}{T} \int_{t_0}^{t_0+T} s(t) \sin\left(\frac{2\pi nt}{T}\right) dt$$

$$c_n = \frac{1}{T} \int_{t_0}^{t_0+T} s(t) e^{-j \frac{2\pi n t}{T}} dt \quad (3.7)$$

The Fourier coefficients can then be given computed as

$$A_n = \sqrt{a_n^2 + b_n^2} \quad (3.8)$$

$$\varphi_N = \tan^{-1} \left(\frac{b_n}{a_n} \right) \quad (3.9)$$

From the above equations, the approximation of $s(t)$ by $s_N(t)$ gets better as $N \rightarrow \infty$. One use of the Fourier series expansion is for radar waveform generation where the goal is to increase robustness to interference as described in [6] and [7].

3.1.4 Sampling Theorem

Digital signal processing is commonly applied in radar processing. A prerequisite for this to be possible is the availability of sampled discrete data. The sampling theorem defines the conditions under which the original signal can be faithfully reconstructed from sampled data. Although there are many recent innovative sampling methods in the literature including the use of splines, the Shannon sampling theorem remains the de facto standard. The sampling theorem states that if $x(t)$ is a bandlimited signal with bandwidth B , then it can be reconstructed from samples obtained at a sampling rate greater than $2B$.

In mathematical form, we can write

$$x(t) = \sum_{n=-\infty}^{\infty} x(nT) g(t - nT), \quad g(t) = \sin c(\omega_s t), \quad (3.10)$$

where $\omega_s = 2\pi f_s$ is the radian sampling frequency.

3.1.5 Discrete Fourier Transforms (DFT)

As shown in Sect. 3.1.1, the Fourier transform is defined by continuous integrals over infinite duration. This presents problems in the applications such as radar processing where the computation of continuous integrals is unimaginable and the infinite duration requirement is not feasible. To overcome this limitation, discretization of the

Fourier transform is necessary. This makes it possible to compute Fourier transforms of sample data using discrete Fourier transforms [1, 5, 8–11].

By definition, the discrete Fourier transform (DFT) of signal $x(n)$ is given by,

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi nk}{N}}, \quad k = 0, 1, \dots, N-1, \quad (3.11)$$

where $x(n)$ is the sampled signal at time n , k is the k th frequency in radians per second, and $X(k)$ is the complex-valued discrete Fourier transform at frequency k . $X(k)$ can also be thought of complex spectrum at frequency k .

We can similarly define the inverse discrete Fourier transform (IDFT) as

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)e^{j\frac{2\pi nk}{N}}, \quad n = 0, 1, \dots, N-1. \quad (3.12)$$

The DFT can be computed by simple multiplication and addition operations and can therefore be easily used in most applications using digital signal processing techniques. In practice, the DFT is efficiently computed using FFT algorithms which reduces computational complexity from order N^2 to $N \log N$. As will be seen in Sect. 3.2, the discrete Fourier transform is used in Range–Doppler estimation.

3.1.6 Power Spectrum Estimation

By definition, the power spectrum is the Fourier transform of the autocorrelation sequence of a stationary process [12, 13]. For a signal $x(t)$, the spectrum estimation $S(f)$ can be given by

$$S(f) = \int_{-\infty}^{\infty} R_{xx}(\tau)e^{-j\omega\tau} d\tau, \quad (3.13)$$

where $R_{xx}(\tau)$ denotes the autocorrelation function. The signal whose spectrum is to be estimated is usually embedded in noise. For this reason, high-resolution spectrum estimation techniques are essential. It is not the intention of this section to go into the details of spectrum estimation but to give a very brief insight of the topic since it is important for automotive radar processing. In-depth and excellent presentation of this subject can be found in [13–15].

Traditional methods of spectrum estimation solely rely on the discrete Fourier transform (DFT) to obtain the power spectrum of an observed data sequence. Popular among these are the periodogram and Blackman–Turkey methods [13]. Their major weakness, however, is that they fail to resolve closely spaced spectral peaks when

short-duration data sequences are used. This weakness is the main motivation behind the so-called high-resolution techniques, notably the autoregressive (AR), moving average (ARMA) [16] and eigendecomposition techniques, [17, 18]. Although these high-resolution techniques give better-resolved spectrum estimates, it is only at high signal-to-noise ratios (HSNRs) that their performance can be guaranteed to be superior to the traditional methods. The reason for this could be attributed to the inability to accurately separate the signal subspace from the noise subspace at low signal-to-noise ratios (LSNRs). The presence of noise is a serious nonlinearity problem in AR parameter estimation. For further details on this topic, refer to [12].

Although the power spectrum is defined in terms of the autocorrelation function, in practice, it is estimated using the DFT. The DFT can be efficiently computed by the FFT and hence its widespread use. In that case, the power spectrum is the magnitude of the FFT result.

For radar processing, spectrum estimation is almost always used to estimate the range and velocity of the target. However, angular position of the target is obtained by high-resolution techniques since it is usually desirable to separate closely spaced targets.

3.1.7 Windowing Techniques

Windows are required in order to improve the spectrum estimation results. The reason is that the periodic assumption used in DFT requires that estimated frequencies are integer multiples of frequency resolution. This cannot be guaranteed, and the result is discontinuities in the spectrum which in turn results in spreading of energy into multiple frequency bins which manifest as sidelobes. This problem can be improved by using window functions. Window functions start near or at zero, then increase to a maximum at the center of the sampled data sequence, and then decrease again [14]. From the convolution principle, windowing has the smoothing effect of the frequency response of the signal. This leads to the reduction in sidelobes when compared to the spectrum without windowing. However, windowing also results in the undesirable widening of the main lobe, which means that a trade-off has to be made.

Window functions have been in use for a long time [19–21], and below we give some of the windows commonly used in automotive radar signal processing.

3.1.7.1 Rectangular Window

The window function is defined by the following expression:

$$w(n) = 1. \quad (3.14)$$

The window has the effect of multiplying the data sequence by a rectangular function of unit amplitude and is thus equivalent to just using the observed sequence.

3.1.7.2 Parzen Window (de La Vallee Poussin Window)

The window function is B-spline function defined by the following expression.

$$w(n) = \begin{cases} 1 - 6\left(\frac{2n}{N}\right)^2\left(1 - \frac{2|n|}{N}\right), & 0 \leq |n| \leq \frac{N}{4} \\ 2\left(1 - \frac{2|n|}{N}\right)^3, & \frac{N}{4} < |n| \leq \frac{N}{2} \end{cases} \quad (3.15)$$

3.1.7.3 Triangular Window (Bartlett Window)

The window function is defined by the following expression.

$$w(n) = 1 - \left| \frac{n - \frac{N-1}{2}}{L/2} \right| \quad (3.16)$$

or

$$w(n) = \begin{cases} \left(\frac{2n}{N}\right), & 0 \leq n \leq \frac{N}{2} \\ \left(2 - \frac{2n}{N}\right)^1, & \frac{N}{2} < |n| \leq N \end{cases} \quad (3.17)$$

3.1.7.4 Welch Window (Parabolic Window)

The window function is a quadratic function defined by the following expression.

$$w(n) = 1 - \left(\frac{n - \frac{N-1}{2}}{\frac{N-1}{2}} \right)^2 \quad (3.18)$$

3.1.7.5 Hanning Window

The Hann window is one of the popular windows in automotive radar signal processing. The window function is defined by the following expression.

$$w(n) = 0.5\left(1 - \cos\left(\frac{2\pi n}{N-1}\right)\right) \quad (3.19)$$

3.1.7.6 Hamming Window

The window function is defined by the following expression.

$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right) \quad (3.20)$$

3.1.7.7 Blackman Window

The window function is defined by the following expression.

$$w(n) = a_0 - a_1 \cos\left(\frac{2\pi n}{N-1}\right) + a_2 \cos\left(\frac{4\pi n}{N-1}\right) \quad (3.21)$$

where $a_0 = 0.42$, $a_1 = 0.5$, and $a_2 = 0.08$.

3.1.7.8 Nuttall Window

The window function is defined by the following expression.

$$w(n) = a_0 - a_1 \cos\left(\frac{2\pi n}{N-1}\right) + a_2 \cos\left(\frac{4\pi n}{N-1}\right) - a_3 \cos\left(\frac{6\pi n}{N-1}\right) \quad (3.22)$$

where $a_0 = 0.355768$, $a_1 = 0.487396$, $a_2 = 0.144232$, and $a_3 = 0.012604$.

3.1.7.9 Blackman–Nuttall Window

The window function is defined by the following expression.

$$w(n) = a_0 - a_1 \cos\left(\frac{2\pi n}{N-1}\right) + a_2 \cos\left(\frac{4\pi n}{N-1}\right) - a_3 \cos\left(\frac{6\pi n}{N-1}\right) \quad (3.23)$$

where $a_0 = 0.3635819$, $a_1 = 0.4891775$, $a_2 = 0.1365995$, and $a_3 = 0.0106411$.

3.1.7.10 Blackman–Harris Window

The window function is defined by the following expression.

$$w(n) = a_0 - a_1 \cos\left(\frac{2\pi n}{N-1}\right) + a_2 \cos\left(\frac{4\pi n}{N-1}\right) - a_3 \cos\left(\frac{6\pi n}{N-1}\right) \quad (3.24)$$

where $a_0 = 0.35875$, $a_1 = 0.48829$, $a_2 = 0.14128$, and $a_3 = 0.01168$.

3.1.7.11 Flat-Top Window

The window function is defined by the following expression.

$$w(n) = a_0 - a_1 \cos\left(\frac{2\pi n}{N-1}\right) + a_2 \cos\left(\frac{4\pi n}{N-1}\right) - a_3 \cos\left(\frac{6\pi n}{N-1}\right) + a_4 \cos\left(\frac{8\pi n}{N-1}\right) \quad (3.25)$$

where $a_0 = 1$, $a_1 = 1.93$, $a_2 = 1.29$, $a_3 = 0.388$, and $a_4 = 0.028$.

3.1.7.12 Turkey Window

$$w(n) = \begin{cases} 0.5 + 0.5 \cos\left(\pi \left(\frac{2n}{a(N-1)} - 1\right)\right), & 0 \leq n < \frac{a(N-1)}{2} \\ 1, & \frac{a(N-1)}{2} \leq n \leq (N-1)\left(1 - \frac{a}{2}\right) \\ 0.5 + 0.5 \cos\left(\pi \left(\frac{2n}{a(N-1)} - \frac{2}{a} + 1\right)\right), & (N-1)\left(1 - \frac{a}{2}\right) < n \leq (N-1) \end{cases} \quad (3.26)$$

Setting $a = 0$ yields the rectangular window, while $a = 1$ results in the Hanning window.

3.1.7.13 Kaiser Window

The window function is defined by the following expression.

$$w(n) = \frac{I_0\left(\pi\alpha\sqrt{1 - \left(\frac{2n}{N-1}\right)^2}\right)}{I_0(\pi\alpha)}, \quad (3.27)$$

where I_0 is the zeroth-order modified Bessel function of the first kind. The trade-off between the main lobe width and sidelobe level is determined by the tuning parameter α .

3.1.7.14 Dolph–Chebyshev Window

The window function is defined by the following expressions.

Table 3.1 Properties of window functions

Window function	Section	Highest sidelobe level (dB)	−3 dB mainlobe width (bins)	−6 dB mainlobe width (bins)	Sidelobe roll-off rate (dB/oct)
Rectangular	3.1.7.1	−13	0.88	1.21	−6
Parzen	3.1.7.2	−53	1.82	2.66	−24
Triangular (Bartlett)	3.1.7.3	−27	1.27	1.78	−12
Welch (Parabolic)	3.1.7.4	−21	1.15	1.59	−12
Hanning	3.1.7.5	−32	1.44	2.00	−18
Hamming	3.1.7.6	−43	1.30	1.81	−6
Blackman	3.1.7.7	−58	1.64	2.30	−18
Nuttall	3.1.7.8	−93	1.91	2.68	18
Blackman–Nuttall	3.1.7.9	−98	1.98	2.62	−6
Blackman–Harris	3.1.7.10	−71	1.62	2.27	−6
Flat-top	3.1.7.11	−44	2.94	3.56	−6
Turkey ($a = 0.5$)	3.1.7.12	−15	1.15	1.57	−18
Kaiser ($\alpha = 3.5$)	3.1.7.13	−82	1.83	2.57	−6
Dolph–Chebyshev ($\alpha = 4.0$)	3.1.7.14	−80	1.65	2.31	0

$$\begin{aligned}
W_0(k) &= \frac{\cos(N \cos^{-1}(\beta \cos(\frac{\pi k}{N})))}{\cos h(N \cos h^{-1}(\beta))} \\
\beta &= \cos h\left(\frac{1}{N} \cos h^{-1}(10^\alpha)\right) \\
\cos h^{-1}(x) &= \ln\left(x + \sqrt{x^2 - 1.0}\right) \\
w_0(n) &= \frac{1}{N} \sum_{k=0}^{N-1} W_0(k) e^{j2\pi kn/N}, \quad -\frac{N}{2} \leq n \leq \frac{N}{2}
\end{aligned} \tag{3.28}$$

As can be observed from Eq. (3.28), the window function is computed by inverse DFT. The properties of the window functions described above are summarized in Table 3.1.

3.2 Multi-dimensional Fourier Transforms (Basis for Range Doppler Estimation)

The Fourier transform theory can be extended to multi-dimensional signals [18]. For radar signal processing, we are mostly interested in two-dimensional (2D) and three-dimensional (3D) discrete extensions. 2D DFT is used to compute Range–Doppler

profiles, which provide the basic functions of the radar. Additionally, 3D DFT can be used for direction-of-arrival DOA estimation. In this section, we give a brief description of 2D DFT as it is the key to understanding higher dimension Fourier transforms. It is worth mentioning that in most applications, the DFT is efficiently performed using the fast Fourier transform (FFT) algorithm.

Imagine that we have two a two-dimensional signal $f(m, n)$ defined on the discrete grid of size $M \times N$. The 2D DFT is defined by

$$F(k, l) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) e^{-j2\pi(\frac{k}{M}m + \frac{l}{N}n)},$$

$$k = 0, \dots, M-1, \quad l = 0, \dots, N-1. \quad (3.29)$$

The inverse transform is given by

$$f(m, n) = \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} F(k, l) e^{j2\pi(\frac{k}{M}m + \frac{l}{N}n)}, \quad m = 0, \dots, M-1, \quad n = 0, \dots, N-1. \quad (3.30)$$

The following table summarizes some of the important properties of the 2D DFT.

Property	Expression	Conditions
Periodicity	$F(k, l) = F(k + aM, l) = F(k, l + bN) = F(k + aM, l + bN)$;	a and b are integers
Conjugate symmetry	$F(k, l) = F^*(-k, -l)$, $ F(k, l) = F^*(-k, -l) $	Input signal $f(m, n)$ is real
Power spectrum	$P(k, l) = F(k, l) ^2$	
Mean value (D.C)	$\bar{f}(m, n) = F(0, 0)$	
Convolution theorem	$f(m, n) * h(m, n) \xrightarrow{FT} F(k, l) \times H(k, l)$, FT denotes Fourier transform, '*' denotes convolution operation.	
Scaling	$f(am, bn) \xrightarrow{FT} \frac{1}{ ab } F(\frac{k}{a}, \frac{l}{b})$	a and b are constants
Translation	$f(m - m_0, n - n_0) \xrightarrow{FT} F(k, l) e^{-j2\pi(\frac{k}{M}m_0 + \frac{l}{N}n_0)}$ $f(m, n) e^{j2\pi(\frac{m}{M}k_0 + \frac{n}{N}l_0)} \xrightarrow{FT} f(k - k_0, l - l_0)$	

3.3 Noise and Clutter Reduction in Radar Signals

In real situations, the signal received by the radar sensor is always embedded in noise and clutter. While noise can be characterized by some assumed statistical distribution, clutter is more difficult to characterize since it is the result of reflections from undesirable obstacles which behave like targets. For automotive radar, the sources of clutter include rain, ground reflections, and roadside structures.

In fact, the reflections within the radar field of view that are not the target of interest can be considered as clutter. The level of interference from clutter can be characterized by the signal-to-clutter ratio, which depends on the RCS, clutter reflectivity, and the degree of exposure of the clutter, among other factors [22]. The s/c ratio can be incorporated into the radar equation although this presents challenges due to the dynamic nature of clutter. In some situations, when the distribution of the clutter signal is Gaussian, it is better to use the combination of noise and clutter as total interference in the received signal such that instead of SNR, we now have signal-to-clutter-plus-noise ratio (SCNR). This then leads to signal-to-interference ratio (SIR) expressed as

$$\text{SIR} = \frac{1}{\frac{1}{\text{SCR}} + \frac{1}{\text{SNR}}}, \quad (3.31)$$

where SCR is the signal-to-clutter ratio. The modified radar equation then becomes

$$R = \sqrt[4]{\frac{P_t G_t A_e \sigma_s}{k T_0 B F_n * \text{SIR} (4\pi)^2 L}}. \quad (3.32)$$

Some strategies to reduce the effect of clutter include multi-target indication (MTI), pulse Doppler processing, and polarization techniques. Using narrow beam width radar antenna and increasing range resolution by widening the bandwidth can be effective for certain types of clutter. The downside of these additional measures is increased radar system complexity, cost, and size. Therefore, it may be better to improve signal processing techniques in order to suppress clutter to desirable levels.

References

1. Sayood, K.: Introduction to Data Compression, 3rd edn, Elsevier Inc. (2006)
2. Gonzalez, R.C., Woods, R.E.: Digital Image Processing, 3rd edn, Pearson (2007)
3. Brown, J.W, Churchill, R.V.: Fourier Series and Boundary Value Problems, 8th edn, McGraw-Hill Education (2011)
4. Smith, J.O.: Mathematics of the Discrete Fourier Transform (DFT): With Audio Applications, 2nd edn, W3K Publishing (2007)
5. Bracewell, R.N.: The Fourier Transform and Its Applications, McGraw-Hill, New York (1999)
6. Gill, G.S.: Ultra-wideband radar using Fourier synthesized waveforms, IEEE Trans. Electro-magn. Compat. **39**(2), 124–131 (1997)

7. Taylor, J.D.: *Ultra-wideband Radar Technology*, CRC Press (2000)
8. Richards, M.A.: *Fundamentals of Radar Signal Processing*, 2nd Ed, McGraw-Hill, (2014)
9. Brandwood, D.: *Fourier Transforms in Radar and Signal Processing*, Artech House (2003)
10. Wang, B.: *Digital Signal Processing Techniques and Applications in Radar Image Processing*, John Wiley & Sons, Inc. (2008)
11. Melvin, W.L., Scheer J.A. (eds.): *Principles of Modern Radar: Radar Applications*, SciTech Publishing (2014)
12. Gamba, J.: *On Noise-Compensated Techniques for Time Delay Estimation*, Ph.D. Thesis (2005)
13. Kay, S.M.: *Modern Spectral Estimation: Theory and Application*. Prentice Hall, Englewood Cliffs, NJ (1988)
14. Hayes, M.H.: *Statistical Digital Signal Processing and Modeling*. Wiley, New York (1996)
15. Therrien, C.W.: *Discrete Random Signals and Statistical Signal Processing*. Prentice Hall, Englewood Cliffs, NJ (1992)
16. Gamba, J., Shimamura, T., Kawasaki, S., Higuchi, M., Murakami, H.: A joint iterative estimation of noise variance and AR parameters. *Int. J. Inf. Sci. Comput. Eng. (IJISCE)* **2**(1), 1–6 (2011)
17. Vaseghi, S.V.: *Advanced Signal Processing and Digital Noise Reduction*. John Wiley and Teubner, England (1996)
18. Gamba, J., Shimamura, T.: Two-dimensional spectral estimation with noise-compensated data extrapolation. In: *Proceedings of International Conference on Fundamentals of Electronics, Communications and Computer Science*, pp. 20–17–20–22. Tokyo, Japan, Mar (2002)
19. Harris, F.J.: On the use of windows for harmonic analysis with the discrete fourier transform. *Proc. IEEE.* **66**(1), 51–83 (1978)
20. Nuttall, A.H.: *Some Windows With Very Good Sidelobe Behavior; Application to Discrete Hilbert Transform*, NUSC Technical Report 6239, Naval Underwater Systems Center, Surface Ship Sonar Systems Department, 9 Apr 1980
21. Heinzel, G., Rüdiger, A., Schilling, R.: *Spectrum and spectral density estimation by the Discrete Fourier transform (DFT)*, including a comprehensive list of window functions and some new flat-top windows, Max Planck Institute for Gravitational Physics (Albert Einstein Institute), Branch Institute Hannover, 15 Feb 2002
22. Curry, G.R.: *Radar Essentials: A Concise Handbook for Radar Design and Performance Analysis*, Spi edn, Scitech Publishing (2011)

Chapter 4

Radar Waveforms and Their Mathematical Models



4.1 Introduction

As introduced in Chap. 1, radar covers a wide range of frequencies in the RF band. This chapter deals with various radar waveforms derived from those frequencies that are used in target detection and localization.

4.2 Waveform Types Overview

Before constructing a radar system, critical decisions about the system properties have to be made. One of these properties is the radar waveform to be used. The waveform not only determines what type of algorithm used for processing the received signal but also has an impact on the cost and complexity of the system's hardware elements.

In general, radar waveform's taxonomy classifies them into continuous waves or pulsed waves. Continuous waves normally require separate receive and transmit antennas. Isolation requirements limit the transmit power but range estimation is generally good. On the other hand, for pulsed signals, the same antenna can be used for both transmission and reception, with some relaxation on power limits but blind ranges could result from radar measurements. These kinds of trade-offs have to be considered when selecting the waveform.

On the second level, modulation techniques further separate the type of final waveforms used. As in communications systems, amplitude, phase, and frequency modulation can be applied to the waveforms. Additionally, the choice of polarization can also have a significant impact on the processing of the received signal.

In the following sections, we introduce the most commonly used radar waveforms and their properties.

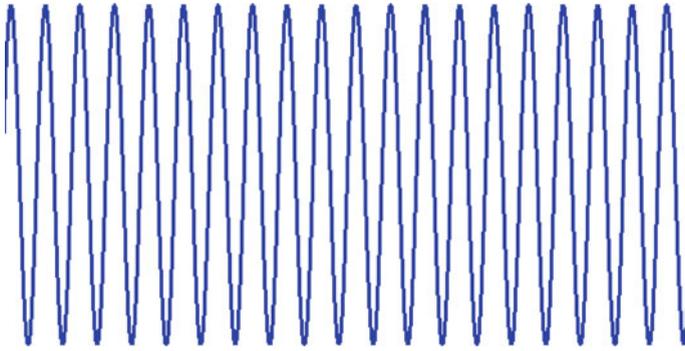


Fig. 4.1 An example of the CW for velocity detection

4.3 Continuous Waveform (CW)

The CW, illustrated in Fig. 4.1, has good Doppler resolution but cannot resolve range. The frequency resolution is given by $\Delta f = \frac{1}{T_{cw}}$, where T_{cw} is the period of the continuous wave.

4.4 Pulse Doppler Radar (PDR) Waveform

With pulse Doppler radar, good range resolution and Doppler resolution are possible. They are, respectively, given by $\Delta f = \frac{1}{N_p * T_p}$ and $\Delta R = c * \frac{T_p}{2}$, where N_p is the number of pulses, T_p is the pulse width, and c is the speed of light. Figure 4.2 is an example of the PDR waveform.

4.5 Frequency-Modulated CW (FMCW) and Their Variations

FMCW is one of the most commonly used types of waveform. This is because it is possible to estimate both range and velocity at reduced device cost [1, 2]. Although pulse Doppler radar can achieve the same functionality, it is dogged by high design and device costs.

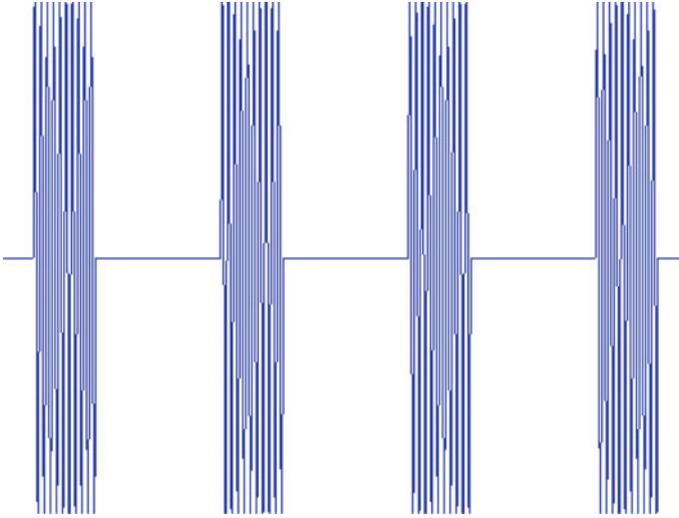


Fig. 4.2 An example of the pulse Doppler radar (PDR) waveform for both range and velocity detection

4.5.1 Linear Frequency-Modulated CW (LFMCW)

Due to the target range and Doppler frequency shift, a beat frequency is generated. The components of the beat frequency are given by the following expressions.

$$f_b = \frac{B}{T_s} * \frac{2R}{c} \quad (4.1)$$

$$f_D = \frac{2v_r}{\lambda} \quad (4.2)$$

These are superimposed as difference frequencies from the up and down chirps, f_{bu} and f_{bd} , respectively, are shown in Fig. 4.3.

$$f_{bu} = f_b - f_d \quad (4.3)$$

$$f_{bd} = f_b + f_d \quad (4.4)$$

From Eqs. (4.3) and (4.4), the target range and radial velocity can be estimated as

$$R = \frac{cT_s}{4B} * (f_{bd} + f_{bu}) \quad (4.5)$$

$$v_r = \frac{\lambda}{4} (f_{bd} - f_{bu}) \quad (4.6)$$

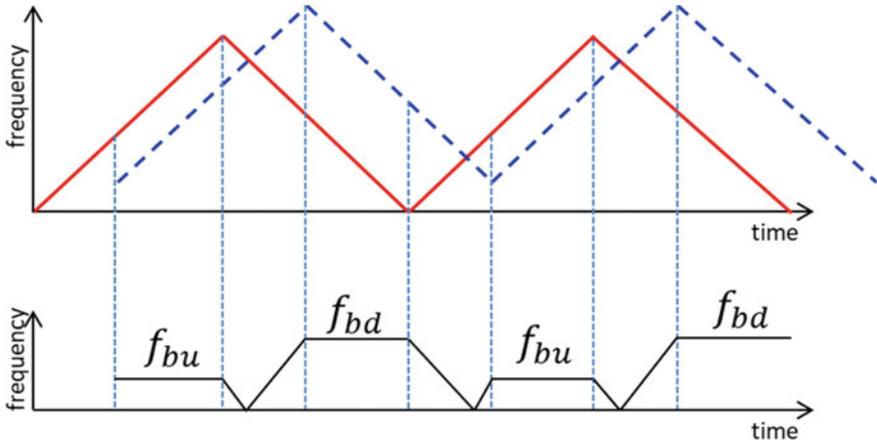


Fig. 4.3 An example of the LFM CW for range and velocity detection

FMCW: Ghost targets

For single-target scenarios, the target range and velocity is normally extracted from the intersection of the range-velocity profiles obtained from the up and down chirps, Fig. 4.4a. For multiple targets, the multiple intersections of range-velocity profiles give rise to ghost targets as shown in Fig. 4.4b.

The presence of ghost target is a serious problem in automotive radar as it can result in wrong range and velocity, which in turn increases the risk of position judgment. Wrong position of judgment can lead to a fatal crash. This ghost target problem is one reason why alternatives to FMCW radar are sought for automotive applications.

4.5.2 Stepped FMCW

A stepped FMCW radar system transmits a sequence of sinusoids at different frequencies and measures the steady-state amplitude and phase shift induced by the radar channel at each discrete frequency. It was originally motivated by the resulting simplified signal processing techniques since fixed stable frequencies could be used [3, 4]. Specifically, the target range profile can be computed using inverse discrete Fourier transformation (IDFT), which can be achieved by high-speed FFT algorithms. Figure 4.5 gives an image of how the stepped FMCW scheme works.

The phase of the reflected signals from a target at range R is given by

$$\varphi = 2\pi f_c * \frac{2R}{c}. \quad (4.7)$$

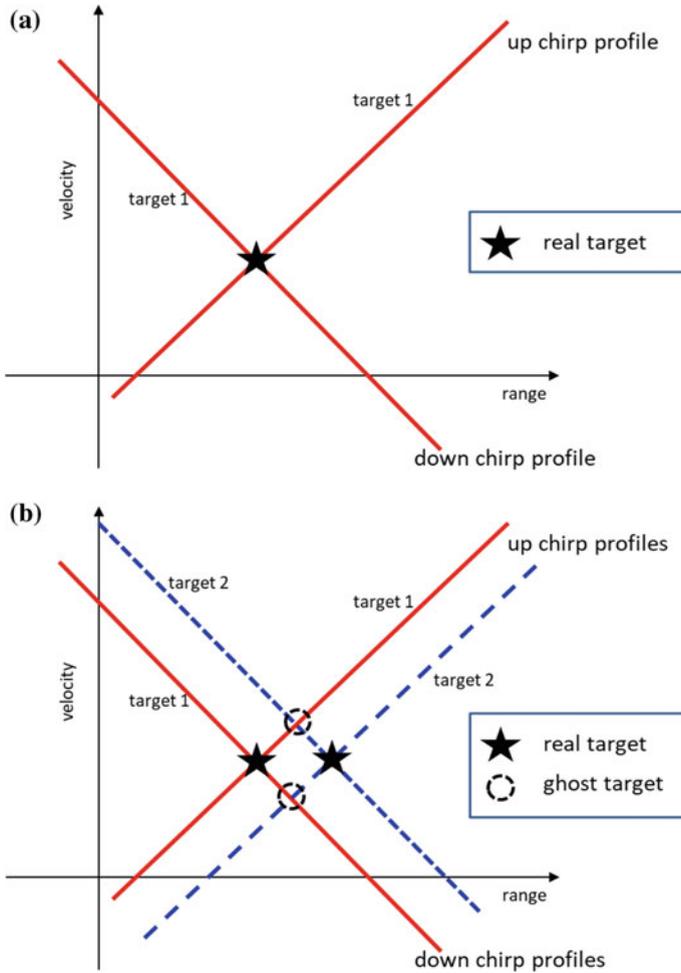


Fig. 4.4 **a** An example of single-target range-velocity profile from LFM CW. Target range and velocity can be extracted unambiguously. **b** An example of multi-target range-velocity profile from LFM CW. Ghost targets result from multiple intersection points between different targets

From Eq. (4.7), the range can be extracted as

$$R = \frac{c\varphi}{4\pi f_c}. \tag{4.8}$$

However, at high frequencies, the maximum unambiguous range is too small to be useful, especially for automotive application. As an example, taking the maximum possible phase of 2π the corresponding maximum range at a center frequency of 77 GHz becomes

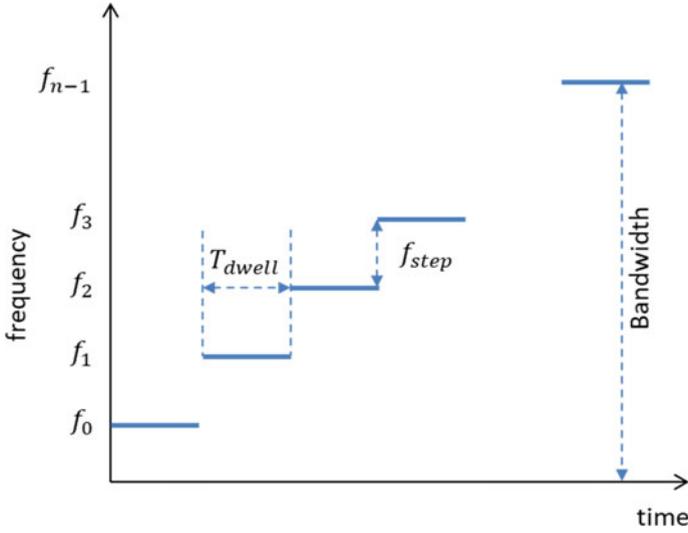


Fig. 4.5 An example of the stepped FMCW for range and velocity detection

$$R_{\max} = \frac{c\varphi}{4\pi f_c} = \frac{2\pi * 3 * 10^8}{4\pi * 77 * 10^9} = \frac{3}{77 * 2 * 10} = 0.195 \text{ cm.} \quad (4.9)$$

This is obviously of no practical use for many applications. However, if the two center frequencies f_1 and f_2 are used, then for a target at range R , the resulting phases from the two frequencies are given by

$$\varphi_1 = 2\pi f_1 * \frac{2R}{c} \quad (4.10)$$

$$\varphi_2 = 2\pi f_2 * \frac{2R}{c}. \quad (4.11)$$

The phase difference $\Delta\varphi$ becomes

$$\Delta\varphi = \varphi_2 - \varphi_1 = \frac{4\pi R}{c}(f_2 - f_1) = \frac{4\pi R}{c}\Delta f \quad (4.12)$$

where the $\Delta f = f_2 - f_1$ is the step frequency. From the above equation, Eq. (4.12), the range can be estimated as

$$R = \frac{c\Delta\varphi}{4\pi\Delta f} \quad (4.13)$$

For the maximum possible phase difference of 2π then a maximum range of

$$R_{\max} = \frac{2\pi c}{4\pi \Delta f} = \frac{c}{2\Delta f}. \tag{4.14}$$

For step frequency of say 10 MHz, then the maximum range becomes

$$R_{\max} = \frac{3 \times 10^8}{2 \times 10 \times 10^6} = 15 \text{ m}. \tag{4.15}$$

By choosing an appropriate value of step frequency, a target maximum range can be set.

This is the basic principle of the stepped FMCW waveform approach to range estimation.

4.5.3 Multi-frequency Shift Keying (MFSK)

The MFSK waveform offers the possibility of simultaneously measuring unambiguous range and velocity [5, 6]. The concept is illustrated in Fig. 4.6. Two intertwined and shifted frequencies *A* and *B* are used. Based on the phase difference of the range spectra measured from f_A and f_B and the step frequency, the range can be estimated as *R*,

$$R = \frac{-c}{4\pi f_{\text{step}}}. \tag{4.16}$$

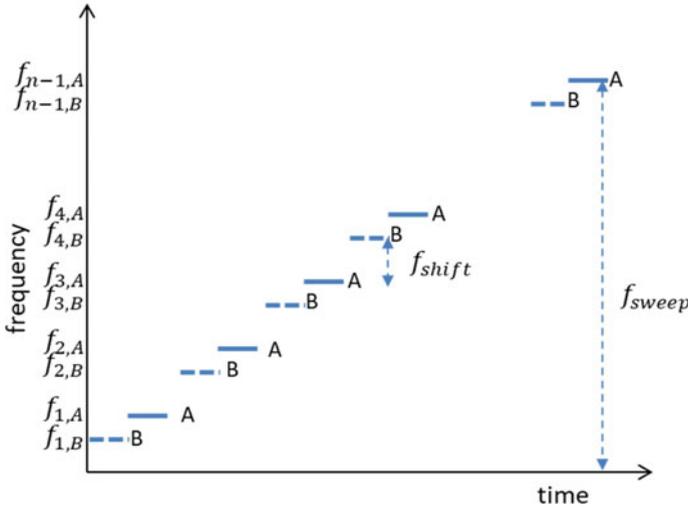


Fig. 4.6 An example of the MFSK waveform for range and velocity detection. Two frequency steps *A* and *B* are employed

where $\Delta\varphi = f_B - f_A$ is the phase difference.

From the FFT spectrum, it is expected that the peak will be detected at the same bin denoted by N_{peak} for both frequencies and the following expression defines ambiguities in range and velocity [6].

$$N_{\text{peak}} = \frac{v}{\Delta v} - \frac{R}{\Delta R}, \quad (4.17)$$

where Δv is the velocity resolution and ΔR the range resolution. The phase difference is given by

$$\Delta\varphi = \frac{v}{\Delta v} * \left(\frac{\pi}{N-1} \right) - \frac{2f_{\text{shift}}}{c} * 2\pi R, \quad (4.18)$$

where N is the number of FFT points used.

Substituting the velocity expression into the phase Eq. (4.18), the unambiguous range can be obtained as

$$R_{\text{unamb}} = \frac{c\Delta R (N-1) * \Delta\varphi - N_{\text{peak}} * \pi}{\pi (c - 4(N-1)\Delta R f_{\text{shift}})}. \quad (4.19)$$

Similarly, the unambiguous velocity can be computed as

$$v_{\text{unamb}} = \frac{(N-1)\Delta v (c\Delta\varphi - 4\pi\Delta R f_{\text{shift}}N_{\text{peak}})}{\pi (c - 4(N-1)\Delta R f_{\text{shift}})}. \quad (4.20)$$

From the above expression (4.20), it can be observed that the availability of phase shift information from frequency spectrum is sufficient to estimate range and velocity unambiguously.

4.5.4 Interrupted FMCW (FMICW)

The FMICW addresses the problem of isolation between the transmitter and receiver. This is achieved by only enabling reception when switching signal is off as shown in Fig. 4.7. Reception is only allowed when the timing signal is off and the received waveform is shown by the shaded portions.

As can be seen in Fig. 4.7, for close targets, the total reception time reduced significantly, making it difficult to detect close range targets. For long-range targets, the effect is opposite. Therefore, a compromise has to be made between short-range and long-range targets [7, 8].

When the round-trip delay from the target is a multiple of the switching period, the received signal power is zero, resulting in blind ranges. To avoid the phenomenon, the switching frequency f_s should be chosen such that

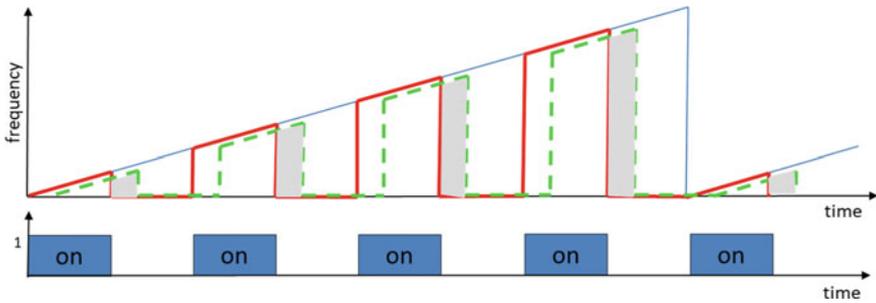


Fig. 4.7 An example of the FMCW for range and velocity detection. The blue waveform is the original uninterrupted waveform. The red is the transmitted waveform while the green-dashed is the reflected waveform

$$f_s = \frac{c}{4R_{\max}}. \quad (4.21)$$

The blind ranges R_B occur at

$$R_B = \frac{c * k}{2f_s}, \quad k = 0, 1, 2, \dots \quad (4.22)$$

The FMCW finds use in automatic cruise control (ACC) radar. This is because these radars traditionally use FMCW waveform and it is important to reduce near range clutter and maximize detection at long range.

4.6 Fast Chirp Ramp Sequence Waveform

Although the FMCW is a popularly used waveform for automotive radar and other applications, its main drawback is that in multi-target environments, it is necessary to match velocity and range values for each target. This is referred to as pairing. Incorrect matching could result in mispairing which leads to wrong target position or velocity estimation. To avoid this problem, the fast chirp ramp sequence shown in Fig. 4.8 makes it possible to estimate both target range and velocity without pairing.

The process of obtaining range and velocity utilizes 2-DFT which is performed first for each individual chirp/ramp to obtain range information and second across ramps to obtain velocity information. By further computing angle information based on the antenna configuration used to receive the chirp ramp sequence, then 3D target data can be constructed and consists of range, velocity, and angle. For angle estimation, high-resolution techniques like Capon, MUSIC, and ESPRIT can be used. An illustration of this concept is shown in Fig. 4.9. Taking the DFT along the chirp samples gives range samples for each chirp while taking FFT along the DFT gives velocity samples. Additionally, by independently computing the angle, using say

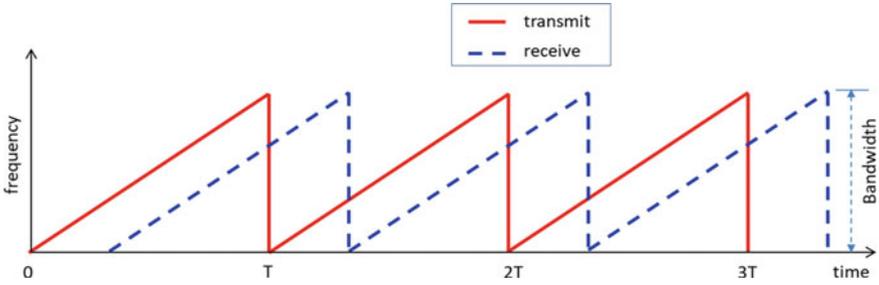


Fig. 4.8 An example of the fast chirp ramp sequence waveform for range and velocity detection

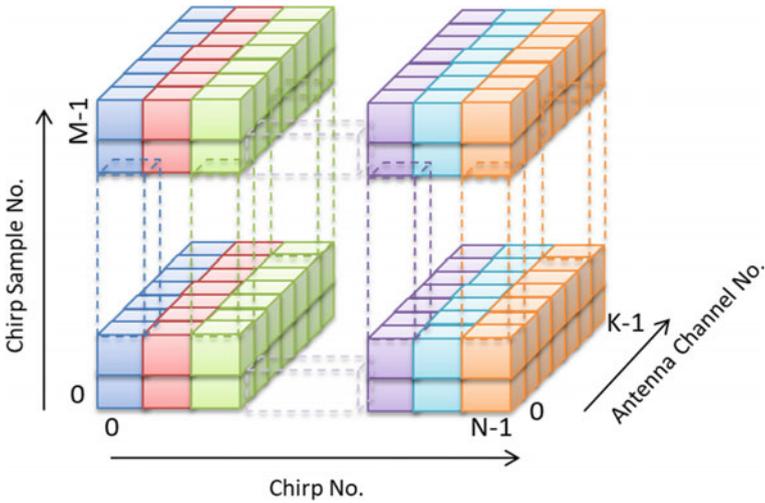


Fig. 4.9 An illustration of the 3D data cube concept from chirp ramp sequence waveform

MUSIC, results in direction of arrival estimates. Finally, the result is a similar 3D data cube consisting of range, velocity, and angle of arrival for each detected target.

If the fast chirp ramp sequence is swept from a carrier frequency f_c , then frequency at any given time instant $f(t)$ can be expressed as

$$f(t) = f_c + \frac{B}{T}t = f_c + \alpha t, \tag{4.23}$$

where B is the sweep bandwidth, T the sweep duration, and $\alpha = \frac{B}{T}$ denotes the chirp rate as illustrated in Fig. 4.8. The corresponding instantaneous phase can be obtained from the relation:

$$\omega(t) = \frac{d\varphi(t)}{dt} = 2\pi f(t). \tag{4.24}$$

From which the

$$\varphi(t) = \int_0^t 2\pi f(t) dt = 2\pi \left(f_c t + \frac{\alpha}{2} t^2 \right) + \varphi_0, \quad (4.25)$$

where φ_0 denotes the initial phase.

Transmitted sinusoidal signal can now be expressed as

$$s(t) = A \cos \left(2\pi \left(f_c t + \frac{\alpha}{2} t^2 \right) + \varphi_0 \right). \quad (4.26)$$

Without taking the initial phase into consideration, a general expression for the transmitted m th chirp ramp is given by the following expressions

$$s(t) = A \cos \left(2\pi f_c t + \frac{\pi \alpha}{2} (t - mT)^2 \right). \quad (4.27)$$

For a target at a range R , and with radial velocity v away from the radar, the round-trip delay τ of the reflected chirp signal is given by

$$\tau = \frac{2(R + vt)}{c}. \quad (4.28)$$

The received $r(t)$ delayed chirp signal becomes

$$r(t) = B \cos \left(2\pi f_c (t - \tau) + \frac{\pi \alpha}{2} (t - \tau - mT)^2 \right). \quad (4.29)$$

Assuming normalized amplitudes A for both the transmitted and received chirp signals, we get

$$s(t) = \cos \left(2\pi f_c t + \frac{\pi \alpha}{2} (t - mT)^2 \right). \quad (4.30)$$

$$r(t) = \cos \left(2\pi f_c (t - \tau) + \frac{\pi \alpha}{2} (t - \tau - mT)^2 \right). \quad (4.31)$$

The transmitted and received signals are passed through a mixer and low-pass filter to get $g(t)$ expressed as

$$g(t) = s(t)r(t) = \cos \left(2\pi f_c t + \frac{\pi \alpha}{2} (t - mT)^2 \right) \cos \left(2\pi f_c (t - \tau) + \frac{\pi \alpha}{2} (t - \tau - mT)^2 \right). \quad (4.32)$$

Using the trigonometric identity $\cos(x) \cos(y) = \frac{1}{2}(\cos(x + y) + \cos(x - y))$ and the fact that the high-frequency component is filtered, we get

$$g(t) \cong \frac{1}{2} \cos(2\pi f_c \tau + 2\pi \alpha \tau (t - mT) - \pi \alpha \tau^2). \quad (4.33)$$

Substituting the τ into $g(t)$ results in

$$g(t) = \frac{1}{2} \cos\left(2\pi f_c \left(\frac{2(R+vt)}{c}\right) + 2\pi \alpha \left(\frac{2(R+vt)(t-mT)}{c}\right) - \pi \alpha \left(\frac{2(R+vt)}{c}\right)^2\right). \quad (4.34)$$

Since it can be assumed that $c^2 \gg (R+vt)^2$ the third term of the $g(t)$ can be considered to be negligible. The filtered mixer output becomes

$$g(t) = \frac{1}{2} \cos\left(2\pi f_c \left(\frac{2(R+vt)}{c}\right) + 2\pi \alpha \left(\frac{2(R+vt)(t-mT)}{c}\right)\right) \quad (4.35)$$

$$g(t) = \frac{1}{2} \cos\left(2\pi \left(\frac{2Rf_c}{c} + \frac{2f_c vt}{c}\right) + 2\pi \alpha \left(\frac{2(Rt - RmT + vt^2 - mvtT)}{c}\right)\right) \quad (4.36)$$

$$g(t) = \frac{1}{2} \cos\left(2\pi \left(\frac{2Rf_c}{c} + \frac{2f_c vt}{c} + \frac{(2R\alpha t - 2R\alpha mT + 2\alpha vt^2 - 2m\alpha vtT)}{c}\right)\right) \quad (4.37)$$

Considering t_s being the time from the start of the m th ramp chirp, we can write

$$t = t_s + mT, 0 \leq t_0 \leq T. \quad (4.38)$$

Substituting the above expression (4.38) into $g(t)$ results in

$$g(t_s) = \frac{1}{2} \cos\left(2\pi \left(\frac{2Rf_c}{c} + \frac{2f_c v(t_s + mT)}{c} + \frac{(2R\alpha(t_s + mT) - 2R\alpha mT + 2\alpha v(t_s + mT)^2 - 2m\alpha v(t_s + mT)T)}{c}\right)\right) \quad (4.39)$$

$$g(t_s) = \frac{1}{2} \cos\left(2\pi \left(\frac{2Rf_c}{c} + \frac{2f_c v(t_s + mT)}{c} + \frac{(2R\alpha(t_s + mT) - 2R\alpha mT + 2\alpha v(t_s^2 + 2mt_sT + m^2T^2) - 2m\alpha v(t_s + mT)T)}{c}\right)\right) \quad (4.40)$$

By assuming that second-order terms are negligible, $g(t_s)$ can be approximated as

$$g(t_s) = \frac{1}{2} \cos\left(2\pi\left(\frac{2Rf_c}{c} + \frac{2vmTf_c}{c} + \frac{(2R\alpha + 2vf_c + 2mBv)t_s}{c}\right)\right) \quad (4.41)$$

where the fact that $B = \alpha T$ is used. Assuming that the target is slowly moving and that the first term corresponds to constant phase, $g(t_s)$ can be compactly expressed as

$$g(t_s) = \frac{1}{2} \cos(2\pi(mTf_d + f_{pk}t_s)) \quad (4.42)$$

where

$$\begin{aligned} f_{pk} &= \frac{(2R\alpha + 2vf_c + 2mBv)}{c} = \frac{(2R\alpha)}{c} + f_d + \frac{(2mBv)}{c} \\ &= f_{\text{beat}} + f_d + f_m. \end{aligned} \quad (4.43)$$

f_{beat} is the beat frequency introduced by the time delay between the transmitted and received signals. f_m is the frequency component due to target motion during sweeps and is generally assumed to be negligible. For the fast chirp ramp, the Doppler shift between ramps is usually considered to be negligible. Therefore, range can be estimated from the beat frequency as

$$f_{\text{beat}} = \frac{(2R\alpha)}{c} \quad (4.44)$$

$$R = \frac{c f_{\text{beat}}}{2\alpha} \quad (4.45)$$

Range accuracy can be improved by taking the $f_d + f_m$ component into consideration. Traditionally, the range is computed by taking the FFT of the beat signal $g(t_s)$, from which the frequency peak of each chirp corresponds to the approximated f_{beat} .

Using the time-shift property, the Fourier transform of $g(t_s)$ is given by

$$G(f) = \frac{1}{4} (e^{i2\pi mTf_d})\delta(f - f_{\text{beat}}) + \frac{1}{4} (e^{-i2\pi mTf_d})\delta(f + f_{\text{beat}}) \quad (4.46)$$

From the peak of FFT spectrum, the corresponding beat frequency f_{beat} can be extracted while the phase at the peak frequency can be used to obtain the Doppler frequency and its corresponding velocity.

Range Resolution and Maximum Range

The resolution depends only on the ramp chirp bandwidth B and can be expressed as

$$\Delta R = \frac{c}{2B}. \quad (4.47)$$

If N points are used to compute the FFT, then the maximum range that can be computed is given by

$$R_{\max} = \left(\frac{N}{2}\right) * \Delta R = \frac{cN}{4B} \quad (4.48)$$

However, independent of FFT samples, the absolute maximum range depends on the chirp period and is given by $cT/2$.

Example Given a sweep bandwidth BW of 4 GHz, the achievable range resolution can be computed as follows:

$$\Delta R = \frac{c}{2B} = \frac{3 \times 10^8}{2 \times 4 \times 10^9} = 0.0375 \text{ [m]}.$$

Table 4.1 gives typical values of BW and corresponding range resolution values.

Velocity Resolution and Maximum Velocity

According to Nyquist theorem, the maximum Doppler frequency depends on the chirp period T and is given by

$$f_{d\max} = \frac{1}{2T}. \quad (4.49)$$

From the definition of Doppler frequency, the corresponding maximum velocity is given by

$$f_{d\max} = \frac{1}{2T} = \frac{2f_c v_{\max}}{c} \quad (4.50)$$

which results in

$$v_{\max} = \frac{c}{2T * 2f_c} = \frac{c}{4f_c T}. \quad (4.51)$$

If there are a total of M chirp ramps in a single scan, the Doppler resolution is $\Delta f_d = \frac{1}{MT}$, and then the velocity resolution Δv is given by

Table 4.1 Typical values of range resolution and corresponding maximum range assuming 256 FFT points

Bandwidth [GHz]	DeltaR [m]	Rmax (256 FFT points) [m]
0.48	0.3125	80.00
0.60	0.2500	64.00
1.00	0.1500	38.40
2.00	0.0750	19.20
4.00	0.0375	9.60

$$\Delta v = \Delta v = \frac{c}{2f_c} * \frac{1}{MT} = \frac{c}{2f_c MT}. \quad (4.52)$$

Increasing the number of ramp chirps M leads to improved velocity resolution.

Example Assuming 64 chirps in a single scan, a center frequency of 79 GHz and chirp period of 40 μ s velocity resolution can be computed as follows:

$$\Delta v = \frac{3 \times 10^8}{2 * 79 \times 10^9 * 64 * 40 \times 10^{-6}} = 2.23 \left[\frac{\text{m}}{\text{s}} \right].$$

References

1. Davis, M.: Radar frequencies and waveforms. In: 12th Annual International Symposium on Advanced Radio Technologies, Boulder, Colorado 27–29 July 2011
2. Lipa, B.J., Barrick, D.E.: FMCW Signal Processing. Mirage Systems, Sunnyvale, California (1990)
3. Ybarra, G.A., et al.: Optimal signal processing of frequency-stepped CW radar data, IEEE Trans. Microwave Theor Tech **43**(1) (1995)
4. Weiss, J.M.: Continuous-wave stepped-frequency radar for target ranging and motion detection. In: Proceedings of the 42nd Annual Midwest Instruction and Computing Symposium, Apr (2009)
5. Marc-Michael, M., Rohling, H.: Combination of LFFMCW and FSK modulation principles for automotive radar systems, German Radar Symposium GRS2000, Berlin 11–12 Oct 2000
6. Rohling, H., Meinecke, M.: Waveform design principles for automotive radar systems. In: 2001 CIE International Conference on Radar, pp. 1–4. IEEE, China (2001)
7. Khan, R.H., Mitchell, D.K.: Waveform analysis for high-frequency FMICW radar. In: IEE Proceedings F—Radar and Signal Processing, vol. 138, Issue 5, pp. 411–419 (1991)
8. Fioranelli, F., Salous, S., Raimundo, X.: Frequency-modulated interrupted continuous wave as wall removal technique in through-the-wall imaging. IEEE Trans. Geosci. Remote Sens. **52**(10), 6272–6283 (2014)

Chapter 5

Radar Target Detection



5.1 Introduction

Target detection is one of the very first operations, which has to be performed on received radar signal. The purpose of detection is normally to distinguish genuine target reflections from noise and clutter. This chapter discusses the main concepts and methods used for radar detection.

5.2 Target Models (Sterling 1–Sterling 4 Models)

The availability of a target’s radar cross-section (RCS) statistical characteristics could significantly improve the performance of target detection algorithms.

For this purpose, Swerling [1] introduced the Swerling models, to describe the statistical properties of the RCS of objects based of chi-square distribution of varying degrees of freedom. Five different Swerling models numbered I through V exist and are summarized below.

Swerling I

The target reflections in a single scan have a constant RCS magnitude σ , but it varies from scan to scan according to the chi-square probability density function (PDF) with two degrees of freedom. The PDF is given by the following expression.

$$f(\sigma) = \frac{1}{\sigma_{\text{avg}}} e^{-\frac{\sigma}{\sigma_{\text{avg}}}}, \quad \sigma \geq 0 \tag{5.1}$$

where σ_{avg} is the mean value of RCS.

Swerling II

The PDF for the RCS is the same as Eq. (4.1) but is independent from pulse to pulse instead of from scan to scan.

Swerling III

The RCS has the same description as Swerling I, with the difference that there are four degrees of freedom. The PDF is given by the following expression.

$$f(\sigma) = \frac{4\sigma}{\sigma_{\text{avg}}^2} e^{-\frac{2\sigma}{\sigma_{\text{avg}}}}, \quad \sigma \geq 0 \quad (5.2)$$

Swerling IV

The RCS varies from pulse to pulse according to Eq. (5.2), instead of from scan to scan.

Swerling V (Swerling 0)

The RCS is constant, which corresponds to infinite degrees of freedom.

The Swerling models are important in theoretical studies and also in the case where a single target of predictable RCS behavior is under investigation. However, in the case of multiple dynamic targets in clutter, determining detection thresholds using the models could be challenging [2].

Summary of Swerling models I to IV

PDF	Scan-to-scan fluctuations	Pulse-to-pulse fluctuations
$f(\sigma) = \frac{1}{\sigma_{\text{avg}}} e^{-\frac{\sigma}{\sigma_{\text{avg}}}}, \quad \sigma \geq 0$	Swerling I	Swerling II
$f(\sigma) = \frac{4\sigma}{\sigma_{\text{avg}}^2} e^{-\frac{2\sigma}{\sigma_{\text{avg}}}}, \quad \sigma \geq 0$	Swerling III	Swerling IV

The Swerling model can be integrated into the target detection probability as follows. If the input signal to the threshold detector which consists of a signal component of amplitude A embedded in Gaussian noise of variance ξ^2 is denoted by $r(t)$, the probability density function of $r(t)$ can be expressed as

$$f(r(t)) = \frac{r(t)}{\xi^2} I_0\left(\frac{rA}{\xi^2}\right) e^{-\frac{r(t)^2 + A^2}{2\xi^2}} \quad (5.3)$$

where $I_0(\cdot)$ is the zeroth-order modified Bessel function of first kind. The above expression defines a Rician probability density function. In case of noise only, we have,

$$f(r(t)) = \frac{r(t)}{\xi^2} I_0\left(\frac{r(t)A}{\xi^2}\right) e^{-\frac{r(t)^2}{2\xi^2}} \quad (5.4)$$

which is a Rayleigh distribution function. For large SNR, the Rayleigh approximates to Gaussian distribution.

$$f(r(t)) \approx \frac{1}{\sqrt{2\xi^2}} e^{-\frac{(r(t)-A)^2}{2\xi^2}} \quad (5.5)$$

Probability of false alarm

For a given detection threshold P_{thr} , the probability of false alarm P_{fa} is given by

$$P_{\text{fa}} = \int_{P_{\text{thr}}}^{\infty} \frac{r(t)}{\xi^2} e^{-\frac{(r(t))^2}{2\xi^2}} d(r(t)) = e^{-\frac{(P_{\text{thr}})^2}{2\xi^2}} \quad (5.6)$$

From the above equation, the threshold can be expressed in terms of the probability of false alarm as follows:

$$P_{\text{thr}} = \sqrt{2\xi^2 \ln\left(\frac{1}{P_{\text{fa}}}\right)}. \quad (5.7)$$

Probability of detection

From the PDF of $r(t)$, we can define the probability of detection as

$$P_D = \int_{P_{\text{thr}}}^{\infty} \frac{r(t)}{\xi^2} I_0\left(\frac{rA}{\xi^2}\right) e^{-\frac{r(t)^2 + A^2}{2\xi^2}} d(r(t)). \quad (5.8)$$

Many approximations of the P_{fa} and P_D are possible, and tables are available from which the required SNR to achieve a given of P_{fa} and P_D can be obtained [3].

The joint probability density function $f(x, \sigma)$ for target detection can be defined as

$$f(x, \sigma) = f(x/\sigma) f(\sigma). \quad (5.9)$$

The probability of detection $f(x)$ is given by

$$f(x) = \int f(x, \sigma) d(\sigma) = \int f(x/\sigma) f(\sigma) d(\sigma) \quad (5.10)$$

where the conditional probability density function $f(x, \sigma)$ is given by

$$f(x/\sigma) = \left(\frac{2x\xi^2}{M\sigma^2}\right)^{\frac{M-1}{2}} e^{-x - \frac{1}{2} * \frac{M\sigma^2}{\xi^2}} I_{M-1}\left(\sqrt{\frac{2Mx\sigma^2}{\xi^2}}\right) \quad (5.11)$$

M is the number of integrated pulses. Performing integration of $f(x)$ results in an incomplete gamma function [4].

From the automotive perspective, an insight into the fluctuating target model can be obtained from the results of an investigation of ground vehicles which consisted

of a large van, large truck, mid-size sedan, and mid-size truck described in [5]. Swerling I target behavior of ground vehicles was shown to be the most general case for all dataset distributions and parameter variations, although angular variation was limited. Additionally, Swerling III target behavior could be applied to about 90% of the sampled data but dependence on radar resolution was identified to be a relevant contributing factor. In conclusion, the choice of the Swerling model becomes a design decision taking into consideration signal-to-noise ratio (SNR), probability of detection, and probability of false alarm performance requirements, although the recommended conservative approach is to use the Swerling I target model. Some recent studies have also shown that, instead of the Gamma distribution used in Swerling models, the Weibull distribution is the best fit for measured radar reflections from a broad range of personal vehicle classes [6].

5.3 Peak Detection

Peak detection is necessary in order to select valid targets from received reflections in the presence of noise and clutter. The frequency spectrum normally consists of multiple peaks as shown in Fig. 5.1. For automotive applications, the task is made more difficult by the fact that roadside obstacles and ground reflections make up the received reflections. Various strategies with different degrees of success are available for accomplishing the task of peak detection. In most cases, peak detection is performed using Range–Doppler profiles obtained by DFT of the received reflections.

In the following subsections, we give a brief description of the theoretical background behind detection methods.

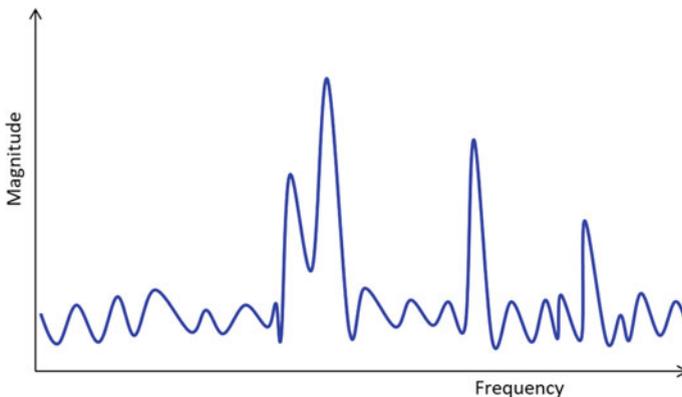


Fig. 5.1 An illustration of typical range spectrum obtained after FFT. The peak detection task is to extract target peaks, 4 in this case, and avoid peaks from the noise floor

5.3.1 Fixed Threshold

The threshold detector (Neyman–Pearson detector) is simplest method that can be used for peak detection. In this case, a predetermined threshold is set such that any target returns exceeding that threshold are considered valid. If the received signal is composed of target reflections $s(t)$ and noise $n(t)$, with the threshold set to P_{thr} , then the detection rule can be expressed as follows:

$$s(t) + n(t) \geq P_{thr}, \quad \text{True Detection} \quad (5.12)$$

$$n(t) \geq P_{thr}, \quad \text{False Alarm} \quad (5.13)$$

The meaning is that a false alarm occurs when target presence is decided when in actual fact only noise exists. The above expressions are true for non-fluctuating targets of identical reflection models but fails when a mixture of different targets exists in radar's field of view. Additionally, target reflection characteristics are affected by range, angle, size, and other factors such target shape. Therefore, multiple thresholds will be needed to cover such scenarios. For this reason, adaptive detection thresholds have been the subject of research for a long time.

5.3.2 Multi-cell Thresholding

Multi-cell thresholding techniques are necessary for dealing with cases which arise in practice where the computed DFT power spreads to adjacent range or Doppler bins. In this situation, it is desirable to detect only a single peak from such a group of bins. The strategy is simply to compare consecutive bins and make a detection decision when there is a change in gradient on either side of the bin under consideration. Using range bins as an example, the assumption here is that targets can be resolved if they are separated by at least one range bin. The drawback of this approach is that some targets exhibit fluctuating behavior around the peak which could in turn lead to detection of multiple targets when in fact only a single target exists. Moreover, without imposing a second threshold for noise, there is a possibility that noise could lead to excessive false alarms.

5.3.3 Constant False Alarm Detection (CFAR)

In order to address the issues associated with fixed thresholding and multi-cell thresholding, the CFAR approach has been applied with some measure of success [7]. It should be borne in mind that this approach doesn't come at no cost. In radar applications to automotive, besides radar size, computation cost is major issue. The increased

performance of the detection algorithm demands an increase in computation speed and device memory for every scan [8]. A trade-off between performance and cost has to be made. In this section, we briefly outline some of the important CFAR algorithms.

False alarm refers to false detection of targets from radar returns. It is traditionally computed by estimating the number of target detections when only noise is present in all range cells.

It is desirable to keep the false constant because detection algorithms are very sensitive to noise and clutter which is almost always present in received radar returns. Therefore, the general objective of all radar detection schemes is to ensure that false alarms do not fluctuate randomly. During the detection process, each cell/bin is evaluated for the presence or absence of a target using a threshold. It is beneficial to be able to detect both high- and low-fidelity targets while maintaining constant false alarm rate. This calls for an adaptive thresholding method, and most modern radars take this approach [9]. Among the most commonly used variations of the basic CFAR methods are cell-averaging CFAR (CA-CFAR), cell-averaging greatest-of CFAR (CAGO-CFAR), cell-averaging smallest-of CFAR (CAGO-CFAR), CA order statistic CFAR (CAOS-CFAR), cell-averaging statistic Hofele (CASH-CFAR), and max–min statistic CFAR (MAMIS-CFAR).

CFAR

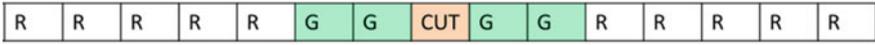
The CFAR principle dates back to the late 1960s [9]. Solutions to the false alarm problem involve implementation of constant false alarm rate (CFAR) schemes that vary the detection threshold as a function of the sensed environment. While there exists a large number of types of CFAR circuits, they are usually based around the “background averager” (sometimes referred to as cell-averaging CFAR). A simplified block diagram is shown in Fig. 5.2.

This circuit estimates the level of interference (noise or clutter) in radar range cells on either side of a range cell and uses this estimate to decide if there is a target present in the cell of interest in the center. The process steps out one cell in range and is repeated until all range cells have been investigated.

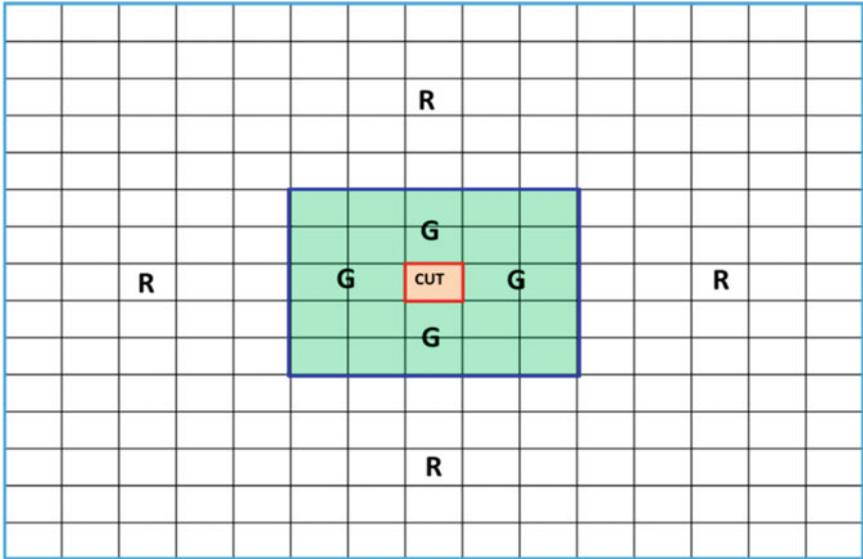
The basic idea behind the circuit is that when noise is present, the cells around the cell of interest will contain a good estimate of the noise in the tested cell; i.e., it is assumed that the noise or interference is spatially or temporarily homogeneous. Theoretically the circuit will produce a constant false alarm rate, which is independent of the noise or clutter level so long as the noise has a Rayleigh distribution in all range cells investigated by the circuitry.

Cell-Averaging CFAR (CA-CFAR)

In this approach, depicted in Fig. 5.3, rather than taking a single fixed value, the threshold is computed by taking the average power of the cells around the cell under test (CUT). The CUT is the cell for which the presence or absence of a target is to be determined. To make sure that the CUT does not influence the threshold calculation, cells immediately around the CUT are excluded from the computation. These cells



(a) 1D CFAR



(b) 2D CFAR

Fig. 5.2 An example of cells used for **a** 1D CFAR and **b** 2D CFAR. CUT is the cell under test. The cells in the region label G are the guard cells, while those in the region labeled R are the reference cells

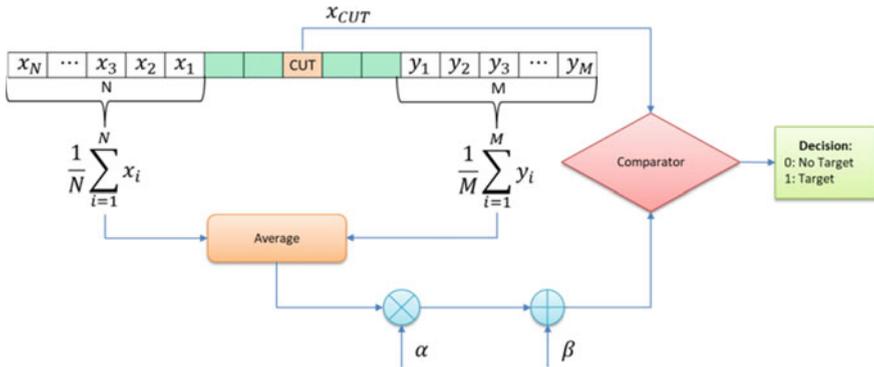


Fig. 5.3 Principle of CFAR detector. A multiplication α and an offset β are used to adjust the average value

are referred to as guard cells. For the 1D case, the guard cells are to the left and right of the CUT, while for 2D case, they form a ring around the CUT.

A target is judged present in the CUT if its power is both greater than that of all the guard cells and also greater than the computed average power level.

Cell-Averaging Greatest-Of CFAR (CAGO-CFAR)

This is a variation of CA-CFAR in which for the 1D case, the averaging is performed separately for the left- and right-side cells. The threshold is then simply the max of the two results. This method is illustrated in Fig. 5.4.

The average value used for threshold computation is determined as follows.

$$T_{\text{aver}} = \text{MAX} \left(\frac{1}{N} \sum_{i=1}^N x_i, \frac{1}{M} \sum_{i=1}^M y_i \right) \quad (5.14)$$

The advantage of this approach is simplified and reduced computation and improvement in target detection performance in some cases. The notable disadvantage is the inability to correctly detect masked targets, but this applies to all CA-CFAR variants.

Cell-Averaging Smallest-Of CFAR (CASO-CFAR)

This is similar to CAGO-CFAR except that the minimum instead of the maximum is computed. As can be imagined, this has the effect of increasing detection of low-power masked targets at the risk of increased misdetections. Figure 5.5 shows how the detector works.

The average value used for threshold computation is determined as follows.

$$T_{\text{aver}} = \text{MIN} \left(\frac{1}{N} \sum_{i=1}^N x_i, \frac{1}{M} \sum_{i=1}^M y_i \right) \quad (5.15)$$

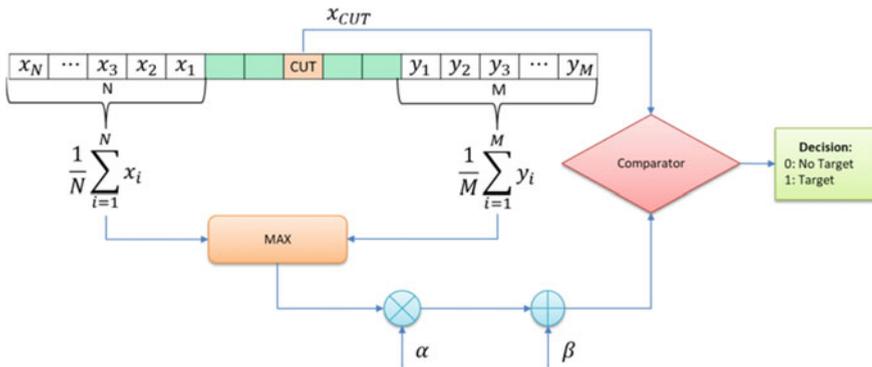


Fig. 5.4 Principle of CAGO-CFAR detector. A multiplication α and an offset β are used to adjust the average value

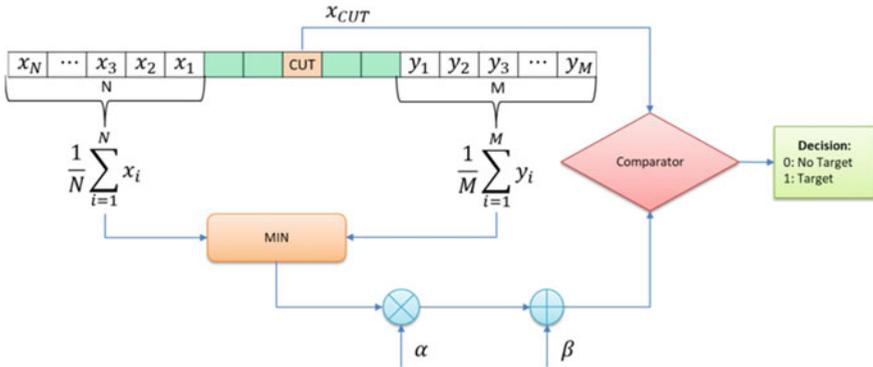


Fig. 5.5 Principle of CASO-CFAR detector. A multiplication α and an offset β are used to adjust the average value

Cell-Averaging Ordered Statistic CFAR (CAOS-CFAR)

Instead of getting averages as in other variants of CFAR, CAOS-CFAR computes the order statistic of the reference cells [10]. The rank in the order statistic is predetermined, and threshold average value can be selected by taking the rank for all the values or by CAGO-CFAR/CASO-CFAR when the left and right sides are considered separately. An improvement in performance is achieved for masked targets. However, the required sorting of cells greatly increases the computational complexity, making it difficult to implement in automotive applications. The circuit diagram of this method is shown in Fig. 5.6.

For any set of values $\{a_1, a_2, \dots, a_n\}$, the order statistic operation first sorts the values into a sequence $\{a_{(1)}, a_{(2)}, \dots, a_{(n)}\}$ from which the k th value is selected.

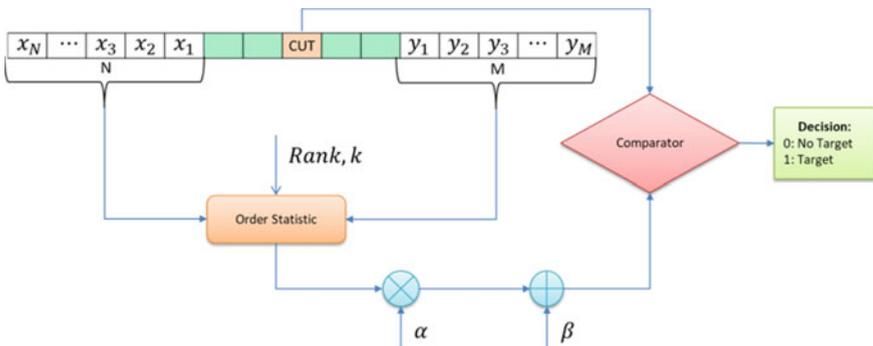


Fig. 5.6 Principle of CAOS-CFAR detector. A multiplication α and an offset β are used to adjust the average value

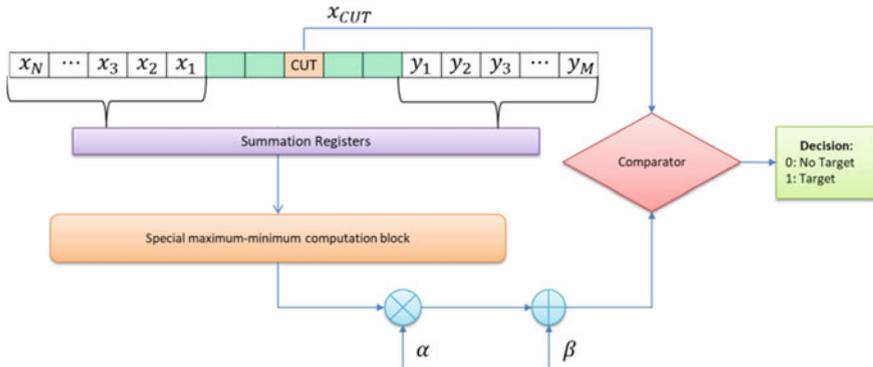


Fig. 5.7 Principle of CASH-CFAR detector. A multiplication α and an offset β are used to adjust the average value

Cell-Averaging Statistic Hofele (CASH-CFAR)

The cell-averaging statistic Hofele CFAR (CASH-CFAR), illustrated in Fig. 5.7, is based on a series of summing elements associated with each range cell and a specific maximum–minimum detector [11–13]. It utilizes a series of subregisters to perform summations from which the minimum is selected. This is then treated just like the average value from CA-CFAR. The advantage of the CASH-CFAR algorithm is reduced computation load compared to CAOS-CFAR while achieving similar performance.

Maximum–Minimum Statistic CFAR (MAMIS-CFAR)

The Maximum–Minimum Statistic CFAR (MAMIS-CFAR) is built on the same principle as CASH-CFAR, with the difference being the use of special maximum–minimum detector instead of summation blocks of the CASH-CFAR algorithm.

Although a variety of other CFAR algorithms such as the MAMIS and CASH have been proposed in the literature, CA-, CAGO-, CASO, and CAOS-CFAR remain the most popular and well-understood methods. As previously mentioned, computational complexity and other considerations may prevent the use of these more robust algorithms in favor of simple thresholding techniques, especially in automotive applications. Nevertheless, with the increasing prospect of reduction in hardware cost and availability of high-speed processors, the drift to high-performance algorithms is inevitable.

References

1. Swerling, P.: Probability of Detection for Fluctuating Targets. RAND Report RM-1217, ASTIA Document Number AD 80638, 17 Mar 1954
2. Buterbaugh, A.L.: Full polarimetric calibration for radar cross-section measurements: performance analysis. *IEEE Trans. Antennas Propag.* **52**(9), 2357–2365 (2004)

3. Marcum, J.I.: A statistical theory of target detection by pulsed radar. *IRE Trans. Inf. Theor.* **IT-6**, 59–267 (1960)
4. Mahafza, B.R.: *Radar Systems Analysis and Design Using MATLAB*, 2nd edn. Chapman & Hall/CRC (2005)
5. Raynal, A.M., Bickel, D.L., Denton, M.M., Bow, W.J., Doerry, A.W.: Radar cross section statistics of ground vehicles at Ku-band. In: *Proceeding of SPIE 8021, Radar Sensor Technology XV*, 80210E, 21 June 2011
6. Buller, W., van Nieuwstadt, L.: Statistical modelling of measured automotive radar reflections. In: *IEEE Instrumentation and Measurement Technology Conference*, May 2013
7. Kuck, M.: *Constant False Alarm Rate Detection of Radar Signals with Artificial Neural Networks*. Thesis (2007)
8. Cai, L.: Performance analysis of some new CFAR detectors under clutter. *J. Comput.* **6**(6), 1278–1285 (2011)
9. Skolnik, M.: *Introduction to Radar Systems*, 3rd edn. McGraw-Hill, New York (2001)
10. Rohling, H.: Radar CFAR thresholding in clutter and multiple target situations. *IEEE Trans. Aerosp. Electron. Syst.* **AES-19**(4), 608–621 (1983)
11. Radartutorial.eu: <http://www.radartutorial.eu/index.en.html>
12. Wiesbeck, W.: *Radar Systems Engineering*, 16th edn. WS 2009/2010
13. Hofele, F.X.: An innovative CFAR algorithm. In: *2001 CIE International Conference on Radar Proceedings*, pp. 329–333 (2001)

Chapter 6

Direction of Arrival (DOA) Estimation



6.1 Introduction

The importance of DOA estimation in radar processing for automotive applications cannot be overstated. It forms the third component of the radar cube: range, velocity, and angle. In practice, DOA estimation is often complicated by the fact that there will be multiple and unknown number of source signals impinging on the receiver array at the same time, with unknown amplitudes. Additionally, the received source signals are almost always corrupted by additive noise and clutter is present. Besides these challenges, we also have to deal with the multipath problem. Although the task of DOA estimation is not an easy one, there are several methods in the literature, developed over a period spanning more than 50 years, that can be used to estimate the number of source signals and their directions. The aim of the chapter is to give some of the actively and continuously researched methods in DOA estimation that are a key part of automotive radar algorithms. However, it is not the intention of this chapter to cover a detailed analysis of each and every method but to give a good insight into the strengths and the limitations of the most popular methods.

6.2 Classification of DOA Estimation Methods

The DOA methods can be broadly classified into quadratic, linear prediction, and subspace methods [1]. Examples include digital beamforming which can be classified as a quadratic method, forward-backward linear prediction and MUSIC which is a subspace method.

6.3 Approaches to DOA Estimation

There are several ways of performing DOA estimation. The methods shown in Table 6.1 have been widely used and research on improvements is currently ongoing.

Comparisons of these methods from a performance and an application point of view in various journals, conference proceedings, etc., have continuously been made until recently [2–4].

Even with advances in technology, there seems to be no magic method that solves the resolution, complexity, and robustness matrix. In automotive applications, the requirements placed on these methods are made even more difficult by limitations on antenna size and placement.

6.3.1 Signal Model

For every method, a signal model needs to be defined. Consider a radar system consisting of an array of M sensors (antennas elements) onto which signals from K sources (targets) are received. The received signals can be expressed as

$$\mathbf{X}(t) = \mathbf{A}(\theta)\mathbf{s}(t) + \mathbf{N}(t), \quad (6.1)$$

where $\mathbf{X}(t) = [x_1(t), \dots, x_M(t)]^T$ is the $M \times 1$ received sensor data vector, $\mathbf{A}(\theta) = [\mathbf{a}(\theta_1), \dots, \mathbf{a}(\theta_K)]$ is the $M \times K$ manifold matrix (also referred to as the steering matrix), $\mathbf{s}(t) = [s_1(t), \dots, s_K(t)]^T$ the $K \times 1$ source signal vector and $\mathbf{N}(t) = [n_1(t), \dots, n_M(t)]^T$ represents the $M \times 1$ sensor noise vector of variance σ^2 . The manifold matrix is made of steering vectors defined as

$$\mathbf{a}(\theta_i) = \left[1, e^{-\frac{j2\pi d \sin(\theta_i)}{\lambda}}, \dots, e^{-\frac{j2\pi d(N-1) \sin(\theta_i)}{\lambda}} \right]^T \quad (6.2)$$

where d is the antenna element spacing (considering a ULA = see figure), λ is the wavelength of propagating signals, θ_i is the direction (angle) of arrival of the signal from the i th source. “T” denotes the transpose operation. The signal model is depicted in Fig. 6.1.

In some methods such as DBF and Capon where optimization of weights is required, it is important to consider weighted output of the sensors as will be shown

Table 6.1 Categorization of popular methods used for DOA estimation

Classification	Methods
Quadratic	Maximum Likelihood (IQML, MODE), DBF, Capon, etc.
Linear Prediction (LP)	Forward–Backward LP, Maximum Entropy
Subspace	MUSIC, Root-MUSIC, ESPRIT, WSF, Minimum Norm, PHD, etc.

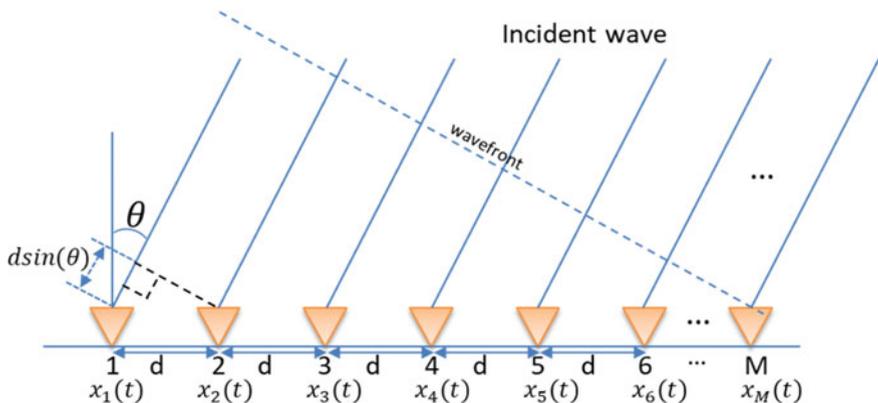


Fig. 6.1 An illustration of the signal model. The direction of arrival θ corresponds to any of the source directions θ_i and d is the antenna element spacing. The minimum number of antenna elements is two

the following sections. In that case, the weighted linear combination of sensor outputs can be expressed as follows:

$$y(t) = \sum_{m=1}^M w_m^* x_m(t) = \mathbf{w}^H \mathbf{X} \quad (6.3)$$

where w_i^* is the weight for the i th sensor. The asterisk (*) denotes complex conjugation while H denotes the conjugate or Hermitian transpose.

The output power, $P(\mathbf{w})$, of the sensor array can be expressed as follows:

$$P(\mathbf{w}) = E[|y(t)|^2] = \mathbf{w}^H E[\mathbf{X} \mathbf{X}^H] \mathbf{w} = \mathbf{w}^H \mathbf{R} \mathbf{w}, \quad (6.4)$$

where $E[\cdot]$ denotes the expectation operation, where \mathbf{R} denotes the input signal covariance matrix.

6.3.2 DOA Estimation Methods

In the following sections, we give details of computation involved in various popular DOA methods.

Digital Beamforming (DBF)

The DBF algorithm maximizes the output power in a specific direction by optimizing the weight vector [5]. The optimum weight vector is given by the following expression.

$$\mathbf{w}_{\text{opt}} = \frac{\mathbf{a}(\theta)}{\sqrt{\mathbf{a}^H(\theta)\mathbf{a}(\theta)}} \quad (6.5)$$

Substituting the optimum weight vector into Eq. (6.4) gives the DBF power spectrum as follows.

$$P_{\text{DBF}}(\theta) = \frac{\mathbf{a}^H(\theta)\mathbf{R}\mathbf{a}(\theta)}{\mathbf{a}^H(\theta)\mathbf{a}(\theta)} \quad (6.6)$$

Properties

The DBF has low computation complexity since it only requires the estimation of covariance matrix \mathbf{R} of the received data vector. By creating a steering vector at equally spaced angles in the desired interval, the angle corresponding to the maximum power can be extracted as the DOA.

The DBF is known for its inability to resolve closely spaced targets. However, it can be used as the first step to narrow the target range for high-resolution methods.

Capon

The Capon algorithm aims to maintain constant gain for signals arriving from a specific direction while giving smaller weight to noise [6]. It is an alternative to maximum likelihood methods that is used to solve for the minimum variance distortionless response (MVDR) of an array such that the signal to interference ratio is maximized. The optimization problem can be expressed as

$$\min(P(\mathbf{w})) \text{ subject to } \mathbf{w}^H\mathbf{a}(\theta) = 1. \quad (6.7)$$

The resulting optimum weight vector is given by

$$\mathbf{w}_{\text{opt}} = \frac{\mathbf{R}^{-1}\mathbf{a}(\theta)}{\mathbf{a}^H(\theta)\mathbf{R}^{-1}\mathbf{a}(\theta)}. \quad (6.8)$$

Substituting the optimum weight vector into Eq. (6.4) gives the Capon power spectrum as follows.

$$P_{\text{Capon}}(\theta) = \frac{1}{\mathbf{a}^H(\theta)\mathbf{R}^{-1}\mathbf{a}(\theta)} \quad (6.9)$$

Properties

The Capon method gives superior performance to the DBF in terms of resolving closely spaced targets [3]. Additionally, the major computation is the determination of the inverse covariance matrix from the data. This makes it very attractive in many situations where high resolution is a requirement. It is also worth noting that both Capon and DBF do not require prior knowledge of the number of signal sources. The DOA and power of the source signals can be simultaneously estimated. The ability

of Capon to separate multiple targets is only limited by SNR and receiver array size which is a common restriction for most high-resolution DOA methods.

The downside of the Capon method lies in the need to stabilize the computation. Determination of the inverse of nearly-singular covariance matrix is not trivial, especially in automotive applications where the received signals are degraded by both noise and clutter. Due to the limited aperture of the antenna, the covariance matrix cannot be reliably computed without some form of regularization as part of pre-processing. This hurdle in the computation of the covariance matrix, however, does not deter the use of Capon. Of course, some preprocessing techniques have to be applied as mentioned above.

Multiple Signal Classifier (MUSIC)

MUSIC takes the subspace approach to the DOA estimation problem [7]. It utilizes eigen-decomposition as the main tool. The theory behind eigen-decomposition can be found in [8]. Here, we give the main points.

The main assumption behind subspace methods is that the signal and noise subspaces are orthogonal. Using the orthogonality principle, a pseudo-spectrum can be computed.

The covariance matrix can be separated into the signal and noise components.

By definition, $\mathbf{R} = E[\mathbf{X}\mathbf{X}^H]$ and can be written as

$$\begin{aligned}\mathbf{R} &= E[(\mathbf{A}(\theta)\mathbf{s}(t) + \mathbf{N}(t))(\mathbf{A}(\theta)\mathbf{s}(t) + \mathbf{N}(t))^H] \\ &= \mathbf{A}(\theta)\mathbf{E}[\mathbf{s}(t)\mathbf{s}^H(t)]\mathbf{A}(\theta)^H + E[\mathbf{N}(t)(\mathbf{N}(t))^H] \\ &= \mathbf{A}(\theta)\mathbf{R}_s\mathbf{A}(\theta)^H + \sigma^2\mathbf{I},\end{aligned}\tag{6.10}$$

where \mathbf{R}_s is the unobserved source signal covariance matrix and \mathbf{I} denotes the identity matrix. For simplicity, explicit dependency on time t in Eq. (6.10) has been omitted. It can be observed that the covariance matrix can be separated into the signal subspace and the noise subspace that is independent of θ .

Through eigen-decomposition, the covariance matrix \mathbf{R} can be decomposed into matrices of eigenvectors and eigenvalues as follows.

Since the signal subspace is of size K , then $M - K$ eigenvalues of R belong to the noise subspace. If \mathbf{u}_i is an eigenvector of \mathbf{R} , then we can write

$$\mathbf{R}\mathbf{u}_i = (\mathbf{A}(\theta)\mathbf{R}_s\mathbf{A}(\theta)^H + \sigma^2\mathbf{I})\mathbf{u}_i = \lambda_i\mathbf{u}_i\tag{6.11}$$

where λ_i , $i = 1, \dots, M$, is the i th eigenvalue of \mathbf{R} . The eigenvalues of R can be partitioned into $\lambda_i = \sigma_i^2 + \sigma^2$ for $i = 1, \dots, K$ and $\lambda_i = \sigma^2$ for $i = K + 1, \dots, M$.

Since \mathbf{R}_s is positive-definite, we can write

$$\mathbf{u}_i^H(\mathbf{A}(\theta)\mathbf{R}_s\mathbf{A}(\theta)^H)\mathbf{u}_i = 0, i = K + 1, \dots, M\tag{6.12}$$

$$(\mathbf{A}(\theta)^H)\mathbf{u}_i = 0, i = K + 1, \dots, M\tag{6.13}$$

$$(\mathbf{a}(\theta_k)^H)\mathbf{u}_i = 0, k = 1, \dots, K, i = K + 1, \dots, M. \quad (6.14)$$

From Eq. (6.14), the MUSIC algorithm finds angles θ_k for which the signal subspace is orthogonal to noise subspace.

The eigenvectors of the noise subspace are defined by

$$\mathbf{U}_N = [\mathbf{u}_{K+1}, \dots, \mathbf{u}_M]. \quad (6.15)$$

Based on Eq. (6.15), the MUSIC pseudo-spectrum can be computed using the following expression.

$$P_{\text{MUSIC}}(\theta) = \frac{1}{\mathbf{a}^H(\theta)\mathbf{U}_N\mathbf{U}_N^H\mathbf{a}(\theta)} \quad (6.16)$$

Properties

The MUSIC algorithm is one of the high-resolution techniques for DOA estimation. It is capable of reliably resolving targets with angular separation of as low as 1° from severely degraded received signals with very few spurious peaks in the pseudo-spectrum. Since MUSIC computes the pseudo-spectrum of the received signal, it cannot be used to estimate the power of the signal sources. For power estimation, DBF or Capon can be used in conjunction with MUSIC. By increasing the number of snapshots used for spectrum estimation, the ability of MUSIC to separate multiple targets can be improved.

Despite the high performance, MUSIC requires prior knowledge of the size of the signal subspace. It also depends on eigen-decomposition which requires high computational complexity.

Root-MUSIC

This is a MUSIC algorithm variant where DOA estimation performed by finding roots of a polynomial [9]. The denominator of $P_{\text{MUSIC}}(\theta)$ in Eq. (6.16) can be expressed as a polynomial $P(z)$ such that

$$P(z) = \mathbf{a}^H(\theta)\mathbf{U}_N\mathbf{U}_N^H\mathbf{a}(\theta) \quad (6.17)$$

where

$$\mathbf{a}(\theta) = [1 \ z^1 \ z^2 \ \dots \ z^{M-1}]^T \quad (6.18)$$

and

$$z = e^{-\frac{j2\pi d}{\lambda}\sin(\theta)}. \quad (6.19)$$

We can therefore expand $P(z)$ into a polynomial of the form

$$P(z) = \sum_{k=-M+1}^{M-1} C_k z^k. \quad (6.20)$$

Under ideal conditions, the roots of $P(z)$ would be on the unit circle, with angles corresponding to the DOA of the received signals. In real situation, the presence of noise results in a shift of the amplitudes from the unit circle. Therefore, the roots of $P(z)$ that lie closest to the unit circle can be taken to be the poles of the MUSIC pseudo-spectrum. The roots can be written as

$$z_k = |z_k| e^{j \arg(z_k)}. \quad (6.21)$$

From the roots with magnitude closest to 1, the DOA can be estimated as

$$\theta_k = \sin\left(\frac{\lambda}{2\pi d} \arg(z_k)\right), k = 1, \dots, K. \quad (6.22)$$

Properties

The Root-MUSIC has similar performance to MUSIC. It could give better performance in some certain situations where phase noise is a concern. As with other subspace algorithms, it is still necessary to know the number of sources in advance. In severe situations of noise and clutter, the magnitudes of the roots could fall far from the unity circle, making it difficult to extract the roots of interest.

Estimation of Signal Parameters via Rotational Invariance Technique (ESPRIT)

ESPRIT is a subspace method that takes a different approach from MUSIC in that it decomposes the array into two sub-arrays, displaced by a known vector, from which the DOA can be estimated without the computationally expensive peak search procedure [10]. An illustration of the displacement of a single array into two sub-arrays is shown in Fig. 6.2. It should be noted that physically a single ULA array exists from which sub-arrays are created. ESPRIT also places less burden on calibration task by relaxing antenna geometry in that the sub-arrays need not be of identical antenna elements and that their physical positions can be arbitrarily chosen [1].

For the two sub-arrays of size M denoted by \mathbf{x} and \mathbf{y} , the received data vectors can be formulated by the following equations

$$\begin{aligned} \mathbf{x}(t) &= \sum_{i=1}^K \mathbf{a}(\theta_i) s_i(t) + \mathbf{n}_x(t) = \mathbf{A} \mathbf{s}(t) + \mathbf{n}_x(t) \\ \mathbf{y}(t) &= \sum_{i=1}^K \mathbf{a}(\theta_i) e^{j\gamma_i} s_i(t) + \mathbf{n}_y(t) = \mathbf{A} \Phi \mathbf{s}(t) + \mathbf{n}_y(t) \end{aligned} \quad (6.23)$$

where $\gamma_i = \frac{\omega_0 \Delta}{c} \sin \theta_i$, $\Phi = \text{diag}(e^{j\gamma_1}, \dots, e^{j\gamma_K})$, c is the speed of light, ω_0 is the center frequency and Δ is translational displacement between the two arrays. The

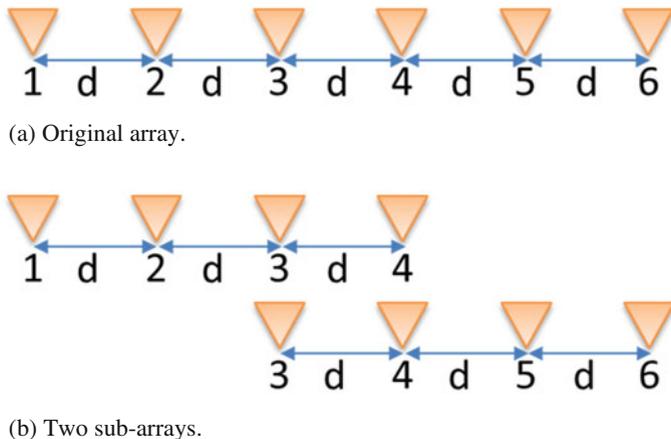


Fig. 6.2 An illustration of two four-element sub-arrays with displacement $2d$ formed from a six-element ULA with an element spacing is of d

$\text{diag}(\cdot)$ operator converts the sequence into a diagonal matrix. The manifold matrix \mathbf{A} is of size $M \times K$. ESPRIT aims to estimate the DOA from the rotation operator Φ . This can be accomplished by first defining a $2M \times 1$ vector $\mathbf{z}(t)$ such that

$$\mathbf{z}(t) = \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{y}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A} \\ \mathbf{A}\Phi \end{bmatrix} \mathbf{s}(t) + \begin{bmatrix} \mathbf{n}_x(t) \\ \mathbf{n}_y(t) \end{bmatrix} = \mathbf{A}_z \mathbf{s}(t) + \mathbf{n}_z(t). \quad (6.24)$$

The covariance matrix of $\mathbf{z}(t)$ is given by

$$\mathbf{R}_z = E[\mathbf{z}(t)\mathbf{z}(t)^H] = \mathbf{A}_z \mathbf{R}_s \mathbf{A}_z^H + \sigma^2 \mathbf{I}. \quad (6.25)$$

Eigen-decomposition of \mathbf{R}_z results in eigenvectors with signal subspace of size K and noise subspace of size $2N - K$. Assuming $K \leq M$, and denoting the denoting the $2N \times K$ signal subspace eigenvectors as \mathbf{E}_s , it is known from generalized eigen-decomposition that \mathbf{E}_s and \mathbf{A}_z have the same span. There exists a nonsingular transformation matrix \mathbf{T} such that

$$\mathbf{E}_s = \mathbf{A}_z \mathbf{T}. \quad (6.26)$$

In addition, \mathbf{E}_s can be partitioned into \mathbf{E}_x and \mathbf{E}_y of rank K as follows

$$\mathbf{E}_s = \mathbf{A}_z \mathbf{T} = \begin{bmatrix} \mathbf{A} \\ \mathbf{A}\Phi \end{bmatrix} \mathbf{T} = \begin{bmatrix} \mathbf{A}\mathbf{T} \\ \mathbf{A}\Phi\mathbf{T} \end{bmatrix} = \begin{bmatrix} \mathbf{E}_x \\ \mathbf{E}_y \end{bmatrix}. \quad (6.27)$$

Defining a matrix of $\mathbf{E}_{xy} = [\mathbf{E}_x \ \mathbf{E}_y]$ of size $N \times 2K$ and rank K , there exist a $2K \times K$ matrix \mathbf{F} such that

$$\begin{aligned}
0 &= \mathbf{E}_{xy} \mathbf{F} = [\mathbf{E}_x \mathbf{E}_y] \begin{bmatrix} \mathbf{F}_x \\ \mathbf{F}_y \end{bmatrix} \\
\mathbf{E}_x \mathbf{F}_x + \mathbf{E}_y \mathbf{F}_y &= \mathbf{A} \mathbf{T} \mathbf{F}_x + \mathbf{A} \Phi \mathbf{T} \mathbf{F}_y = 0 \\
\mathbf{A} \Phi \mathbf{T} &= -\mathbf{A} \mathbf{T} \mathbf{F}_x \mathbf{F}_y^{-1}.
\end{aligned} \tag{6.28}$$

The above expression means that \mathbf{F} spans the null-space of $[\mathbf{E}_x \mathbf{E}_y]$ since we are working on the assumption that $K \leq M$. Since \mathbf{T} is full column rank, its inverse exist, leading to

$$\begin{aligned}
\mathbf{A} \Phi &= -\mathbf{A} \mathbf{T} \mathbf{F}_x \mathbf{F}_y^{-1} \mathbf{T}^{-1} \\
\Phi &= \mathbf{T} \mathbf{F}_x \mathbf{F}_y^{-1} \mathbf{T}^{-1} = \mathbf{T} \psi \mathbf{T}^{-1}
\end{aligned} \tag{6.29}$$

where $\psi = -\mathbf{F}_x \mathbf{F}_y^{-1}$.

The eigenvalues of ψ are equal to the diagonal elements of Φ from which the DOA can be estimated. The presence of noise and calibration errors in measurements usually leads to the TLS method being used for DOA estimation from Φ .

TLS ESPRIT

The TLS algorithm can be summarized by the following steps.

- 1) Compute an estimate of \mathbf{R}_z from measurements.
- 2) Perform generalized eigen-decomposition of \mathbf{R}_z to get $\mathbf{R}_z \mathbf{E}_z = \Sigma_z \mathbf{E}_z \Lambda_z$.
- 3) Estimate the signal subspace size K .
- 4) Estimate \mathbf{E}_s and decompose into \mathbf{E}_x and \mathbf{E}_y .
- 5) Define the matrix \mathbf{E}_D and perform eigen-decomposition as follows:

$$\mathbf{E}_D = \begin{bmatrix} \mathbf{E}_x^H \\ \mathbf{E}_y^H \end{bmatrix} [\mathbf{E}_x \mathbf{E}_y] = \mathbf{E} \Lambda \mathbf{E}^H \tag{6.30}$$

- 6) Partition the \mathbf{E} into $4K \times K$ matrices as follows

$$\mathbf{E} = \begin{bmatrix} \mathbf{E}_{11} & \mathbf{E}_{12} \\ \mathbf{E}_{21} & \mathbf{E}_{22} \end{bmatrix} \tag{6.31}$$

- 7) Calculate the eigenvalues of $\psi = -\mathbf{E}_{12} \mathbf{E}_{22}^{-1}$ to obtain

$$\hat{\phi}_k = \lambda_k(-\mathbf{E}_{12} \mathbf{E}_{22}), k = 1, \dots, K \tag{6.32}$$

- 8) The DOA estimates are obtained using

$$\hat{\theta}_k = \sin^{-1} \left(c \frac{\arg(\hat{\phi}_k)}{\omega_0 \Delta} \right) \tag{6.33}$$

where the eigenvalues are extracted from ψ as explained above.

Several variations and improvements of the ESPRIT algorithm such as unitary ESPRIT and Beamspace ESPRIT exist in the literature and whose details can be found in [1].

Properties

As outlined above, ESPRIT has advantages associated with not requiring the exhaustive peak search in order to determine the DOAs. Therefore, it is more attractive in terms of computational complexity. However, the requirement of additional sensors could result in noisy DOA estimates. As with all algorithms, the key to performance lies in the ability to accurately estimate the covariance matrix. The covariance estimate can be improved with the availability of more sensors.

6.3.3 Spatial Smoothing

As can be observed from the above algorithms, estimation of the covariance matrix \mathbf{R} is a key component of the DOA estimation process. The size of the correlation matrix is determined by the size of the received data vector size which in turn depends on the length of the receive antenna array. For example, in automotive radar applications, receive antenna array sizes of less than 10 elements are commonly used due to limitations placed on the radar sensor size. Typically, the number of the antenna elements can be small as four elements. The data received from these elements is the input to the covariance matrix estimation method. In most cases, in order to reduce computational complexity, the received signal is transformed to the frequency domain by discrete Fourier transformation which is almost always implemented by FFT algorithm.

For eigen-based algorithms, the DOA estimation is only possible if the covariance matrix is nonsingular, which means the signals are non-coherent or uncorrelated [11]. The source of coherence is due to natural propagation characteristics like multipath, or it could be a result of artificial signals like jamming. One method used to decorrelate the signals is spatial smoothing (SS). Spatial smoothing is accomplished by first dividing the antenna array into sub-arrays from an average covariance matrix can be computed. Assuming that the array can be divided into M sub-arrays and denoting the covariance matrix computed from each sub-array as R_m^f , $m = 1, \dots, M$, and the superscript indicates computation in the forward direction, i.e., in increasing element number as shown in Fig. 6.3, we can compute the covariance matrix using the following expression.

$$R_{SS}^f = \frac{1}{M} \sum_{m=1}^M R_m^f. \quad (6.34)$$

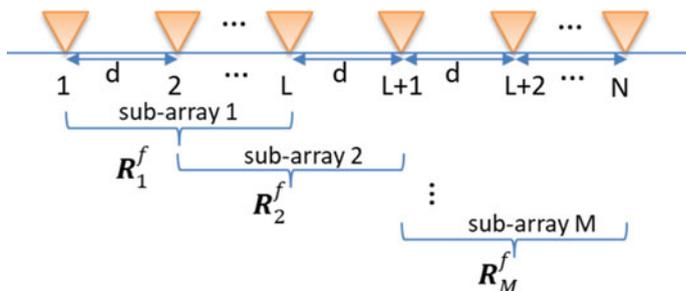


Fig. 6.3 Illustration of sub-array construction for spatial smoothing processing

The spatially smoothed covariance matrix R_{SS}^f can be used to compute eigenvalues and eigenvectors for DOA estimation. If the number of elements in the sub-array is p , then the number of sub-arrays $M = N - p + 1$. In order to detect K sources, p must be greater than or equal to $K + 1$. The minimum number of elements required will be $2K$. Therefore, spatial smoothing effectively reduces the number of detectable targets by half. For an N -element array, up to $N - 1$, targets can be detected without spatial smoothing in the non-coherent case compared to $N/2$ for the coherent case. This means that for a fixed number of array elements, a trade-off has to be made between the number of sub-arrays and detectable targets.

As an improvement on spatial smoothing the target detection problem, forward-backward spatial smoothing (FBSS) was proposed in [12]. As the name suggests, the FBSS method averages the covariance matrices computed in the forward direction as described in the SS method, and covariance matrices computed from N th element back to the first element. The backward computation can be efficiently accomplished by using the exchange matrix, \mathbf{J} . The overall covariance matrix \mathbf{R}_{FBSS} is calculated using the following expression.

$$\mathbf{R}_{FBSS} = \frac{\mathbf{R}_{SS}^f + \mathbf{J}(\mathbf{R}_{SS}^f)^H \mathbf{J}}{2} \quad (6.35)$$

where

$$\mathbf{J} = \begin{bmatrix} 0 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 0 \end{bmatrix}. \quad (6.36)$$

For an N -element array, the number of coherent sources that can be detected increases to $2N/3$.

Example As examples of DOA estimation, we consider three of widely used methods: DBF, Capon, and MUSIC.

The following system parameters are used:

Speed of light, c : 3.0e8 [m/s]

Center frequency, f_0 : 77.5e9 [Hz]

Number of ULA antenna elements, N : 5

Antenna separation, d : $0.5 * \lambda$, (λ is the wavelength)

Number of sources, N : 2

Direction arrival of source 1, theta1: -10 [deg], -2 [deg]

Direction arrival of source 2, theat2: $+10$ [deg], $+2$ [deg]

Noise variance: 0.015.

Spatial smoothing is applied to the covariance matrix using sub-arrays of four-element length.

This gives two sub-arrays in total for averaging. The simulation results are shown in Figs. 6.4, 6.5 and 6.6.

The results illustrate that the subspace approach, MUSIC, gives sharper spectral picks than the more conventional DBF and Capon methods. When the sources are closer together at -2 [deg] and $+2$ [deg], MUSIC resolves them better than both DBF and Capon as shown in Figs. 6.7, 6.8 and 6.9.

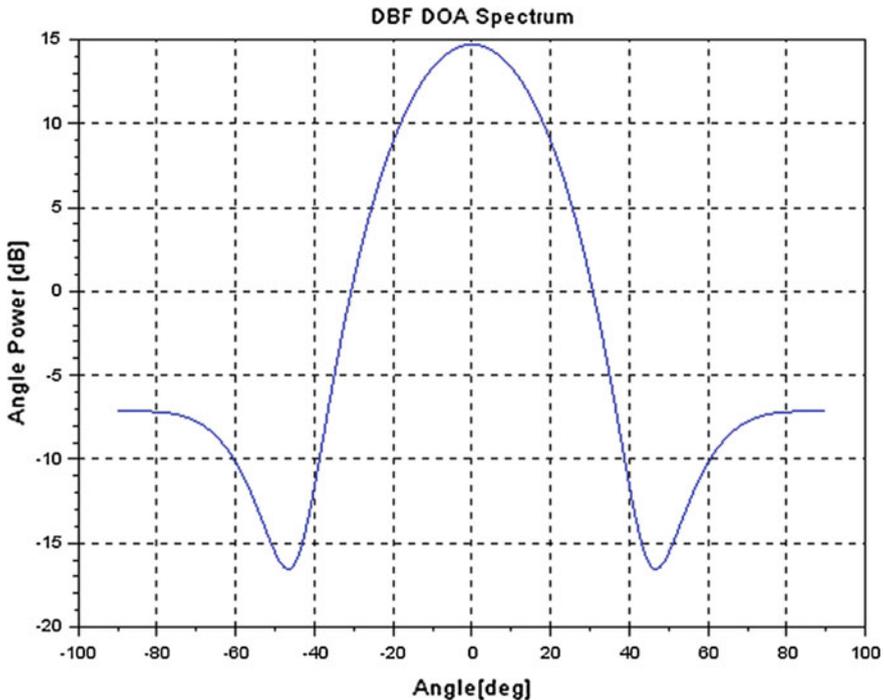


Fig. 6.4 DOA estimation by the DBF method with sources two at -10 [deg] and $+10$ [deg] using a five-element ULA antenna

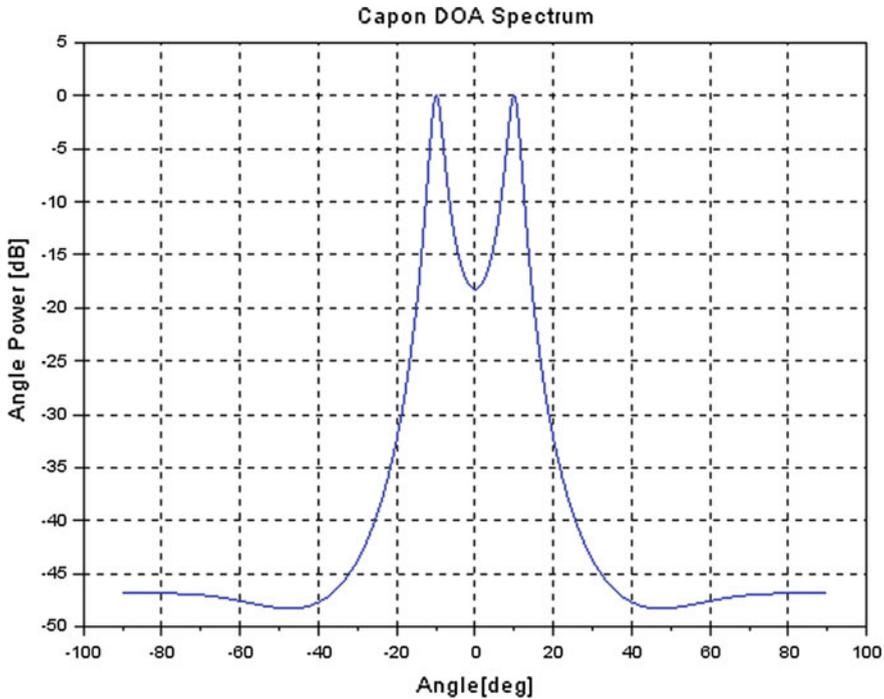


Fig. 6.5 DOA estimation by the Capon method with sources two at -10 [deg] and $+10$ [deg] using a five-element ULA antenna

6.3.4 Other DOA Algorithms

Besides the algorithms given above, some other algorithms such as ML, LP, and WSF exist [1]. However, the stability and computational complexity issues make them difficult to apply in automotive applications. This does not necessarily mean that these algorithms are inferior but they are limited by currently available technology to be used in real-time situation. Some recent methods like propagator method avoid eigen-decomposition altogether and instead employ least squares methods which makes them worth exploring [13, 14]. We briefly explain some of these algorithms below.

Minimum Norm Method

The minimum norm method is one of the oldest high-resolution methods for DOA estimation [15]. From the estimated covariance matrix \mathbf{R} , singular value decomposition (SVD) is performed to obtain the matrices U , S , and V . The noise subspace eigenvectors are extracted as $E_N = U(:, K + 1 : N)$, which means all columns of U from the $(K + 1)$ -th to the N th column. The spectrum is constructed based on the minimum norm vector lying in the noise subspace whose first element equals 1. The spectrum is computed from the following expression:

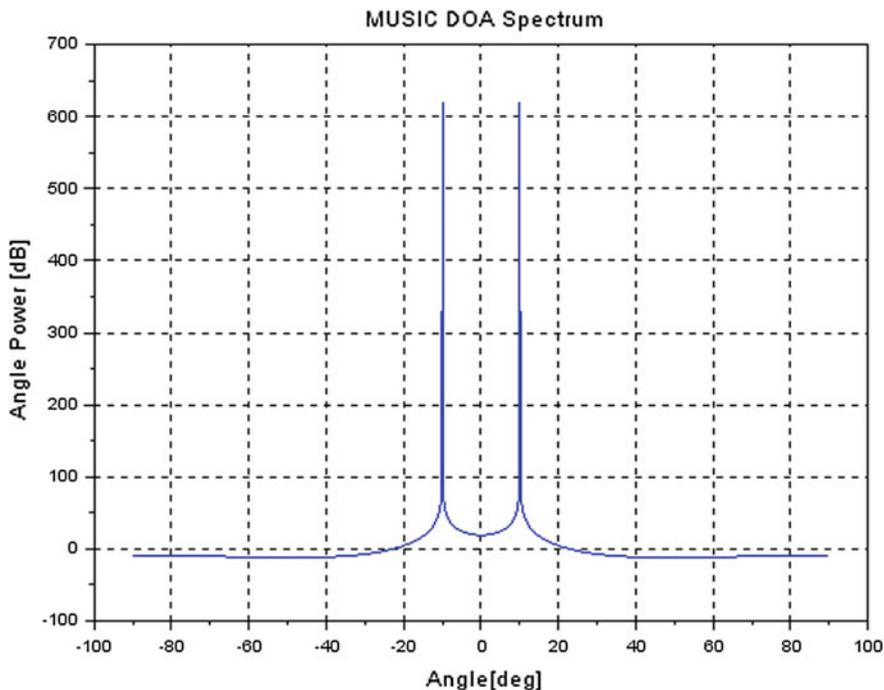


Fig. 6.6 DOA estimation by the MUSIC method with sources two at -10 [deg] and $+10$ [deg] using a five-element ULA antenna

$$P_{\text{MinNorm}}(\theta) = \frac{1}{|a(\theta)E_N E_N^H u|^2}, \quad (6.37)$$

where $u = [1 \ 0 \ 0 \ \dots \ 0]^T$ and $a(\theta)$ is the steering vector as previously defined.

Since the minimum norm method belongs to eigen-based approaches, it cannot be used to reliably estimate the power of source signals. It is primarily used for DOA of estimation. Information about the number of sources must be known beforehand. Estimation of DOA in degraded environments is possible, but there is an increased tendency of obtaining spurious peaks in the spectrum.

Maximum Entropy Method (MEM)

The MEM is spectral estimation method based on the extrapolation of the autocorrelation function that aims at maximizing the signal entropy (“uncertainty”), by using autoregressive (AR) coefficients [16]. The AR coefficients, \mathbf{a} , minimize the prediction error, i.e.,

$$\mathbf{a} = \operatorname{argmin}\{a^H \mathbf{R}\}, \quad (6.38)$$

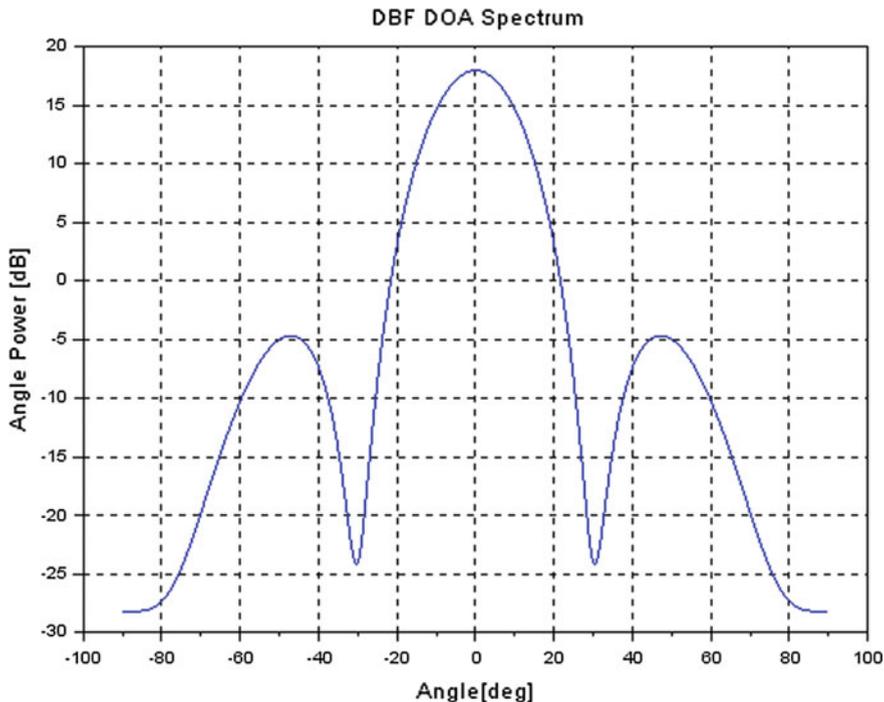


Fig. 6.7 DOA estimation by the DBF method with sources two at -2 [deg] and $+2$ [deg] using a five-element ULA antenna

subject to the constraint that $\mathbf{a}^H \mathbf{e}_1 = 1$ where $\mathbf{e}_1 = [1 \ 0 \ \dots \ 0]^T$. Details of other noise-robust methods of estimating AR parameters can be found in [17]. Using Lagrange method, the AR coefficients can be computed as

$$\mathbf{a} = \frac{\mathbf{R}^{-1} \mathbf{e}_1}{\mathbf{e}_1^T \mathbf{R}^{-1} \mathbf{e}_1}. \quad (6.39)$$

The spectrum estimation is performed using

$$P_{\text{MEM}}(\theta) = \frac{1}{|\mathbf{a}(\theta)^H \mathbf{C}_j|^2}, \quad (6.40)$$

where \mathbf{C}_j represents the j th column of the inverse of the covariance matrix. The choice of j is arbitrary and affects performance of the MEM. The relationship between the AR parameters and spectral estimates is based on the assumption that the autocorrelation function exists and that the extrapolation is valid. There are some issues with the validity of these assumptions which have been outlined in [16]. The MEM could be very difficult to extend to multi-dimensional DOA estimation problems.

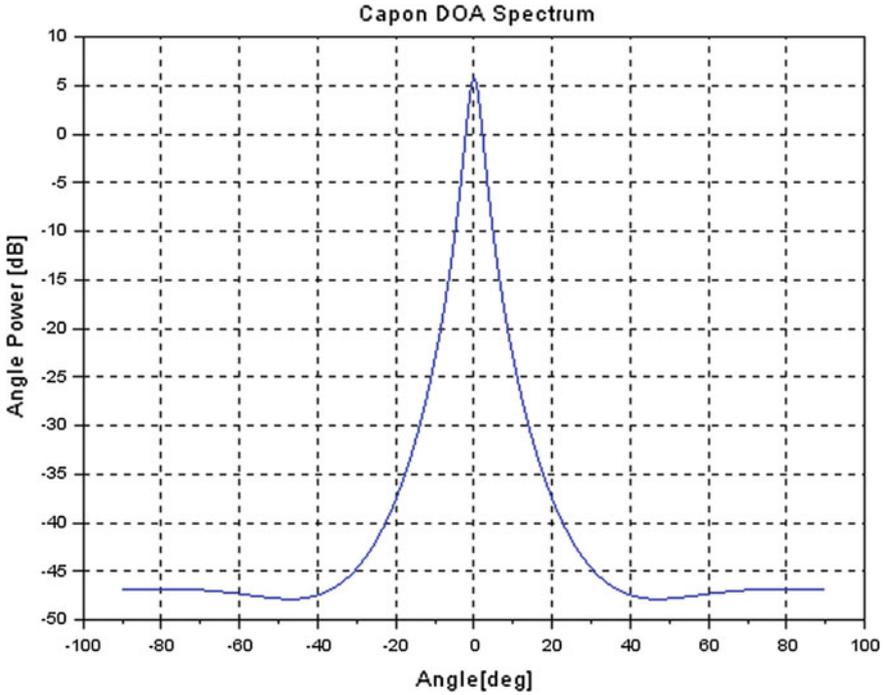


Fig. 6.8 DOA estimation by the Capon method with sources two at -2 [deg] and 2 [deg] using a five-element ULA antenna

Linear Prediction

The linear prediction (LP) method is widely used in audio and speech processing [15, 18]. The idea behind LP is the minimization of mean output power subject to the constraint that the weight of an arbitrarily selected element of the array is unity. The array weight vector is given by

$$\mathbf{w} = \frac{\mathbf{R}^{-1}\mathbf{u}}{\mathbf{u}^H \mathbf{R}^{-1}\mathbf{u}}, \quad (6.41)$$

where \mathbf{u} is i th column vector of an $N \times N$ identity matrix which corresponds to the i th selected element of the array. The power spectrum can be computed using the expression

$$P_{LP}(\theta) = \frac{\mathbf{u}^H \mathbf{R}^{-1}\mathbf{u}}{|\mathbf{u}^H \mathbf{R}^{-1}\mathbf{a}(\theta)|^2}. \quad (6.42)$$

The LP method's performance is degraded by the presence of noise and usually works well when SNR is high.

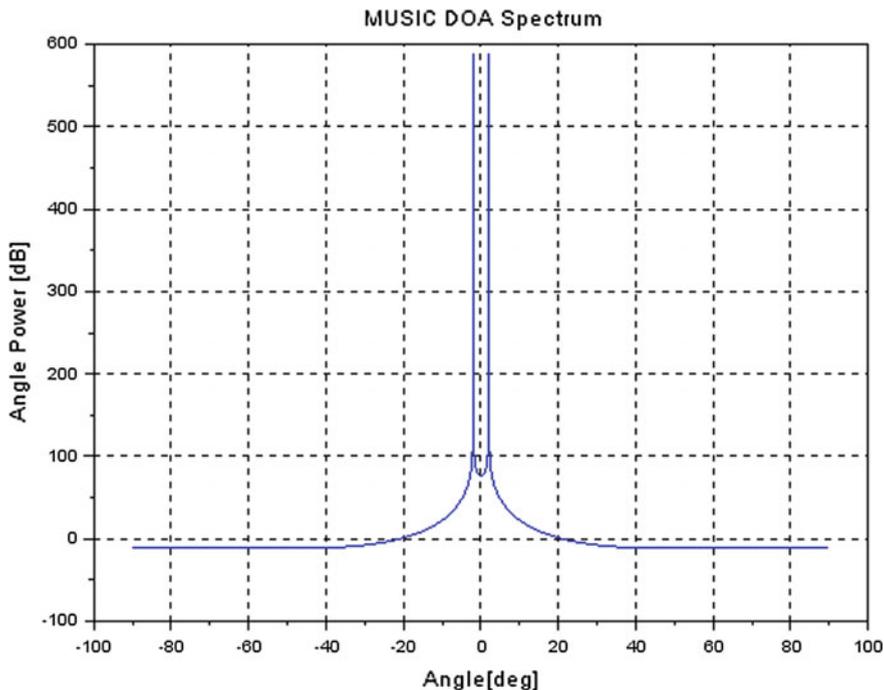


Fig. 6.9 DOA estimation by the MUSIC method with sources two at -2 [deg] and 2 [deg] using a five-element ULA antenna

Propagator Method

The propagator method (PM) is based on a linear operator which can be easily extracted from the receiver data matrix by partitioning the manifold matrix. It is not necessary to apply eigen-decomposition or SVD to the covariance matrix in order to estimate the DOA. The direction of arrival estimation can be formulated as follows. The manifold matrix is partitioned such that we have

$$\mathbf{A} = [\mathbf{A}_1 \ \mathbf{A}_2]^T, \quad (6.43)$$

where \mathbf{A}_1 is an $K \times K$ matrix and \mathbf{A}_2 is an $(N - K) \times K$ matrix. The matrix \mathbf{A}_2 is a linear transformation of \mathbf{A}_1 and it is assumed that \mathbf{A}_1 is non-singular. The relationship between the sub-matrices is given by

$$\mathbf{A}_2 = \mathbf{P}^H \mathbf{A}_1 \quad (6.44)$$

where \mathbf{P} is referred to as the projector matrix. From the projector matrix, a matrix \mathbf{Q} can be constructed such that

$$\mathbf{Q} = [\mathbf{P}^H - \mathbf{I}_{N-P}] \quad (6.45)$$

where \mathbf{I} is identity matrix. Since $\mathbf{Q}^H \mathbf{A} = \mathbf{0}$, a pseudo-spectrum can be extracted as from the relationship below:

$$P(\theta) = \frac{1}{|a(\theta) \mathbf{Q} \mathbf{Q}^H a(\theta)|}. \quad (6.46)$$

As in MUSIC, the peaks in the pseudo-spectrum correspond to angles of arrival of the received signals.

The projection matrix \mathbf{P} is estimated from the covariance matrix $\mathbf{R} = E[\mathbf{X} \mathbf{X}^H]$. The covariance matrix is then partitioned into two matrices such that

$$\mathbf{R} = [\mathbf{R}_1 \ \mathbf{R}_2]. \quad (6.47)$$

In the case of zero-noise received signal, the projection matrix is computed from

$$\mathbf{R}_2 = \mathbf{P}^H \mathbf{R}_1. \quad (6.48)$$

In the presence of noise, which is the normal case, the least squares method can be used to estimate the projection matrix by minimization of the Frobenius norm $\|\mathbf{R}_2 - \mathbf{P}^H \mathbf{R}_1\|_F$. This results in the expression:

$$\mathbf{P}^H = \mathbf{R}_2 (\mathbf{R}_1^H \mathbf{R}_1)^{-1} \mathbf{R}_1^H. \quad (6.49)$$

Weighted Subspace Fitting (WSF)

In subspace methods like MUSIC, the covariance matrix \mathbf{R} has to be estimated from limited data. This results in the violation of the orthogonality assumption between signal and noise subspace. The weighted subspace tries to overcome this limitation by using least squares approach for subspace estimation.

The weighted subspace fitting framework can be expressed as [19, 20]

$$\left[\hat{\mathbf{A}}, \hat{\mathbf{T}} \right] = \underset{\mathbf{A}, \mathbf{T}}{\operatorname{argmin}} \|\mathbf{M} \mathbf{W}^{1/2} - \mathbf{A}(\boldsymbol{\theta}) \mathbf{T}\|_F^2, \quad (6.50)$$

where \mathbf{W} is a positive-definite weighting matrix, the subscript “ F ” denotes the Frobenius norm. After some manipulation, the problem reduces to

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \operatorname{Tr}\{\mathbf{P}_{\mathbf{A}(\boldsymbol{\theta})} \hat{\mathbf{E}}_s \mathbf{W} \hat{\mathbf{E}}_s^H\}, \quad (6.51)$$

where $\mathbf{P}_{\mathbf{A}} = \mathbf{A}(\mathbf{A}^H \mathbf{A})^{-1} \mathbf{A}^H$ is the projection matrix that projects on the column space of \mathbf{A} , Tr denotes trace of the matrix, and $\hat{\mathbf{E}}_s$ is an estimate of the signal subspace eigenvectors. The matrix \mathbf{M} representations of the data can be chosen in different ways leading to different cost functions. \mathbf{T} represents a measure the degree of matching between \mathbf{A} and \mathbf{M} . For example, the choice \mathbf{M} can lead to the WSF,

multi-dimensional MUSIC, weighted ESPRIT, etc. For WSF, the \mathbf{M} is chosen to be $\hat{\mathbf{E}}_s \mathbf{W}_{\text{opt}}^{1/2}$, where the optimum weighting matrix is computed via eigen-decomposition of the covariance matrix into the signal and noise subspaces resulting in the following expression

$$\mathbf{R} = \mathbf{E}_s \mathbf{\Lambda}_s \mathbf{E}_s^H + \mathbf{E}_n \mathbf{\Lambda}_n \mathbf{E}_n^H. \tag{6.52}$$

The estimate of \mathbf{W}_{opt} is given by

$$\mathbf{W}_{\text{opt}} = \left(\hat{\mathbf{\Lambda}}_s - \hat{\sigma}^2 \mathbf{I} \right)^2 \hat{\mathbf{\Lambda}}_s^{-1}, \tag{6.53}$$

where $\hat{\sigma}^2$ is an estimate of the noise variance which can be computed by averaging the noise subspace eigenvalues. For further details, refer to [20].

6.3.5 Multi-dimensional DOA Algorithms

In this chapter, we have mainly focused on one-dimensional (1D) algorithm for DOA estimation. In automotive applications, it is normally of great interest to perform multi-dimensional DOA estimation for a number of reasons. These include among others the increasing necessity to measure the height of on-road objects and road infrastructure such as pedestrian, vehicles and bridges in a dynamic way. Normally, it is sufficient to estimate azimuth DOA by 1D technique outlined in the preceding sections in order to fix the object position. With the need for elevation angle added, it becomes necessary to employ 2D DOA estimation algorithms. There are two approaches available. The first approach is to perform 1D DOA separately in both the azimuth and elevation directions as illustrated in Fig. 6.10. The advantage of this

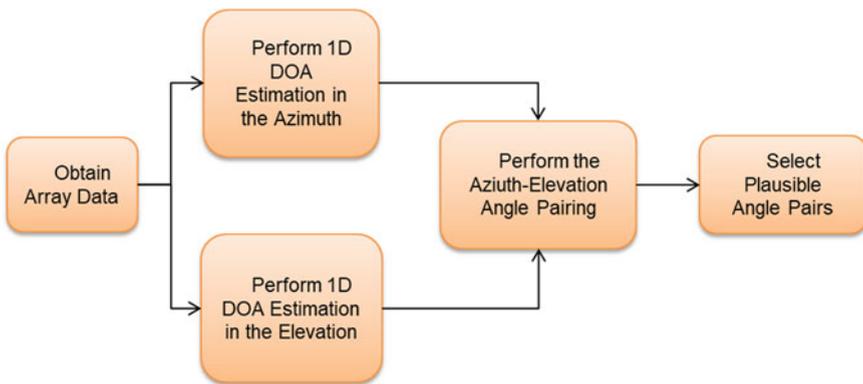


Fig. 6.10 Illustration of 1D approach to 2D DOA estimation

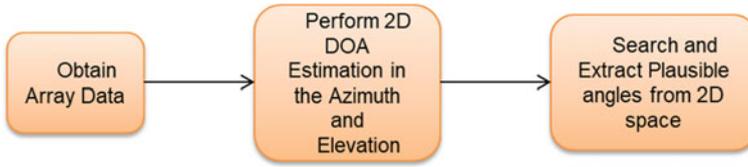


Fig. 6.11 Illustration of 2D approach 2D DOA estimation

approach is that already tested and trusted 1D algorithms can be handily used. The downside is that it becomes necessary to perform pairing of the estimated azimuth and elevation angles, which is not a trivial task and could lead to mispairing and wrong position estimation, which could in turn result to be fatal accidents. On the other hand, the second available option is to use 2D DOA algorithms from which direct extraction of both azimuth and elevation angles is possible without pairing as illustrated in Fig. 6.11. These algorithms already exist in the form of 2D DBF, 2D MUSIC, 2D ESPRIT, etc. The downside is the increased computation load required by the algorithms. The increased computation complicates real-time implementation and antenna design as well. However, given the choice, one would prefer the 2D algorithm and find strategies to optimize computation.

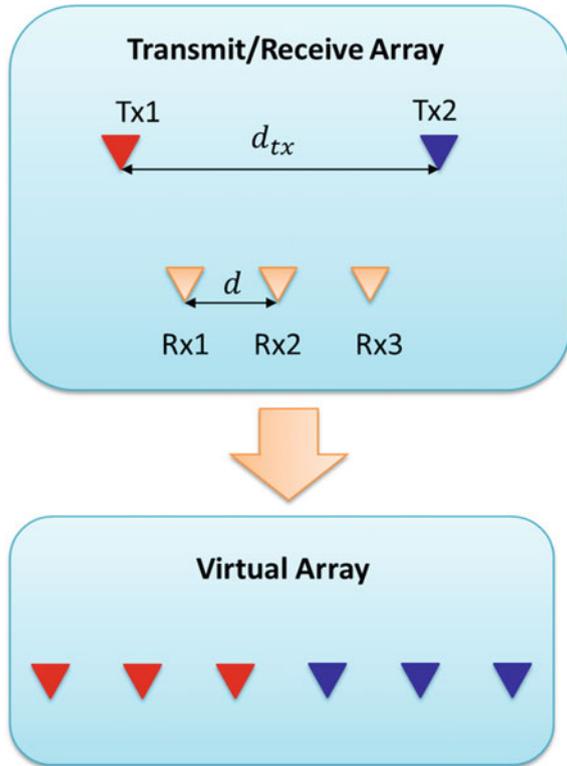
6.3.6 Recent Approaches to Estimation

The number of targets that can be reliably separated is limited by the array aperture, or simply stated, by the number of antenna elements. However, increased aperture increases the size of the antenna, which is undesirable from cost and vehicle integration point of view. To overcome the aperture problem, virtual array processing has been recently studied as a possible solution [21]. The idea is to use multiple transmit antenna to expand the array aperture as illustrated in Fig. 6.12. By appropriate choice of transmit antenna separation based on the receiver antenna element spacing, the number of virtual array antenna elements can be increased to $N_{tx} * N_{rx}$, where N_{tx} is the number of transmit antennas elements and N_{rx} is the number of receive array antenna elements. For a ULA, the transmit antenna spacing is given by $d_{tx} = d * N_{rx}$, where d is the receive antenna spacing. In Fig. 6.12, we get a virtual array of six receive antennas from two transmit antennas and three receive antennas.

The advantage of virtual arrays or MIMO radar as it is sometimes referred to, is the ability to expand the aperture with small-size physical antennas. Applying high-resolution algorithms to the resulting virtual arrays can result in considerable increase in the number of targets that can be separated.

In addition, the virtual array approach can be applied to sparse arrays where the aim is to reduce the number of receiver antenna elements. The missing elements in the sparse array can be inserted as virtual elements.

Fig. 6.12 Illustration of virtual array construction from two transmit and three receive antennas elements to get a six-element virtual array antenna



References

1. Van Trees, H.: Optimum Array Processing Part IV. Wiley-Interscience (2002)
2. Karim, H., Viberg, M.: Two decades of array signal processing research. *IEEE Sign. Process. Mag.* pp. 67–94 (July 1996)
3. Gupta, P., Aditya, K., Datta, A.: Comparison of conventional and subspace based algorithms to estimate Direction of Arrival (DOA). In: 2016 International Conference on Communication and Signal Processing (ICCSP) (6–8 April 2016)
4. Cho S., et al.: A new direction-of-arrival estimation method using automotive radar sensor arrays. *Int. J. Distrib. Sens. Netw.* **13**(6), 1–12 (June 2017)
5. Van Veen, B.D., Buckley, K.M.: Beamforming-A versatile approach to spatial filtering. *IEEE ASSP Mag.* pp. 4–24 (April 1988)
6. Capon, J.: High-resolution frequency-wavenumber spectrum analysis. *Proc. IEEE* **57**, 1408–1418 (1969)
7. Schmidt, R.O.: Multiple emitter location and signal parameter estimation. *IEEE Trans. Antennas Propag.* **34**, 276–280 (1986)
8. Golub, G.H., Van Loan, C.F.: *Matrix Computations*, 3rd edn. Johns Hopkins University Press, Baltimore, MD (1996)
9. Forsythe, K.W.: Utilizing waveform features for adaptive beamforming and direction finding with narrowband signals. *Lincoln Lab. J.* **10**(2), 99–126 (1997)
10. Roy, R., Kailath, T.: ESPRIT—estimation of signal parameters via rotational invariance techniques. *IEEE Trans. Acoust Speech Sign. Process.* **37**, 984–995 (1989)

11. Shan, T.J., Wax, M., Kailath, T.: On spatial smoothing for direction-of-arrival estimation of coherent sources. *IEEE Trans. Acoust. Speech Sign. Process.* **33**(4), 806–811 (August 1985)
12. Pillai, U., Kwon, B.H.: Forward/backward spatial smoothing techniques for coherent signal identification. *IEEE Trans. Acoust. Speech Sign. Process.* **37**(1), 8–15 (January 1989)
13. Munier, J., Delisle, G.Y.: Spatial analysis using new properties of the cross-spectral matrix. *IEEE Trans. Sig. Process* **39**(3), 746–749 (1991)
14. Marcos, S., Marsal, A., Benidir, M.: The propagator method for source bearing estimation. *Sig. Process.* **42**(2), 121–138 (1995)
15. Tufts, D.W., Kumaresan, R.: Estimation of frequencies of multiple sinusoids: making linear prediction perform like maximum likelihood. *Proc. IEEE* **70**, 975–989 (1982)
16. Kay, S.M.: *Modern Spectral Estimation: Theory and Application*. Prentice Hall, Englewood Cliffs, NJ (1988)
17. Gamba, J.: *On Noise-Compensated Techniques for Time Delay Estimation*. Ph.D. Thesis (2005)
18. Makhoul, J.: Linear prediction: a tutorial review. *Proc. IEEE* **63**(4), 561–580 (1975)
19. Paulraj, A., Ottersten, B., Roy, R., Swindlehurst, A., Xu, G., Kailath, T.: Subspace methods for directions-of-arrival estimation. In: Bose, N.K., Rao, C.R. (eds.) *Handbook of Statistics*, vol. 10, pp. 693–739. Elsevier Science Publishers B.V. (1993)
20. Viberg, M., Ottersten, B.: Sensor array processing based on subspace fitting. *IEEE Trans. Sign. Process.* **39**, 1110–1121 (May 1991)
21. Chen, C.-Y., Vaidyanathan, P.P.: Minimum redundancy MIMO radars. In: *2008 IEEE International Symposium on Circuits and Systems*, pp. 45–48 (2008)

Chapter 7

Target Filtering and Tracking



7.1 Introduction

Up to this point, we have presented methods for detection of target properties, specifically, range, velocity, and DOA. Although this information representing instantaneous target state could be the main objective of radar processing, in automotive radar processing, tracking moving targets is of paramount importance. The processing of detected radar targets using filtering and tracking methods for the purpose of capturing target motion dynamics is the goal of this chapter. Among the key methods, Kalman filtering has been widely used as the conventional approach to target tracking [1]. However, Bayesian approaches have recently gained attention due to the fact they address some of the shortcomings of Kalman filtering [2]. We will give an overview of these approaches and also indicate some of the points that present challenges in practical applications. This chapter is not meant to be a substitute for excellent and detailed treatment of these topics that is widely available the literature but to expose the reader to challenges involved in radar tracking. Some more detailed treatment of these topics can be found in [3, 4].

7.2 Kalman Filter

Basic Concepts

The Kalman filter is a computationally efficient recursive filter based on linear dynamic system theory that estimates the state of a discrete-time linear dynamic system from noisy measurements [2, 4, 5]. Since its inception, Kalman filter (KF) has been extensively researched for a wide range of applications.

The KF considers a discrete-time linear dynamic system described by the following process model:

$$\mathbf{x}(k) = \mathbf{F}(k-1)\mathbf{x}(k-1) + \mathbf{G}(k-1)\mathbf{u}(k-1) + \mathbf{v}(k-1), \quad (7.1)$$

where $\mathbf{x}(k)$ is the state vector at time k , $\mathbf{F}(k-1)$ is the state transition matrix, $\mathbf{G}(k-1)$ the input-control matrix, $\mathbf{u}(k-1)$ the control input, and $\mathbf{v}(k-1)$ the zero-mean white Gaussian process noise. The covariance matrix of the process noise at time k is defined by $E(\mathbf{v}(k)\mathbf{v}(k)^H) = \mathbf{Q}(k)$.

The measurement equation is given by

$$\mathbf{z}(k) = \mathbf{H}(k)\mathbf{x}(k) + \mathbf{w}(k), \quad (7.2)$$

where $\mathbf{z}(k)$ is the measurement vector and $\mathbf{w}(k)$ is zero-mean white Gaussian noise. The covariance matrix of the measurement noise is defined by $E(\mathbf{w}(k)\mathbf{w}(k)^H) = \mathbf{R}(k)$.

To put the above into the automotive radar processing perspective, $\mathbf{x}(k)$ is unobserved and has to be estimated using $\mathbf{z}(k)$. $\mathbf{z}(k)$ is obtained from target state measurements, i.e., range, velocity, and angle (DOA). For the filtering and tracking problem, the vector $\mathbf{x}(k)$ consists of target position, velocity, and acceleration in the x - and y -directions. For position only tracking $[x \ y]^T$, $\mathbf{x}(k)$ is two-dimensional vector $[x \ y \ v_x \ v_y]^T$, for position and velocity tracking it is a four-dimensional vector, while for position, velocity, and acceleration tracking, it is a six-dimensional vector $[x \ y \ v_x \ v_y \ a_x \ a_y]^T$.

The matrices \mathbf{F} , \mathbf{G} , \mathbf{Q} , \mathbf{H} , and \mathbf{R} are assumed to be known. Although $\mathbf{G}(k)$ and $\mathbf{u}(k)$ are important in control applications, they are less relevant in automotive applications.

The KF operation requires that the a priori $\mathbf{P}(k|k-1)$ and the a posteriori $\mathbf{P}(k|k)$ estimates of the error covariance matrices based on respective state estimates. The a priori state estimate $\hat{\mathbf{x}}(k|k-1)$ is a state estimate at time k based only on the available measurements up to $k-1$, without taking the current measurement $\mathbf{z}(k)$ into account.

On the other hand, the a posteriori state estimate $\hat{\mathbf{x}}(k|k)$ is an estimate of the system at time k calculated taking into consideration the current measurement $\mathbf{z}(k)$. From these values, the error covariance is estimated.

$$\mathbf{P}(k|k-1) = E[(\mathbf{x}(k) - \hat{\mathbf{x}}(k|k-1))(\mathbf{x}(k) - \hat{\mathbf{x}}(k|k-1))^T] \quad (7.3)$$

$$\mathbf{P}(k|k) = E[(\mathbf{x}(k) - \hat{\mathbf{x}}(k|k))(\mathbf{x}(k) - \hat{\mathbf{x}}(k|k))^T] \quad (7.4)$$

After initialization, the KF computation proceeds in two distinct steps which are prediction and update and described below.

Initialization

Initialize the state estimate and covariance estimates as $\hat{\mathbf{x}}(0)$ and $\hat{\mathbf{P}}(0)$.

Prediction

State prediction (a priori estimate):

$$\hat{\mathbf{x}}(k|k-1) = \mathbf{F}(k-1)\hat{\mathbf{x}}(k-1|k-1) \quad (7.5)$$

Error covariance prediction (a priori estimate):

$$\mathbf{P}(k|k-1) = \mathbf{F}(k-1)\mathbf{P}(k-1|k-1)\mathbf{F}(k-1)^T + \mathbf{Q}(k-1) \quad (7.6)$$

Update

Measurement $\mathbf{z}(k)$ is assumed to be available.

Kalman gain computation:

$$\mathbf{K}(k) = \mathbf{P}(k|k-1)\mathbf{H}(k)^T(\mathbf{H}(k)\mathbf{P}(k|k-1)\mathbf{H}(k)^T + \mathbf{R}(k))^{-1} \quad (7.7)$$

Update state estimate (a posteriori estimate):

$$\hat{\mathbf{x}}(k|k) = \hat{\mathbf{x}}(k|k-1) + \mathbf{K}(k)(\mathbf{z}(k) - \mathbf{H}(k)\hat{\mathbf{x}}(k|k-1)) \quad (7.8)$$

Update error covariance estimate (a posteriori estimate):

$$\mathbf{P}(k|k) = \mathbf{P}(k|k-1) - \mathbf{K}(k)\mathbf{H}(k)\mathbf{P}(k|k-1) \quad (7.9)$$

The optimal Kalman gain $\mathbf{K}(k)$ in the above formulas minimizes the a posteriori error covariance. Detailed derivation can be found in [3, 6]. Postprocessing can be applied to $\hat{\mathbf{x}}(k|k)$ to get smoothed estimates.

The KF algorithm filter outlined above has several shortcomings when applied to radar measurements. The first well-known problem is the assumption that the underlying process is linear. However, not all practical systems obey this linearity requirement. Additionally, the assumption that both process and observations noise are Gaussian is not always true. The Kalman filter also fails to effectively track rapidly accelerating or decelerating targets. This calls for improvements in the underlying assumptions. Based on the above facts, the extended Kalman filter (EKF) and unscented Kalman filter (UKF) have been proposed in the literature [7–12]. In the automotive radar context, nonlinearities can be introduced by the underlying transformation of range and DOA estimates into Cartesian coordinates for position estimation.

Example Consider as an example tracking by Kalman filter using the model $x(t) = 1 + 2t + 2t^2$ to generate 2000-point test position data. The measurement noise variance is assumed to be 0.01, the process noise variance of 0.18. The sampling period is set to 1. The tracking result is shown in Fig. 7.1.

The prediction error is illustrated in Fig. 7.2. For this example, the tracking error can be considered to be small.

Extended Kalman Filter

The EKF models both the state transition and measurement as differentiable functions of the state that are not explicitly linear as follows [8, 9, 11].

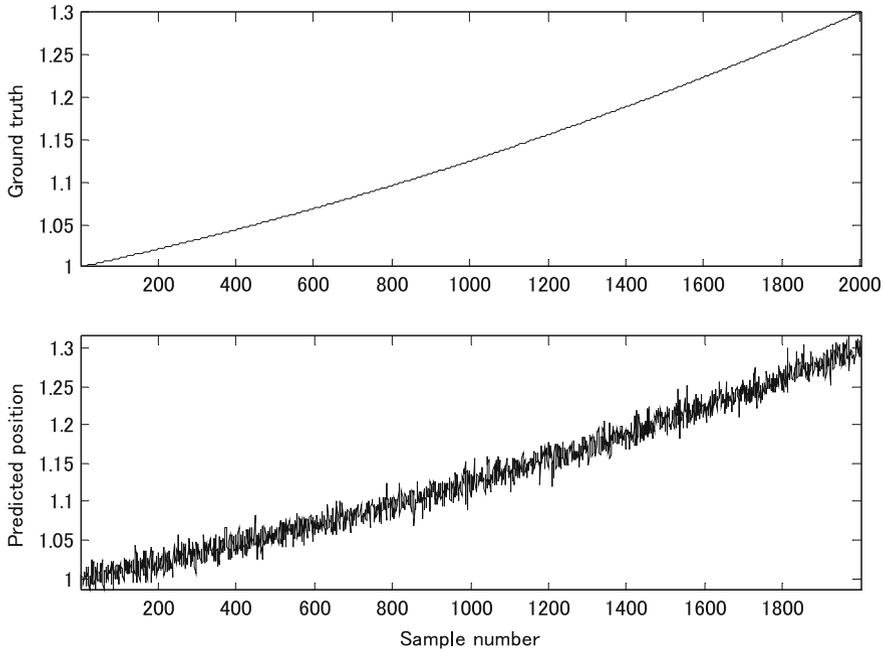


Fig. 7.1 Example of tracking by the Kalman filter

$$\begin{aligned}\mathbf{x}(k) &= f(\mathbf{x}(k-1), \mathbf{u}(k)) + \mathbf{v}(k), \\ \mathbf{z}(k) &= f(\mathbf{x}(k)) + \mathbf{w}(k).\end{aligned}\quad (7.10)$$

Instead of using linear state transition matrix $\mathbf{F}(k)$ in the model, a more general function $f(\mathbf{x}(k-1), \mathbf{u}(k))$ is used. The assumptions on the noise are similar to the KF. The prediction and update steps of the filter proceed as described below.

Prediction

State prediction (a priori estimate):

$$\hat{\mathbf{x}}(k|k-1) = f(\hat{\mathbf{x}}(k-1|k-1), \mathbf{u}(k-1)) \quad (7.11)$$

Error covariance prediction (a priori estimate):

$$\mathbf{P}(k|k-1) = \mathbf{F}(k)\mathbf{P}(k-1|k-1)\mathbf{F}(k)^T + \mathbf{Q}(k-1) \quad (7.12)$$

where $\mathbf{F}(k)$ is defined by the Jacobian matrix of $f(\hat{\mathbf{x}}(k-1|k-1), \mathbf{u}(k-1))$ evaluated at $\hat{\mathbf{x}}(k-1|k-1)$.

$$\mathbf{F}(k) = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}(k-1|k-1)} \quad (7.13)$$

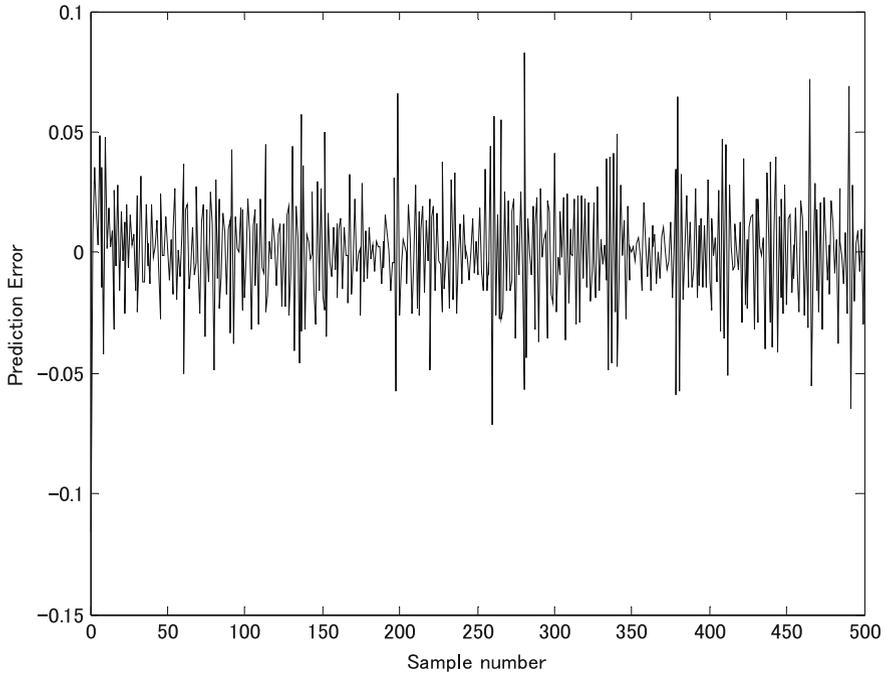


Fig. 7.2 Example of prediction error by the Kalman filter

Update

Measurement $\mathbf{z}(k)$ is assumed to be available.

Kalman gain computation:

$$\mathbf{K}(k) = \mathbf{P}(k|k-1)\mathbf{H}(k)^T(\mathbf{H}(k)\mathbf{P}(k|k-1)\mathbf{H}(k)^T + \mathbf{R}(k))^{-1} \quad (7.14)$$

where $\mathbf{H}(k)$ is defined by the Jacobian matrix of $\mathbf{h}(\hat{\mathbf{x}}(k|k-1))$ evaluated at $\hat{\mathbf{x}}(k|k-1)$

$$\mathbf{H}(k) = \left. \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}(k|k-1)} \quad (7.15)$$

Update state estimate (a posteriori estimate):

$$\mathbf{P}(k|k) = \mathbf{P}(k|k-1) - \mathbf{K}(k)\mathbf{H}(k)\mathbf{P}(k|k-1) \quad (7.16)$$

Although the EKF has been applied to navigation and GPS tracking for a long time, it has known implementation issues. One is that the state estimate is not optimal in the minimum-mean-squared error (MMSE) sense.

Wrong initial state estimates could lead to divergence of the filter. In addition, the computational complexity associated with Jacobians is high for most applications, especially automotive ones. Finally, the Gaussian assumption on process and measurement models still remains, which as for KF could be violated in practice.

Unscented Kalman Filter

The unscented Kalman filter (UKF) is an attempt to deal with problems associated with the EKF while keeping KF ideas in place. It utilizes the nonlinear unscented transformation as a substitute for the linearization operation of the EKF.

Unscented Transformation (UT)

The unscented transformation (UT) is a method for calculating the statistics of a random variable which undergoes a nonlinear transformation [10]. The motivation behind the UT is to use an exact nonlinear function applied to an exact approximating probability distribution of the state instead of a Jacobian approximation used in the EKF. Consider propagating a random variable \mathbf{x} (dimension L) through a nonlinear function, $\mathbf{z} = f(\mathbf{x})$. Assume \mathbf{x} has mean $\bar{\mathbf{x}}$ and covariance \mathbf{P}_x . The next step computes what are referred to as sigma vectors or sigma points (in L -dimensional space) χ_i as follows.

$$\begin{aligned}\chi_0 &= \bar{\mathbf{x}} \\ \chi_i &= \bar{\mathbf{x}} + \left(\sqrt{(L + \lambda)\mathbf{P}_x} \right)_i, \quad i = 1, \dots, L \\ \chi_i &= \bar{\mathbf{x}} - \left(\sqrt{(L + \lambda)\mathbf{P}_x} \right)_{i-L}, \quad i = L + 1, \dots, 2L\end{aligned}\quad (7.17)$$

where $\lambda = \alpha^2(\kappa + L)$ is a scaling factor with α representing the spread of the sigma points around $\bar{\mathbf{x}}$, and κ is a tuning parameter normally set to $3-L$. The notation $(\cdot)_i$ denotes the i th column the matrix. The matrix square root can be computed by Cholesky factorization. The nonlinear function that propagates the sigma points is given by

$$\mathbf{z}_i = f(\chi_i), \quad i = 0, \dots, 2L \quad (7.18)$$

From (7.18), the mean and covariance of \mathbf{z} can be calculated using the following equations.

$$\bar{\mathbf{z}} \cong \sum_{i=0}^{2L} W_i^{(m)} \mathbf{z}_i \quad (7.19)$$

$$\mathbf{P}_z \cong \sum_{i=0}^{2L} W_i^{(c)} (\mathbf{z}_i - \bar{\mathbf{z}})(\mathbf{z}_i - \bar{\mathbf{z}})^T, \quad (7.20)$$

where the weights for computing the mean, $W_i^{(m)}$, and those for computing the covariance, $W_i^{(c)}$, are given by

$$\begin{aligned} W_0^{(m)} &= \frac{\lambda}{L + \lambda}, \\ W_0^{(c)} &= \frac{\lambda}{L + \lambda} + 1 - \alpha^2 + \beta, \\ W_i^{(m)} &= W_i^{(c)} = \frac{\lambda}{2(L + \lambda)}, \quad i = 0, \dots, 2L \end{aligned} \quad (7.21)$$

where β incorporates prior knowledge of distribution and is set to 2 for Gaussian assumption. The UT is applied to the UKF as described below.

System equations:

$$\mathbf{x}(k) = f(\mathbf{x}(k-1), \mathbf{u}(k)) + \mathbf{v}(k), \quad (7.22)$$

$$\mathbf{z}(k) = h(\mathbf{x}(k)) + \mathbf{w}(k). \quad (7.23)$$

Use this for algorithm:

Define a vector of concatenated state and noise vectors as $\mathbf{x}^a(k) = [\mathbf{x}^T(k) \mathbf{v}^T(k)]^T$.

Initialize:

Initialize the state estimate and covariance estimates as $\hat{\mathbf{x}}(0)$ and $\mathbf{P}(0)$.

$$\begin{aligned} \hat{\mathbf{x}}(0) &= E[\mathbf{x}(0)] \\ \mathbf{P}(0) &= E[(\mathbf{x}(0) - \hat{\mathbf{x}}(0))(\mathbf{x}(0) - \hat{\mathbf{x}}(0))^T] \\ \hat{\mathbf{x}}^a(0) &= [\hat{\mathbf{x}}^T(0) E[\mathbf{v}^T(0)]]^T \\ \mathbf{P}^a(0) &= E[(\mathbf{x}^a(0) - \hat{\mathbf{x}}^a(0))(\mathbf{x}^a(0) - \hat{\mathbf{x}}^a(0))^T] \end{aligned} \quad (7.24)$$

Predict:

$$\hat{\mathbf{x}}^a(k-1) = [\hat{\mathbf{x}}^T(k-1) E[\mathbf{v}^T(k-1)]]^T \quad (7.25)$$

$$\mathbf{P}^a(k-1) = \begin{bmatrix} \mathbf{P}(k-1) & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}(k-1) \end{bmatrix} \quad (7.26)$$

Generate sigma points

$$\chi_0(k-1) = \hat{\mathbf{x}}^a(k-1)$$

$$\chi_i(k-1) = \hat{\mathbf{x}}^a(k-1) + \left(\sqrt{(L + \lambda)\mathbf{P}^a(k-1)} \right)_i, \quad i = 1, \dots, L$$

$$\chi_i(k-1) = \hat{\mathbf{x}}^a(k-1) - \left(\sqrt{(L+\lambda)\mathbf{P}^a(k-1)} \right)_{i-L}, \quad i = L+1, \dots, 2L \quad (7.27)$$

The weights $W_i^{(m)}$ and $W_i^{(c)}$ are computed as described above.

Propagate the sigma points, state, and covariance:

$$\begin{aligned} \chi_i(k|k-1) &= f(\chi_i(k-1)), \quad i = 0, \dots, 2L \\ \hat{\mathbf{x}}(k|k-1) &= \sum_{i=0}^{2L} W_i^{(m)} \chi_i(k|k-1) \\ \mathbf{P}(k|k-1) &= \sum_{i=0}^{2L} W_i^{(c)} \left[(\chi_i(k|k-1) - \hat{\mathbf{x}}(k|k-1))(\chi_i(k|k-1) - \hat{\mathbf{x}}(k|k-1))^T \right] \end{aligned} \quad (7.28)$$

Update:

$$\begin{aligned} \mathbf{x}^a(k|k-1) &= [\hat{\mathbf{x}}^T(k|k-1)E[\mathbf{v}^T(k)]]^T \\ \mathbf{P}^a(k|k-1) &= \begin{bmatrix} \mathbf{P}(k|k-1) & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}(k) \end{bmatrix} \\ \chi_0(k|k-1) &= \mathbf{x}^a(k|k-1) \\ \chi_i(k|k-1) &= \mathbf{x}^a(k|k-1) + \left(\sqrt{(L+\lambda)\mathbf{P}^a(k|k-1)} \right)_i, \quad i = 1, \dots, L \\ \chi_i(k|k-1) &= \mathbf{x}^a(k|k-1) - \left(\sqrt{(L+\lambda)\mathbf{P}^a(k|k-1)} \right)_{i-L}, \quad i = L+1, \dots, 2L \end{aligned} \quad (7.29)$$

Project the sigma points through the measurement model:

$$\boldsymbol{\gamma}_i(k) = \mathbf{h}(\chi_i(k|k-1)), \quad i = 0, \dots, 2L \quad (7.30)$$

From (7.30), compute the predicted measurement and its covariance.

$$\hat{\mathbf{z}}(k) = \sum_{i=1}^{2L} W_i^{(m)} \boldsymbol{\gamma}_i(k), \quad (7.31)$$

$$\mathbf{P}_{z(k)z(k)} = \sum_{i=1}^{2L} W_i^{(c)} (\boldsymbol{\gamma}_i(k) - \hat{\mathbf{z}}(k))(\boldsymbol{\gamma}_i(k) - \hat{\mathbf{z}}(k))^T. \quad (7.32)$$

Compute the cross-covariance for the computation Kalman gain.

$$\mathbf{P}_{\mathbf{x}(k)z(k)} = \sum_{i=1}^{2L} W_i^{(c)} (\chi_i(k|k-1) - \hat{\mathbf{x}}(k|k-1))(\boldsymbol{\gamma}_i(k) - \hat{\mathbf{z}}(k))^T \quad (7.33)$$

Kalman gain:

$$\mathbf{K}(k) = \mathbf{P}_{x(k)z(k)} \mathbf{P}_{x(k)z(k)}^{-1} \quad (7.34)$$

State update:

$$\hat{\mathbf{x}}(k|k) = \hat{\mathbf{x}}(k|k-1) + \mathbf{K}(k)(\mathbf{z}(k) - \hat{\mathbf{z}}(k)) \quad (7.35)$$

Covariance update:

$$\mathbf{P}(k|k) = \mathbf{P}(k|k-1) - \mathbf{K}(k) \mathbf{P}_{z(k)z(k)} \mathbf{K}(k)^T \quad (7.36)$$

Other KF Variants

Other variants of the KF such as the **ensemble KF** and **fast Kalman filter** exist, but the current state of the art in terms of Kalman filtering is the UKF. The UKF has been shown to perform better than its predecessor, EKF, but still has issues related to parameter selection. In some cases, better performance by EKF has been reported [13].

7.3 Bayesian Filtering (Sequential Monte Carlo)

Basic Concepts

Particle Filter

When the system dynamics and measurement models are linear, the KF gives the optimal minimum-mean-squared error (MMSE) state estimate. However, the KF is not suitable for nonlinear/non-Gaussian systems. To circumvent this limitation, EKF and UKF have been proposed [7–12]. However, the EKF has already been shown to be difficult to implement [10], and thus, the UKF is the recommended practical solution for tracking and navigation systems. Although the UKF provides a better KF-based performance, it still relies on base KF assumptions. To deal completely with nonlinear/non-Gaussian systems, an alternative approach is required. This is the motivation behind Bayesian filtering approaches and in particular the particle filter [2]. The driving factor comes from the known and painful fact that in most real situations, the desired linear/Gaussian model is rarely obeyed by systems under consideration. The Bayesian approach relaxes the assumptions on system dynamics and thus leads to wider application fields covering linear/nonlinear and Gaussian/non-Gaussian processes. In this section, we give an overview of the Bayesian approach to filtering with emphasis on automotive applications. Excellent treatment of this topic in detail can be found in [4, 14].

Recursive Bayesian Estimation

The Bayesian filtering approach recursively generates an approximation of the state probability density function using a set of random samples as opposed to the functional approaches of EKF and UKF [2]. The system model can be expressed as

$$\mathbf{x}(k) = f(\mathbf{x}(k-1), \mathbf{v}(k-1)), \quad (7.37)$$

where $f(\cdot)$ is a transition function and $\mathbf{v}(\cdot)$ are zero-mean, white noise of known PDF.

The measurement equation is given by

$$\mathbf{z}(k) = h(\mathbf{x}(k), \mathbf{w}(k)), \quad (7.38)$$

where $h(\cdot)$ is a transition function and $\mathbf{w}(\cdot)$ are zero-mean, white noise of known PDF. The PF algorithm approximates the posterior PDF $p(x(k)|z(1:k))$ by a set of weighted random samples, referred to as particles. The initial prior distribution of the state $p(\mathbf{x}(0))$ is assumed to be known. It is also assumed that the probability density function $p(x(k-1)|z(1:k-1))$ at previous time instant $k-1$ is available. The PDF $p(x(k)|z(1:k))$ can be computed as follows:

$$p(x(k)|z(1:k)) = \int p(x(k)|x(1:k-1))p(x(k-1)|z(1:k-1))dx(k-1) \quad (7.39)$$

The prior (or prediction) can be updated using the current measurement $y(k)$ based on Bayes theorem:

$$p(x(k)|z(1:k)) = \frac{p(y(k)|x(k))p(x(k)|z(1:k-1))}{p(z(k)|z(1:k-1))} \quad (7.40)$$

where $p(y(k)|z(1:k-1))$ is considered as normalizing constant (independent of state). The above Eqs. (7.37)–(7.40) form the basis of optimal Bayesian estimation. Using the MMSE criteria as an example, the optimal state estimate can be obtained as

$$E[x(k)|z(1:k)] = \int x(k)p(x(k)|z(1:k))dx(k). \quad (7.41)$$

Alternatively, the maximum a posterior (MAP) estimate that maximizes $p(x(k)|z(1:k))$ can be computed.

The biggest drawback of the above approach is that the indefinite integrals are computationally intractable, and hence, approximations are necessary. These approximations include the sequential Monte Carlo methods (SMC) that form the basis of the particle filter.

The Particle Filter (PF)

The PF represents the posterior PDF by using a set of random samples (particles) $x_i(k)$, $i = 1, \dots, N$, where N is the number of particles, associated with weights $w_i(k)$ in order to obtain state estimates [14–21]. In most case, importance sampling in conjunction with a resampling strategy is utilized to determine the weights [4]. In particular, some of the resampling strategies include multinomial resampling, stratified resampling, systematic resampling, and residual resampling [19, 20].

The weights are normalized such that their sum equal to 1. If it is assumed that samples can be generated from a density $q(x)$ that is similar to $p(x)$, then the posterior density is approximated by

$$p(x(k)|z(1:k)) \cong \sum_{i=1}^N w_i(k) \delta(x(k) - x_i(k)), \quad (7.42)$$

where $\delta(x)$ is the Dirac delta function and

$$w_i(k) \propto \frac{p(x_i(k)|z(1:k))}{q(x_i(k)|z(1:k))} = w_i(k-1) \frac{p(z(k)|x_i(k))p(x_i(k)|x_i(k-1))}{q(x_i(k)|x_i(k-1), z(k))}, \quad (7.43)$$

where $p(z(k)|x(k))$ is the likelihood function defined by the measurement model, and $p(x_i(k)|x_i(k-1))$ is a transitional function of the process model.

A common problem with PF is that of particle degeneracy where the weight is concentrated in one particle, with other particles having weights close to zero and thus contributing nothing to the posterior distribution. The degeneracy is measured by effective sample size (ESS) given by

$$N_{\text{eff}} = \frac{1}{\sum_{i=1}^N w_i^2(k)}. \quad (7.44)$$

Resampling strategies use N_{eff} by setting a threshold such that resampling is performed whenever N_{eff} falls below the threshold.

The bootstrap PF algorithm can be summarized as follows.

Bootstrap Filter:

The bootstrap filter is the most basic PF algorithm. It can be described by the following steps.

Step 1: Initialization, $k = 0$

For $i = 1, \dots, N$, sample $\mathbf{x}_i(0) \sim p(\mathbf{x}(0))$.

Step 2: Importance sampling

For $i = 1, \dots, N$:

- (i) Sample $\hat{\mathbf{x}}_i(k) \sim p(\mathbf{x}_i(k)|\mathbf{x}_i(k-1))$ and make $\hat{\mathbf{x}}_i(0:k) = [\hat{\mathbf{x}}_i(0:k-1), \hat{\mathbf{x}}_i(k)]$.

(ii) Compute importance weights

$$\hat{w}_i(k) = p(z(k)|\hat{x}_i(k)) \tag{7.45}$$

(iii) Normalize the weights

$$\hat{w}_i(k) = \frac{\hat{w}_i(k)}{\sum_{i=1}^N \hat{w}_i(k)} \tag{7.46}$$

Step 3: Resampling

(i) Compute the effective sample size (ESS) given by

$$N_{\text{eff}} = \frac{1}{\sum_{i=1}^N w_i^2(k)} \tag{7.47}$$

(ii) If N_{eff} is less than a predetermined threshold, resample with replacement N particles $[x_i(0:k), i = 1, \dots, N]$ set $(\hat{x}_i(0:k), i = 1, \dots, N)$ according to the importance weights.

Step 4: Increment to the time step and repeat the procedure from Step 2.

For resampling step (Step 3), methods detailed in [19, 20] can be applied. Among these methods, systematic resampling is the simplest and commonly used method. Figure 7.3 illustrates computations performed in the bootstrap filter.

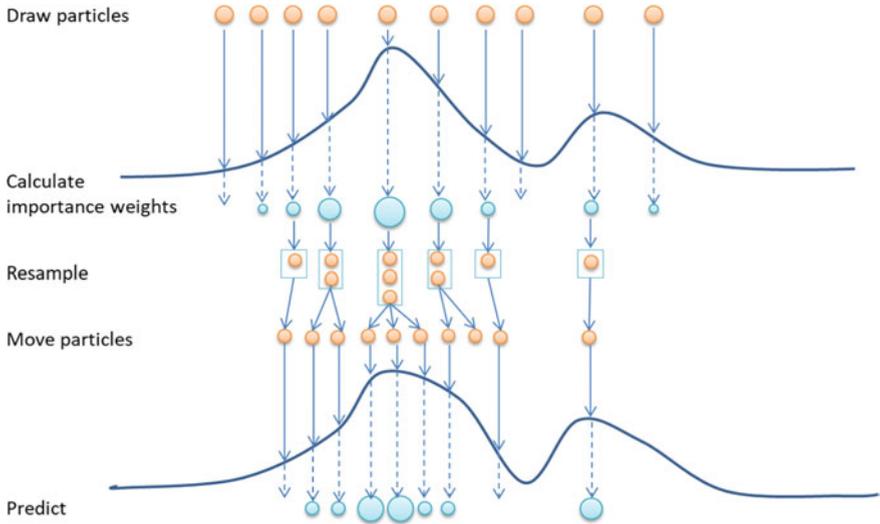


Fig. 7.3 Illustration of particle computation flow using $N = 10$ particles

Variants of PF:

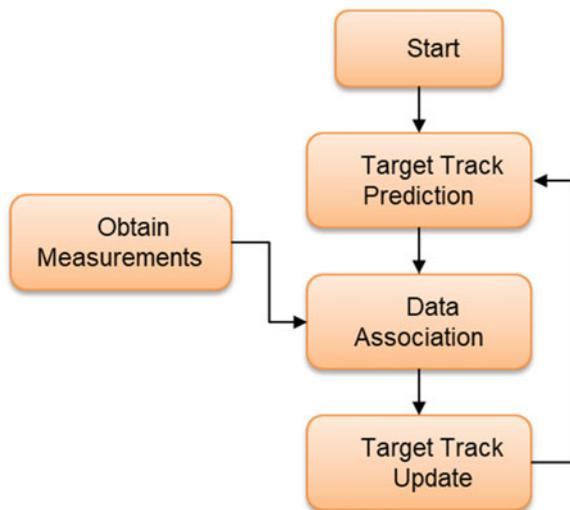
Some of the variants of the PF include the auxiliary PF [4], Rao-Blackwellized [21], parallel partitioned particle filter [18], and unscented PF (UPF) [2, 17]. These approaches and others provide a glimpse of the level of research activity related to PFs.

7.4 Data Association in Multi-target Tracking

In multi-target tracking environments, where new targets appear and disappear during the observation period, data association is an important topic [3, 22]. In automotive radar tracking, associating measurements with targets is more challenging due to the presence of obstacles (clutter), false alarms, ambiguous measurements, and in some cases loss of continuous measurements. By definition, the main task of data association is to associate measurements with either existing targets or new targets. The general flow of multi-target processing is shown in Fig. 7.4. Using initial track or existing data, the prediction is performed so that the expected measurement data space can be determined. With the availability of new measurements, data association can be carried out.

When the association with existing targets is decided, the existing targets tracks are updated. On the other hand, when no targets can be associated with measurements, new target tracks are created. In the process, targets that cease to exist have to be removed from the tracking list. As an example, suppose we have two target tracks and three measurements are validated. Then, the relationship between target tracks and measurements can be illustrated in Fig. 7.5.

Fig. 7.4 General flow of multi-target processing



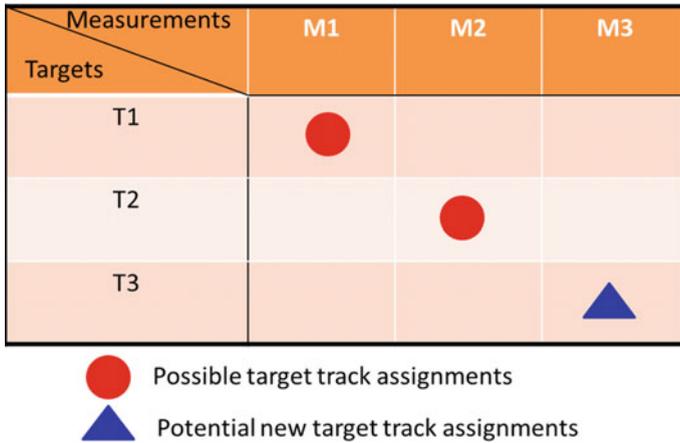


Fig. 7.5 Relationship between target tracks and measurements

Various methods have been proposed in the literature in order to achieve the above task of data association. These include nearest neighbors, generalized nearest neighbors (GNN), probabilistic data association (PDA), multiple hypothesis testing (MHT), assignment algorithm, etc. Detailed treatment of these topics is beyond the scope of this section, but we will instead give an outline of some of the main methods that can be considered for automotive radar tracking.

Generalized Nearest Neighbor (GNN)

The GNN uses a distance measure, such as the Mahalanobis distance or Euclidean distance, to associate tracks with measurements. This leads to a minimization problem of the form

$$\text{minimize } \sum d_{jk}^2 w_{jk} \text{ subject to } \sum_j w_{jk} = 1, \sum_k w_{jk} = 1,$$

where d_{jk}^2 is the distance measure between track j and measurement k , and w_{jk} can be considered as associated weights.

The advantage of the GNN approach is that it is very simple to implement. The GNN shows poor performance when ambiguities exist in the measurements and sometimes lead to local minima.

Joint Probabilistic Data Association Filter (JPDAF)

The PDAF is used for single target tracking where for each valid measurement, the probability β_i that it comes from the considered track is computed. The computation of β_i incorporates the probability of detection and false alarm rate. Using the computed probabilities, the target data can be updated. The JPDAF is an extension of this single target concept to multiple target tracking. In this case, the probability β_{ij}

that a measurement i comes from target track j is computed. Details of this approach can be found in [22].

Multiple Hypothesis Testing

The multiple hypothesis filter generates a hypothesis tree by considering three possible associations where a measurement either belongs to an existing track or is due to false alarm or is new. Based on Bayes rule, the probability of each hypothesis can be evaluated. Finally, the likelihood of each possible association is computed [3]. Although it can be considered as the most general approach to data association, the MHT approach is computationally expensive and complex to implement in real time. Additionally, only approximate probabilities can be computed increasing the possibilities of tracking errors.

Assignment Algorithms

The assignment algorithms originated from operations research where the objective is to optimize in terms of cost the assignment of jobs to available resources. The optimum solution is one that minimizes the total cost. The Hungarian algorithm is one of the representatives and well-known algorithm that solves the assignment problem. It was originally developed by Kuhn [23] and the further refined James Munkres to become the Munkres algorithm [24]. Other algorithms that address the same problem include Murty's algorithm [25] and Jonker–Volgenant algorithm [26]. Since the Hungarian algorithm is well understood and easily applicable to the target assignment problem in data association, we will give a brief description of the steps involved in the algorithm followed by a simple example.

The Hungarian algorithm is used to find an optimum target track assignment for a given cost matrix. The cost matrix consists of elements C_{ij} , which represent the cost of assigning measurement i to target track j . For the Hungarian algorithm, the cost matrix C must be a square matrix. The elements of the cost matrix are normally computed from the likelihood function. These exact values of the likelihood function are determined taking into consideration sensor properties such as measurement noise, probabilities of detection and false alarms, density functions for track initialization, and the likelihood that a measurement among other tracking algorithm and target characteristics [26].

Based on an $N \times N$ cost matrix, the Hungarian algorithm proceeds according to the following steps [27, 28]:

- (1) **Row reduction:** Subtract the smallest entry in each row from all the entries of its row.
- (2) **Column reduction:** Subtract the smallest entry in each column from all the entries of its column.
- (3) **Zero assignment:** Draw lines through appropriate rows and columns so that all the zero entries of the cost matrix are covered and the minimum number of such lines is used.
- (4) **Test for Optimality:** (i) If the minimum number of covering lines is N , an optimal assignment of zeros is possible and we are finished. (ii) If the minimum

number of covering lines is less than n , an optimal assignment of zeros is not yet possible. In that case, proceed to step (5).

- (5) Determine the smallest entry not covered by any line. Subtract this entry from each uncovered row, and then, add it to each covered column. Return to step (3).

Example Considering an idea case where we have three valid measurements and three valid target tracks and the following cost matrix, C .

$$C = \begin{bmatrix} 0.81 & 0.91 & 0.28 \\ 0.90 & 0.63 & 0.55 \\ 0.13 & 0.09 & 0.95 \end{bmatrix} \quad (7.48)$$

Applying the above steps in the cost matrix, C , we obtain the matching matrix M

$$M = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad (7.49)$$

where 1 indicated the assignment. If the rows represent target tracks and columns represent measurements, then measurement 3 is assigned target 1, measurement 2 is assigned target 2, and measurement 1 is assigned target 3. The minimum cost can be computed as $\text{Cost} = 0.13 + 0.63 + 0.28 = 1.04$.

The same idea can be extended to more than three targets and measurements. A lot of practical issues arise from this seemingly simple and straight forward approach. The first is that the number of observations or measurements are not always equal to the number of tracks. One reason could be missing detections. Another is a new detections. It is also possible to have multiple detections for a single target and so forth.

When there are more measurements than target tracks or vice versa, the cost matrix is first converted to a square matrix before the matching is performed.

When missing data or noisy measurements are obtained, the algorithm has to be modified by, for example, using gating functions or thresholding methods.

7.5 Challenges in Target Filtering and Tracking

The particle filter offers some of the best filtering results when dynamics and statistics of the process under investigation are not accurately known. As outlined in the introduction of this section, it results in better target tracking in many applications. This is also true for automotive radar tracking where sudden maneuvers can present serious challenges to the KF approaches. One of the challenges associated with the practical use of the PF includes the choice of the proposal distribution for importance sampling which is non-trivial. An effective approach that has shown promising results is UPF,

which applies the UKF to generate the proposal distribution [21]. However, the computational complexity is excessive for most automotive target tracking applications. It also carries the baggage of the UKF to particle filtering, thereby further complicating initialization and parameter selection. On more issue of practical significance is data association. Although various methods are available for consideration, accuracy and computational load problems arise among other challenges. In most cases, algorithm parameters have to be tuned in order to be useful to the available data. Although not peculiar to PF, target state update and data association present a huge percentage of the filter processing making it necessary to find innovative approaches. However, for pedestrian detection, the PF has recently been seen as one of the feasible approaches [29, 30]. This could be attributed to its ability to track nonlinear dynamics.

References

1. Kalman, R.E.: A new approach to linear filtering and prediction problems. *J. Basic Eng.* **82**(Series D), 35–45 (1960)
2. Gordon, N.J., Salmond, D.J., Smith, A.F.M.: Novel approach to nonlinear/non-Gaussian Bayesian state estimation. In: *Radar and Signal Processing*, IEE Proceedings F, vol. 140, issue 2, pp. 107–113, Apr 1993
3. Bar-Shalom, Y., Li, X., Kirubarajan, T.: *Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software*. Wiley (2002)
4. Doucet, A., de Freitas, N., Gordon, N. (eds.): *Sequential Monte Carlo Methods in Practice* (2001)
5. Welch, G., Bishop, G.: *An Introduction to the Kalman Filter*. University of North Carolina at Chapel Hill, Chapel Hill, NC (2006)
6. Haykin, S.: *Adaptive Filter Theory*, 3rd edn. Prentice-Hall, Inc., Upper Saddle River, NJ, USA (1996)
7. Julier, S.J., Uhlmann, J.K.: Unscented filtering and nonlinear estimation. *Proc. IEEE* **92**(3), 401–422 (2004)
8. Julier, S.J., Uhlmann, J.K., Durrant-Whyte, H.: A new approach for filtering nonlinear systems. In: *Proceedings of the American Control Conference*, pp. 1628–1632 (1995)
9. Julier, S.J., Uhlmann, J.K.: A general method for approximating nonlinear transformations of probability distributions. Technical Report, RRG, Department of Engineering Science, University of Oxford, Nov 1996
10. Julier, S.J., Uhlmann, J.K.: A new extension of the Kalman filter to nonlinear systems. In: *Proceedings of AeroSense: The 11th International Symposium on Aerospace/Defence Sensing, Simulation and Controls* (1997)
11. Julier, S.J., Uhlmann, J.K., Durrant-Whyte, H.: A new method for the nonlinear transformation of means and covariances in filters and estimators. *IEEE Trans. Autom. Control* **45**(3), 477–482 (2000)
12. Wan, E.A., van der Merwe, R.: The Unscented Kalman Filter. In: *Kalman Filtering and Neural Networks*. Wiley (2001)
13. LaViola, J.J.: A comparison of unscented and extended Kalman filtering for estimating quaternion motion. In: *Proceedings on American Control Conference*, vol. 3, pp. 2435–2440 (2003)
14. Andrieu, C., Doucet, A., Holenstein, R.: Particle Markov chain Monte Carlo methods. *J. R. Stat. Soc. Ser. B* **72**(Part 3), 269–342 (2010)
15. Doucet, A., Johansen, A.M.: A tutorial on particle filtering and smoothing: fifteen years later. In: *Crisan, D., Rozovsky, B. (eds.) Handbook of Nonlinear Filtering*. Cambridge University Press, Cambridge (2009)

16. Gustafsson, Fredrik: Particle filter theory and practice with positioning applications. *IEEE Aerosp. Electron. Syst. Mag.* **25**(7), 53–82 (2010)
17. van der Merwe, R., Doucet, A., de Freitas, J.F.G., Wan, E.: The unscented particle filter. Technical Report CUED/F-INFENG/TR 380, Cambridge University Engineering Department (2000)
18. Gamba, J., Yamano, C.: Target tracking using parallel partition particle filtering. In: Society Conference of IEICE, Hokkaido University (Sapporo), Japan, 20–23 Sept 2005
19. Hol, J.D., Schon, T.B., Gustafsson, F.: On resampling algorithms for particle filters. In: 2006 IEEE Nonlinear Statistical Signal Processing Workshop, 13–15 Sept 2006
20. Li, T., Bolic, M., Djuric, P.M.: Resampling methods for particle filtering: classification, implementation, and strategies. *IEEE Signal Process. Mag.* **32**(3), 70–86 (2015)
21. Särkkä, S., Vehtari, A., Lampinen, J.: Rao-Blackwellized particle filter for multiple target tracking. *Inf. Fusion* **8**(1), 2–15 (2007)
22. Kirubarajan, T., Bar-Shalom, Y.: Probabilistic data association techniques for target tracking in clutter. *Proc. IEEE* **92**(3), 536–557 (2004)
23. Kuhn, H.W.: The Hungarian method for the assignment problem. *Naval Res. Logistics Q.* **2**(1–2), 83–97 (1955)
24. Munkres, J.: Algorithms for assignment and transportation problems. *J. Soc. Ind. Appl. Math.* **5**(1), 32–38 (1957)
25. Murty, K.G.: An algorithm for ranking all the assignments in order of increasing cost. *Oper. Res.* **16**(3), 682–687 (1968)
26. Jonker, R., Volgenant, T.: A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing* **38**(11), 325–340 (1987)
27. Poore, A.B., Gadaleta, S.: Some assignment problems arising from multiple target tracking. *Math. Comput. Model.* **43**, 1074–1091 (2006)
28. The Assignment Problem and the Hungarian Method. http://www.math.harvard.edu/archive/20_spring_05/handouts/assignment_overheads.pdf
29. Heuer, M., Al-Hamadi, A., Rain, A., Meinecke, M., Rohling, H.: Pedestrian tracking with occlusion using a 24 GHz automotive radar. In: 2014 15th International Radar Symposium (IRS), 16–18 June 2014
30. Gaddigoudar, P.K., Balihalli, T.R., Ijantkar, S.S., Iyer, N.C., Maralappanavar, S.: Pedestrian detection and tracking using particle filtering. In: 2017 International Conference on Computing, Communication and Automation (ICCCA), pp. 110–115 (2017)

Chapter 8

Target Recognition and Classification Techniques



8.1 Introduction

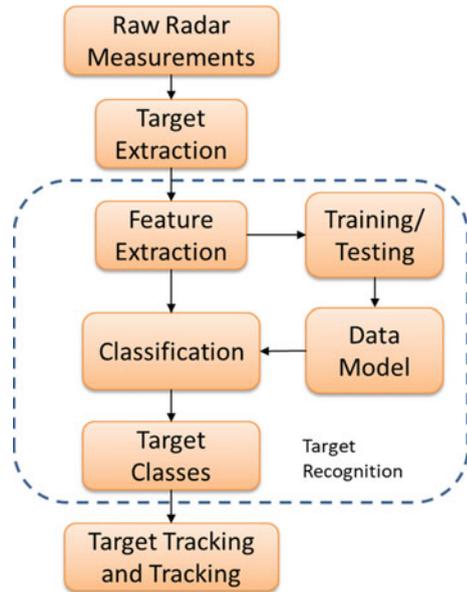
Target recognition is increasingly becoming an important part of radar processing for automotive applications [1]. The reason for this development is that the environment in which the automotive radar operates is highly cluttered which makes it essential to distinguish targets of interest with a high degree of precision. Additionally, there is a growing demand to improve the ability to recognize pedestrians on the roads [2–4]. For the above reasons, target recognition could be utilized to distinguish and classify detected objects. After the recognition is performed, it becomes possible to optimize signal processing algorithms for a particular type of target. For example, different tracking and filtering algorithms could be applied to for passenger cars and pedestrians. The general flow of radar processing involving target recognition is depicted in Fig. 8.1. In order to recognize targets, various machine learning algorithms are available. There are two major categories of machine learning: supervised machine learning and unsupervised machine.

A supervised machine learning algorithm uses labeled input data to learn a mapping function which generates an appropriate output when given new unlabeled data. The term supervised learning comes from the fact that the process of algorithm learning uses training dataset can be viewed as an instructor supervising the learning process. Supervised learning can be divided into classification and regression. The classification process results in discrete or categorized outputs such as car, bicycle, pedestrian, or truck. The output class can be labeled as an integer. On the other hand, regression results in real-valued outputs such as height or width.

An unsupervised machine learning algorithm utilizes input data without using explicitly provided labels. This is in contrast to supervised learning where training data is required.

In this chapter, we explore/summarize some of the algorithms that can be used for target recognition and classification, mainly focusing on supervised learning. It is not the intention of this chapter to cover all machine learning algorithms but to provide to the reader an insight into some of the methods that may find utility in

Fig. 8.1 Signal flow of radar processing involving target recognition



automotive radar processing. There is a vast amount of reference material online and in the literature on recognition/classification that is applicable to statistics, image processing, among other fields [5–7].

8.2 Methods of Target Recognition

K-Nearest Neighbors (KNN)

The k-nearest neighbors (KNN) algorithm is a very well-established, simple and widely used nonparametric supervised machine learning algorithm that can be applied to both classification and regression problems [8]. Figure 8.2 illustrates the concept of KNN. It uses a distance metric, such as the Euclidean distance, to classify given data.

For classification, the KNN can be performed by the following simple steps.

1. Set the parameter K , which is the number of nearest neighbors.
2. Calculate the distance between the query instance and all the training sample data.
3. Sort the distances in ascending order, keeping the indices from the training sample data.
4. Select the first K entries from the sorted distances.
5. Extract the labels of the selected K entries.
6. Calculate the mode of the K labels.

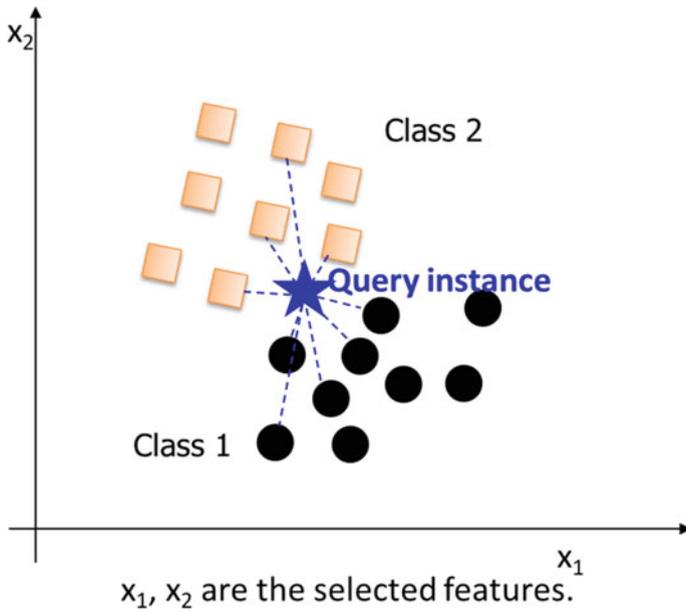


Fig. 8.2 Illustration of the concept of KNN shows the calculation of distances for some selected samples. Distance calculation must be performed for all samples

The KNN algorithm is advantageous for its simplicity, places no assumptions on data, has relatively high accuracy, and can be used for both classification and regression. However, on the downside, the KNN requires storage of all training data which could become expensive in terms of computational complexity and memory requirements for large datasets. It is also known to be sensitive to noise, which could be challenging for automotive radar data. Below we give a simple example to illustrate the KNN algorithm.

Example Consider two class problems with Class 1 and Class 2 data centered around $(x_1, x_2) = (2, 3)$ and $(x_1, x_2) = (3, 4)$, respectively. Using this as training data, we wish to classify query points $p_1 = (2.4, 3.4)$ and $p_2 = (2.8, 3.8)$. This is illustrated in Fig. 8.3 where the Classes 1 and 2 are shown as red squares and black circles, respectively, while query points are marked in blue (asterisk). These query points are classified by KNN. The result of using $K = 3$ is shown in Fig. 8.4.

As shown in Fig. 8.4, the KNN algorithm is able to classify the query data correctly, i.e., the lower query point with Class 1 and the upper query point with Class 2. Problems arise when classes are not well separated, which could result in misclassification.

Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA)

In general, there are three classes of discriminant functions which result in three corresponding methods. LDA is based on a linear discriminant function; quadratic

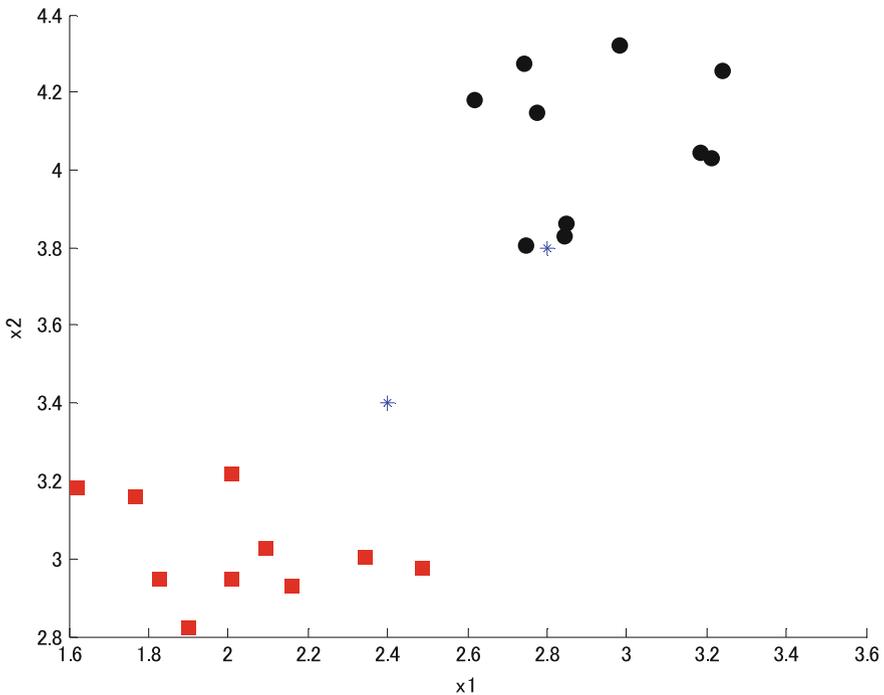


Fig. 8.3 Data used in the example to illustrate KNN classification

discriminant analysis (QDA) is based on a second-order discriminant function, while nonlinear discriminant analysis (NDA) is based on a nonlinear discriminant function. We will mainly focus on LDA and QDA since they are commonly used in practice.

Linear Discriminant Analysis (LDA)

LDA is one of the simplest and well-known methods of classifying objects. It was originally proposed by Fisher [9] and is therefore also referred to as Fisher's linear discriminant analysis in the literature. Basically, LDA operates on the principle of orthogonality as shown in Fig. 8.5. It is a linear classifier that can be used to separate two or more classes from observed data. LDA is especially known to work very effectively for two-class problems although extension to more than two classes is possible giving rise to what is referred to as multiple discriminant analysis.

Based on Bayes' classifier, the LDA discriminant function maximizes the probability of assigning a class given feature measurements.

$$f(x) = \operatorname{argmax}_{j=1,\dots,K} P(C = j | X = x) \quad (8.1)$$

C is the class variable, and x represents feature measurements. The idea is to select the class j which maximizes the conditional probability. The LDA assumes that the

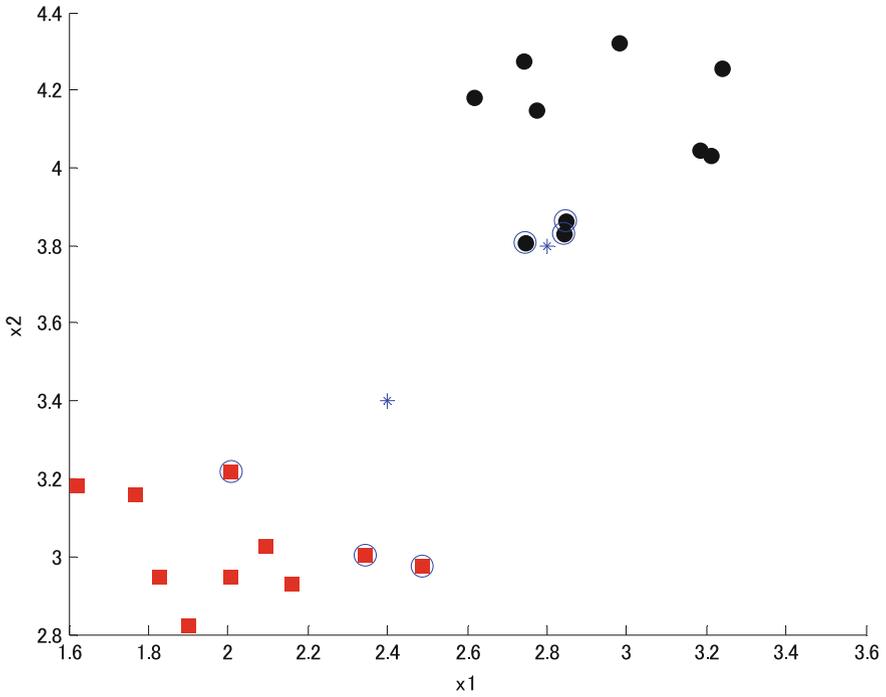


Fig. 8.4 Result of classification by KNN. Classified nearest neighbors are circled blue

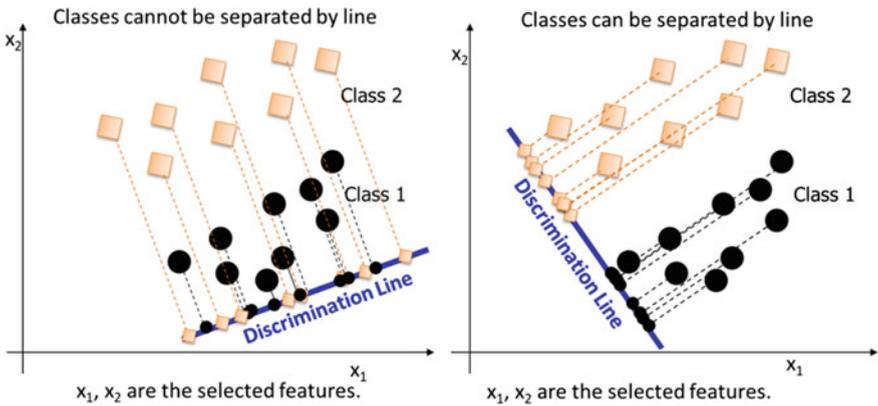


Fig. 8.5 Illustration of the operating principle of LDA

class data follows Gaussian distribution. While each class is to have a different mean μ_j , all classes are assumed to have a common covariance matrix Σ .

With some mathematical manipulation, it can be shown that the discriminant rule $\hat{d}_j(x)$ can be expressed as

$$\begin{aligned}\hat{d}_j(x) &= x^T \hat{\Sigma}^{-1} \hat{\mu}_j - \frac{1}{2} \left(\hat{\mu}_j^T \hat{\Sigma}^{-1} \hat{\mu}_j \right) + \log(\hat{p}_j) \\ &= a_j + b_j^T x.\end{aligned}\tag{8.2}$$

$$\hat{f}(x) = \operatorname{argmax}_{j=1,\dots,K} \hat{d}_j(x),\tag{8.3}$$

where p_j is the probability of class j . The “hat” on the variables denotes estimate values since these have to be computed from data. To use this model as a classifier, we just need to estimate from the training data the class prior probabilities which are normally calculated by taking the proportion of each class in the training data. For example, if an equal number of training data used for each class in a 2-class problem, then the prior probabilities for each class will be 0.5. Similarly, the class means can be estimated from data.

It can be seen that the $f(x)$ is a linear function of the feature measurements x . A practical approach to perform LDA is through eigenvalue decomposition of the common covariance matrix. Classification then follows the following steps [10].

1. Estimate the prior class probabilities \hat{p}_j , class mean $\hat{\mu}_j$, and common covariance $\hat{\Sigma}$.
2. Perform singular value decomposition of the covariance $\hat{\Sigma} = UDU^T$, where D is diagonal, and U is an orthogonal matrix.
3. Transform class centroids such that $\tilde{\mu}_j = D^{-1/2}U^T\hat{\mu}_j$
4. Transform query points using the relation $\tilde{x} = D^{-1/2}U^T x$
5. Classify according to the rule: $\min_j \left(\frac{1}{2} \|\tilde{x} - \tilde{\mu}_j\|_2^2 - \log(\hat{p}_j) \right)$.

Using the above procedure, the performance of LDA can be evaluated.

Example Consider a simple example of two classes, which are well separated (within-class covariance is small and between-class covariance is large). The randomly generated data for the two classes is shown in Fig. 8.6.

The result of the classification of the data by LDA can be summarized in the confusion matrix in Table 8.1.

The results show that LDA can accurately classify the data. In this case, it is possible to find a discriminant line that separates the two classes.

Although LDA can perform well for well-separated classes, it cannot handle non-linear decision boundaries. The data distribution assumption also limits the application of LDA to a subset of problems.

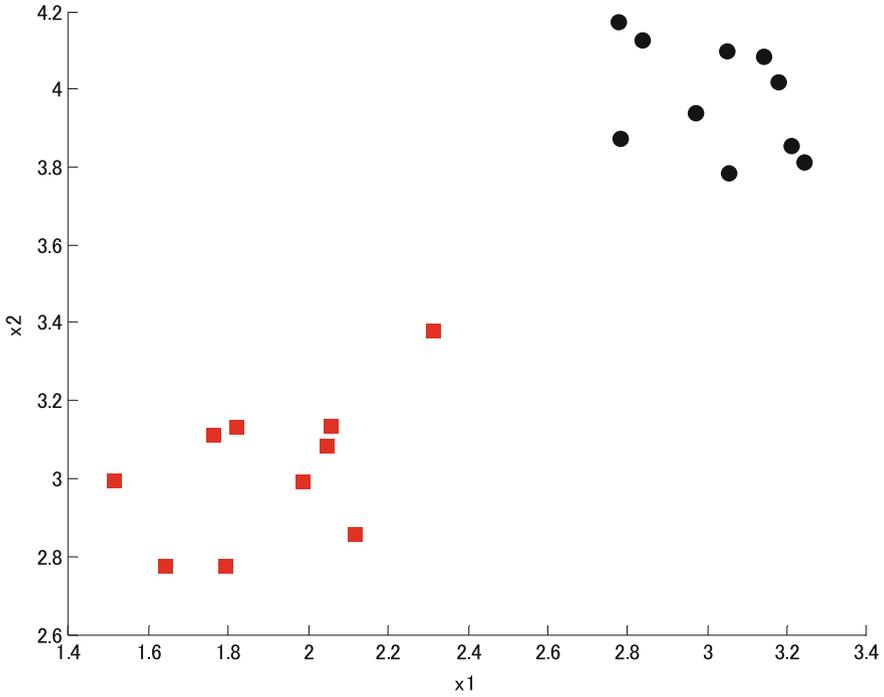


Fig. 8.6 LDA example data, where the Classes 1 and 2 are shown as red squares and black circles, respectively

Table 8.1 Confusion matrix showing classification results by LDA

Actual	Predicted	
	Class 1 (%)	Class 2 (%)
Class 1	100	0
Class 2	0	100

Quadratic Discriminant Analysis (QDA)

Due to the inability of LDA to separate classes correctly for the nonlinear problem, the QDA was formulated. An example of such class separation is shown in Fig. 8.7. The only difference between the two classifiers is that QDA uses a quadratic discriminant function instead of the linear one used by LDA.

$$\hat{d}_j(x) = -\frac{1}{2} * \log\left(\left|\frac{\Sigma}{n_j}\right|\right) - \frac{1}{2} * (x - \mu_j)^T \Sigma_j^{-1} (x - \mu_j) + \log(\hat{p}_j) \tag{8.4}$$

$$\Sigma_j = \frac{1}{n_j} * \sum_{k \in C_j} (x_k - \mu_j)^T (x_k - \mu_j) \tag{8.5}$$

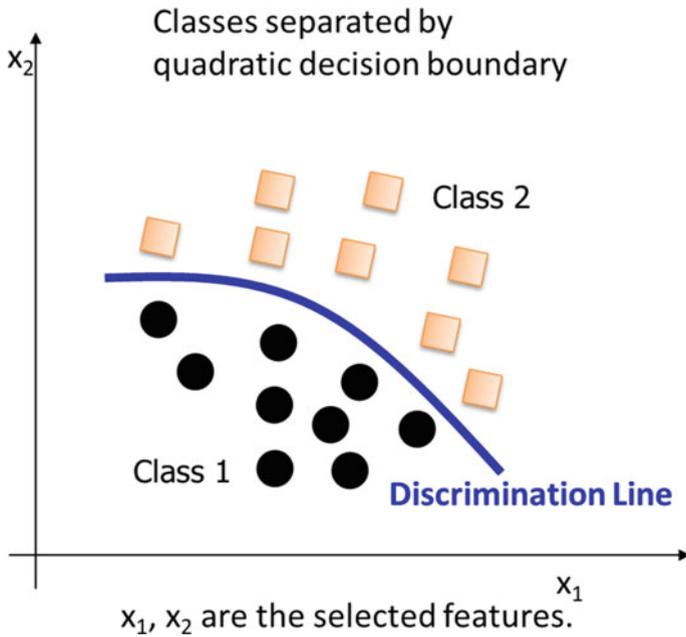


Fig. 8.7 Example of nonlinear separation between classes that can be solved by using QDA

Besides QDA, the nonlinear functions can be used as discriminant functions resulting in what is referred to as nonlinear discriminant analysis (NDA). Table 8.2 gives a comparison of the characteristics of discriminant analysis methods.

Support Vector Machine (SVM)

Support vector machines (SVMs) are machine learning algorithms based on statistical learning theory, which perform classification by constructing hyperplanes in a multi-dimensional space [11, 12]. The SVM algorithms were introduced to solve supervised classification and regression problems [13, 14]. While the roots of SVMs are in image recognition, to date they have since been expanded to other areas including remote sensing [15, 16]. In general, SVMs select the decision boundary from an infinite number of potential ones, leaving the greatest margin between the closest data points to the hyperplane (Fig. 8.8), which are referred to as “support vectors.” SVMs employ

Table 8.2 Comparison of the properties of discriminant analysis methods

Property	Method		
	LDA	QDA	NDA
Discriminant function	Linear	2nd order	Nonlinear
Class covariance	Common	Distinct	Distinct
Data distribution	Gaussian	Gaussian	Mixed Gaussian

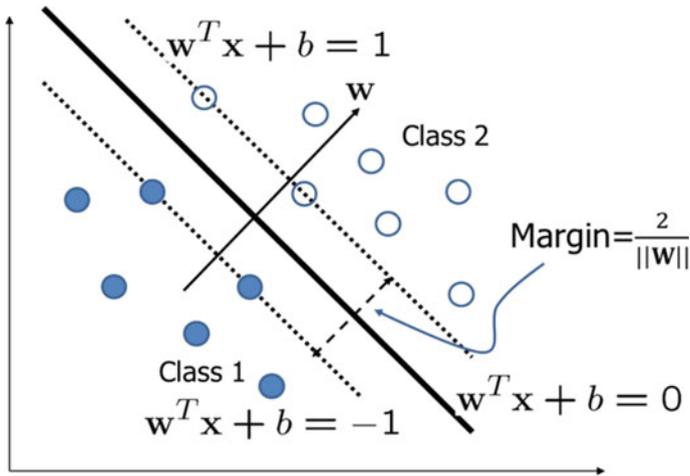


Fig. 8.8 Linear separating hyperplanes for the separable case, where solid circles belong to Class 1, while the white circles belong to Class 2. The optimum hyperplane separating the classes is shown by the thick black line, while support vectors are the circles on the dotted lines

a kernel function to transform the training data into higher dimensional feature space for nonlinear classification problems [17]. In this regard, SVMs are considered as belonging to kernel methods since kernel functions are used to maximize the margin between classes.

From Fig. 8.8, it is desirable to maximize the distance from the decision boundary $w^T x + b = 0$ to either of the two classes. This can be achieved by maximizing the margin $m = 2/\|w\|$. The approach to this maximization utilizes a constrained optimization theory where the underlying problem can be formulated as follows. Suppose the dataset $\{x_1, \dots, x_n\}$ is available, and two class labels for the data are defined by $y_i = \{-1, 1\}$. Then, the decision boundary that classifies all points correctly is given by:

$$y_i(w^T x_i + b) \geq 1 \quad \forall i \tag{8.6}$$

where $\forall i$ denotes “for all i .”

The constrained optimization problem can be formulated as follows:

$$\text{minimize : } \frac{1}{2} \|w\|^2 \tag{8.7}$$

$$\text{subject to : } y_i(w^T x_i + b) \geq 1 \quad \forall i. \tag{8.8}$$

The above problem can be formulated by first defining the following Lagrangian:

$$L = \frac{1}{2}\|w\|^2 + \sum_{i=1}^n \alpha_i (1 - y_i (w^T x_i + b)) \quad (8.9)$$

where α_i are the Lagrange multipliers. The Lagrangian defines the necessary conditions for the solution to the optimization problem to exist. Each constraint is weighted by a Lagrange multiplier, and the task is to find the Lagrange multipliers that minimize the function L . Taking derivatives of L with respect to $\frac{1}{2}\|w\|^2$ and b , and equating the result to zero, we obtain the solutions

$$w = \sum_{i=1}^n \alpha_i y_i x_i - \sum_{i=1}^n \alpha_i y_i = \mathbf{0}. \quad (8.10)$$

Substituting w into L , it can be shown that

$$L = \sum_{i=1}^n \alpha_i y_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j x_i^T x_j. \quad (8.11)$$

The original constrained optimization problem can now be stated as follows:

$$\text{maximize : } \sum_{i=1}^n \alpha_i y_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j x_i^T x_j \quad (8.12)$$

$$\text{subject to : } \sum_{i=1}^n \alpha_i y_i = 0, \quad \alpha_i \geq 0 \quad (8.13)$$

All the variables in the above problem are known except for α_i . If α_i can be found, then w can also be determined, and hence, the margin $m = 2/\|w\|$ can be computed. The Lagrange multipliers α_i can be obtained through quadratic programming (QP) methods [14, 18]. Except for the support vectors shown in Fig. 8.8, the Lagrange multipliers for the all the other samples will be zero.

The abovementioned procedure works well when the hyperplane is linear and the classes are completely separable. However, in practice this is not always the case, making it necessary to modify the optimization criterion in order to handle linear non-separable and nonlinear problems. This modification can be achieved by introducing slack variables and kernel functions. Firstly, for the linear non-separable case (Fig. 8.9), a slack variable ξ can be introduced. The slack variable takes into account misclassified samples. Although the majority of samples can still be correctly classified linearly, only a few samples (noise) fall into the wrong class. The optimization problem can now be stated as:

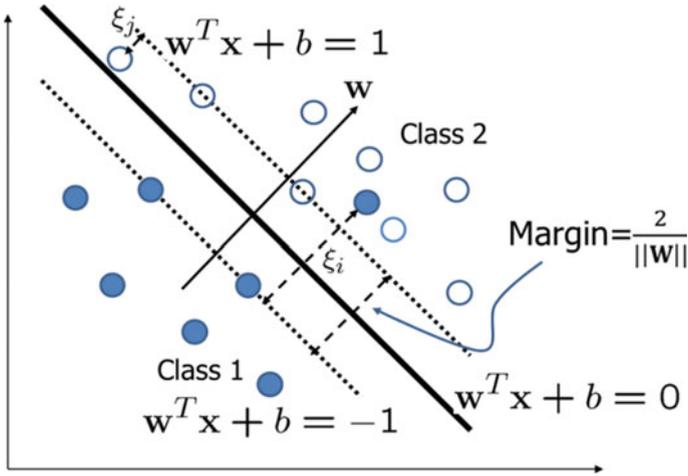


Fig. 8.9 Illustration of the linear non-separable case. The slack variable ξ adds a penalty for violating the optimization constraints

$$\text{minimize : } \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n \xi_i \tag{8.14}$$

$$\text{subject to : } y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \tag{8.15}$$

The regularization parameter C controls the trade-off between the complexity of SVMs and the number of non-separable points, which is determined through cross-validation as an error penalty. Thus, C controls the trade-off between maximizing the margin and minimizing the training error. A small C -value tends to emphasize the margin while ignoring the outliers in the training data, whereas a large C -value may overfit the training data.

Following the steps outlined for the linear separable case, the optimization problem becomes:

$$\text{maximize : } L(\alpha) = \sum_{i=1}^n \alpha_i y_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \tag{8.16}$$

$$\text{subject to : } \sum_{i=1}^n \alpha_i y_i = 0, \quad C \geq \alpha_i \geq 0 \tag{8.17}$$

As with the linear separable case, the non-separable case can also be solved using QP methods.

For the nonlinear separable case (Fig. 8.10), kernel functions can be used to obtain an acceptable solution. Kernel functions allow the separating hyperplane to be transformed to the linear domain, where computations can then be performed.

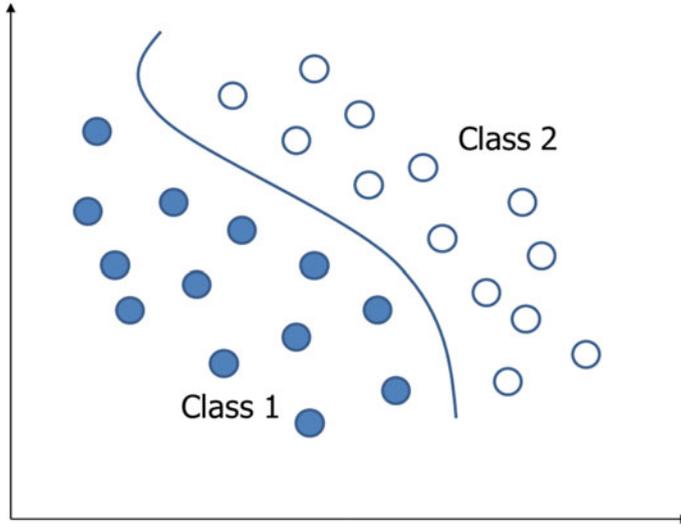


Fig. 8.10 Nonlinear separable case. The decision boundary can be approximated by a nonlinear function

However, explicit transformation is computationally expensive, so the “kernel trick” is employed, where transformation into the inner product space is possible without explicit computation. The kernel function is assumed to satisfy Mercer’s theorem [19]. Mercer’s theorem states that a symmetric function $K(x, y)$ can be expressed as an inner product $K(x, y) = \Phi(x)\Phi(y)$ for some Φ if and only if $K(x, y)$ is positive semi-definite. This implies that the n by n kernel matrix in which the (i, j) -th entry is $K(\mathbf{x}_i, \mathbf{x}_j)$ which is always positive semi-definite, which in turn guarantees the existence of a solution to the optimization problem.

The kernel functions for the nonlinear case are generally defined by the following equation:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i)\Phi(\mathbf{x}_j). \quad (8.18)$$

The nonlinear optimization problem can then be expressed as follows:

$$\text{maximize : } L(\alpha) = \sum_{i=1}^n \alpha_i y_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \quad (8.19)$$

$$\text{subject to : } \sum_{i=1}^n \alpha_i y_i = 0, \quad C \geq \alpha_i \geq 0 \quad (8.20)$$

Besides the regularization parameter C , the problem now involves the choice of an appropriate kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$. Examples of typical kernels include the polynomial, radial basis function (Gaussian), inverse multi-quadratic, and to some

extent sigmoid functions. A detailed analysis of kernel-based approaches is beyond the scope of this section, but it suffices to say that low-degree polynomial or radial basis functions (RBF) could be a good starting point when selecting kernel functions.

While SVMs were originally meant to be binary classifiers, several approaches can be used to address multi-class problems that are standard for the classification of radar sensor data. The first approach, which is the most common and simplest, is referred to as one-against-all (OAA). This approach uses one binary SVM for each class and training. The second approach is known as one-against-one (OAO) or pairwise. In this approach, each classifier is trained to separate a pair of classes, and all classification results are combined into one final result. The third approach is known as multi-class ranking SVM (or structured SVM).

8.3 Challenges and Trends in Target Recognition Based on Radar

There are several challenges in target recognition. In this section, we briefly summarize some of these challenges and give a snapshot of current trends that are focussed on finding practical solutions.

8.3.1 Challenges

Feature selection

Feature selection is one of the big challenges for radar-based recognition. Some of the examples of features that can be considered are found in [1]. The variance and standard deviation of data extracted multiple ranges and velocity detections around the targets can be used as features. However, received data is usually corrupted by both noise and clutter, and hence, intricate and diligent preprocessing is required. Another source of feature is to use the micro-Doppler signature from radar reflection [2–4]. The features of interest are mean velocity (mean Doppler frequency), fundamental gait frequency, stride length, and radar cross section (RCS). Extraction of these features is normally accomplished via short-time Fourier transformation. This approach is effective when pedestrians have to be recognized among the targets [20].

Recognition rates

For automotive applications, the recognition rates by most machine learning algorithms can sometimes be lower than 70%. These levels are unacceptable, especially when one of the targets under consideration is vulnerable road users like pedestrians. Misclassification could, for example, result in unpredictable system behavior and increase the probability of collision with objects. Therefore, recognition rates

greater than 99% will be demanded in such cases. This is especially true for autopilot systems where near-perfect recognition rates are required.

Processing requirements

The nature of the radar sensor places stringent requirements on processing requirements. The choice of radar parameters have has a great impact on performance [21]. For short-range radar, the radar data memory is normally less than 1 MB [22]. In addition, the processing time is normally limited to between 50 and 100 ms. These radar requirements demand careful consideration when selecting the processing algorithm. This means memory-intensive algorithms cannot be easily implemented with current radar architectures.

8.3.2 Current Trends

In order to solve the problems associated with machine learning algorithms such as low recognition and need retrain for new objects, deep learning has recently been intensively investigated as one of the candidate solutions [23–26]. To understand the origins of deep learning, it is instructive to give some insight into the position of these algorithms in the broader picture of artificial intelligence. The field of artificial intelligence encompasses both machine learning and deep learning. Artificial intelligence has a multitude of definitions, but the common point is the desire to make machines which have some level of human intelligence. Thus, computer scientists and engineers define artificial intelligence as the ability of computer systems to perform intelligent tasks. Some notable examples include computer vision, natural language processing, machine learning, pattern recognition, robotics, expert systems, and neural networks.

On the other hand, machine learning (ML) is concerned with the study of computer algorithms and statistical models that can accomplish intelligent tasks. As previously mentioned, these algorithms can be categorized into supervised learning, semi-supervised learning, and unsupervised learning. Finally, deep learning has roots in artificial neural networks which in turn are modeled along human neurons, referred to as perceptrons [5]. Details of neural networks are beyond the scope of this section, but a perceptron takes several binary inputs and produces a single binary output as illustrated in Fig. 8.11. The output can be computed using the following expressions:

$$\text{output} = \begin{cases} 0, & \sum_{i=0} w_i x_i \leq \theta_0 \\ 1, & \sum_{i=0} w_i x_i > \theta_0 \end{cases} \quad (8.21)$$

where $x_0 = 0$ and θ_0 is a predetermined threshold.

The concept can be extended to multiple layers of perceptrons to produce complex decisions as shown in Fig. 8.12.

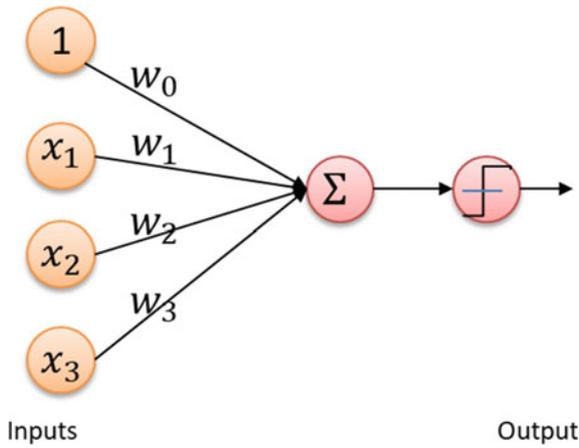


Fig. 8.11 Simple perceptron model

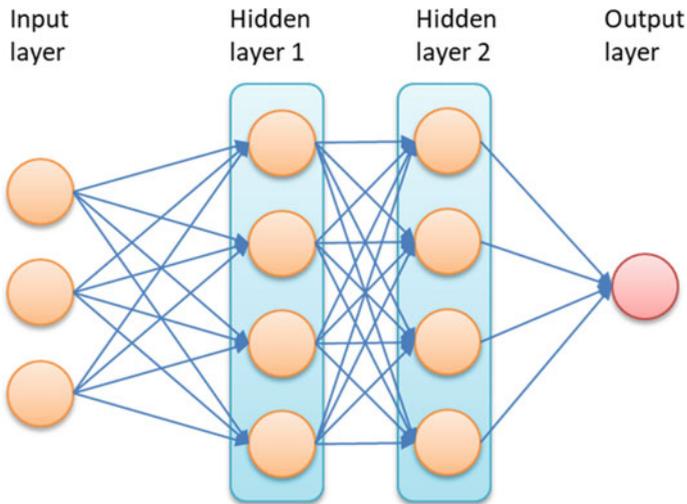


Fig. 8.12 Example of a multilayer perceptron model

Deep learning networks differ from neural networks by the number of node layers used which brings in the concept of depth. Neural networks normally have one to two hidden layers and are used for supervised prediction or classification. Deep learning networks can have several hidden layers with the possibility of unsupervised training. Figure 8.13 illustrates one example of such a network. Examples of widely used deep learning architectures include deep neural networks (DNN), deep belief networks (DBF), and recurrent neural networks (RNN). The main advantage of DNN over traditional neural networks is the ability to learn complex tasks in an

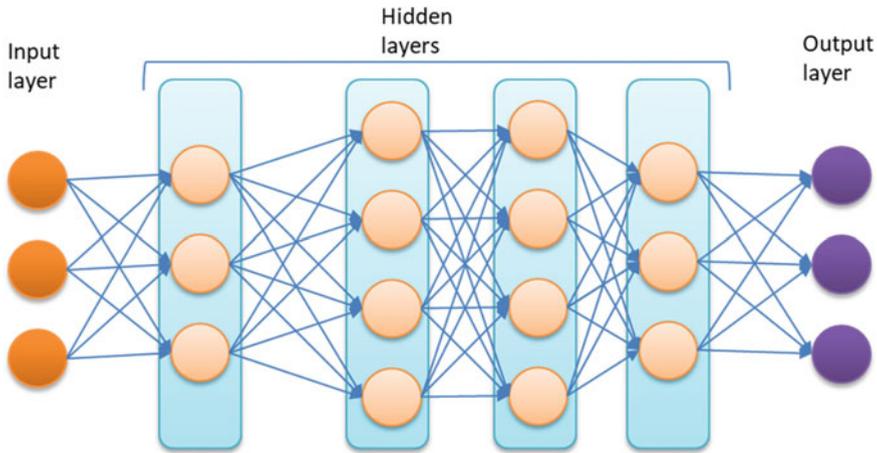


Fig. 8.13 Example of a deep neural network model

unsupervised manner. However, this advantage comes at no cost. Large amounts of data are required for building the network, high computational complexity is a big burden, and difficulties arise when attempting to analyze the algorithms and also inability to predict the output precisely, among other challenges. For automotive applications, DNN has a promising future although direct application to automotive radar is still currently in its research phase [27]. The obvious reason being the computational intensiveness of the algorithms and also the complexity of automotive radar environments.

For the interested reader, further details about DBM and RNN can be found in [28, 29], respectively. It should be noted that RNNs have found better success in natural language process (NLP).

References

1. Heuel, S., Rohling, H.: Pedestrian recognition based on 24 GHz radar sensors. In: 11-th International Radar Symposium, 12 Aug 2010
2. Chen, V., et al.: Micro-doppler effect in radar: phenomenon, model, and simulation study. *IEEE Trans. Aerosp. Electron. Syst.* **42**(1), 2–21 (2006)
3. Bilik, I., Tabrikian, J.: Radar target classification using doppler signatures of human locomotion models. *IEEE Trans. Aerosp. Electron. Syst.* **43**(4), 1510–1522 (2007)
4. Yan, H., Doerr, W., Ioffe, A., Clasen, H.: Micro-doppler based classifying features for automotive radar VRU target classification. In: 25th International Technical Conference on the Enhanced Safety of Vehicles (ESV), Detroit Michigan, United States (2017)
5. Bishop, C.M.: *Neural Networks for Pattern Recognition*. Oxford University Press, Inc., New York (1995)
6. Scikit-learn, <https://scikit-learn.org/stable/>
7. OpenCV, <https://www.learnopencv.com/>

8. Cover, T., Hart, P.: Nearest neighbor pattern classification. *IEEE Trans. Inf. Theory* **13**(1), 21–27 (1967)
9. Fisher, R.A.: The use of multiple measurements in taxonomic problems. *Ann. Eugen.* **7**, 179–188 (1936)
10. Hastie, T., Tibshirani, R., Friedman, J.: *The Elements of Statistical Learning*. Springer (2008)
11. Cortes, C., Vapnik, V.: Support-vector network. *Mach. Learn.* **20**(3), 273–297 (1995)
12. Huang, Z., Lee, B.G.: Combining non-parametric models for multisource predictive forest mapping. *Photogram. Eng. Remote Sens.* **70**, 415–425 (2004)
13. Boser, B.E., Guyon, I.M., Vapnik, V.N.: A training algorithm for optimal margin classifiers. In: *Fifth Annual Workshop on Computational Learning Theory*, pp. 144–152. ACM (1992)
14. Vapnik, V.N.: *The Nature of Statistical Learning Theory*. Wiley, New York (1998)
15. Camps-Valls, G., Bruzzone, L.: Kernel-based methods for Hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **43**(6), 1351–1362 (2005)
16. Bruzzone, L., Persello, C.: A novel context-sensitive semi-supervised SVM classifier robust to mislabeled training samples. *IEEE Trans. Geosci. Remote Sens.* **47**(7) (2009)
17. Liu, J., et al.: Radar target classification using support vector machine and subspace methods. *IET Radar Sonar Navig.* **9**(6), 632–640 (2015)
18. Vapnik, V.N.: *Statistical Learning Theory*. Wiley, New York (1998)
19. Burges, C.J.C.: *A Tutorial on Support Vector Machines for Pattern Recognition*, pp. 1–43. Kluwer Academic Publishers, Boston (1998)
20. Fioranelli, F., Ritchie, M., Griffiths, H.: Bistatic human micro doppler signatures for classification of indoor activities. In: *2017 IEEE Radar Conference (Radar Conf)*, Seattle, WA, USA, 08–12 May 2017
21. Bloecher, H.-L., Andres, M., Fischer, C., Sailer, A., Goppelt, M., Dickmann, J.: Impact of system parameter selection on radar sensor performance in automotive applications. *Adv. Radio Sci.* **10**, 33–37 (2012)
22. Singh, J., Ginsburg, B., Rao, S., Ramasubramanian, K.: AWR 1642 mm wave sensor: 76–81-GHz radar-on-chip for short-range radar applications. Texas Instruments, May 2017
23. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. *Science* **313**(5786), 504–507 (2006)
24. Hinton, G.E., Osindero, S., Teh, Y.: A fast learning algorithm for deep belief nets. *Neural Comput.* **18** (2006)
25. Furukawa, H.: Deep learning for end-to-end automatic target recognition from synthetic aperture radar imagery. *IEICE Technical Report*, vol. 117, No. 403, SANE 2017-92, pp. 35–40 (2018)
26. Angelov, A., Robertson, A., Murray-Smith, R., Fioranelli, F.: Practical classification of different moving targets using automotive radar and deep neural networks. *IET Radar, Sonar Navig.* **12**(10), 1082–1089 (2018)
27. Wheeler, T.A., Holder, M.F., Winner, H., Kochenderfer, M.J.: Deep stochastic radar models. In: *IEEE Intelligent Vehicles Symposium (IV)* (2017)
28. Salakhutdinov, R., Hinton, G.E.: Deep Boltzmann Machines. *AISTATS*, pp. 448–455 (2009)
29. Graves, A., Mohamed, A., Hinton, G.E.: Speech recognition with deep recurrent neural networks. In: *ICASSP*, pp. 6645–6649 (2013)

Chapter 9

Automotive Radar Applications



9.1 Introduction

Key Applications Background

This chapter gives some important current and future automotive applications that utilize the radar as a sensor. In automotive applications, the signal to be processed can be acquired by ultrasonic sensor, radar sensor, camera, LIDAR, GPS, and other engine control unit sensors. Audio, image, and array processing techniques can be applied to the acquired signals in order to extract information that can be used to automatically control vehicle dynamics or alert the driver of impending dangers around the vehicle. Although there are multiple ways to look at automotive applications, automotive safety applications that apply radar signal processing will be the main focus of this chapter. Safety is fast becoming a very important component of automotive performance metrics. Safety can be passive, such as safety belts, or active, such as automatic emergency braking. We will mostly be concerned with active safety as it is directly related to future driving technology such a driverless or autonomous driving that uses radar sensing technology. In fact, most of this section will deal with autonomous driving-related applications.

It is predictable that in the future, autonomous driving systems will utilize radar, imaging, and optical sensors as key components of active safety systems. Although imaging and optical sensors can be a part of the sensor fusion approach, radar has distinctive advantages over these technologies. The advantages include the ability to detect both range and velocity simultaneously, ability to detect objects during the day and night, ability to operate in adverse conditions such as rain and snow, and the ability to be installed behind the front bumper fascia, thereby maintaining vehicle esthetics intact. Additionally, with fewer small-sized sensors, millimeter-level accuracy in range detection is possible with the radar sensor for both long-range and short-range applications.

In most automotive applications, the main idea is to sense the imminent future and warn or assist the driver in mitigating dangerous situations. With regard to vehicles, applications can be broadly divided into short-range and long-range categories. In the

short-range category, there is blind spot detection, rear cross-traffic detection for rear-looking sensors and cut-in warning, front cross-traffic, pedestrian/cyclist detection, and side impact warning for the front- and side-sensors. In these applications, the sensor acquires the reflected signal from objects by the use of a single or an array of sensors. Through signal processing means, the distance, speed, and angle of the detected objects can be estimated. This information can then be used to determine whether the detected objects pose any danger in the path of the vehicle. In case of danger, automatic avoidance systems are activated and if the driver is not responsive, the measure can be taken to reduce the impact of the danger. In one driver assistance application such as pedestrian detection, fatalities can be avoided by warning the driver of pedestrians in the vehicle's path. Therefore, signal processing plays a key role in extracting moving object information. Typically, the short-range radars cover ranges up to 30 m and speed up to 150 km/h. They are characterized by a wide field of view (FOV) of more than $\pm 40^\circ$ and large bandwidth of up to 5 GHz for high-range resolution.

In the long-range applications, automatic cruise control (ACC), forward collision warning, and pedestrian collision warning are essential. In most markets, it is expected that new vehicle's models will be able to reliably detect pedestrians from 2020 onwards. In contrast to the short-range radar sensors, long-range radars typically cover ranges up to 250 m and speed up to 250 km/h. They are characterized by narrow FOV of less than $\pm 10^\circ$ and narrow signal bandwidth of less than 1 GHz. Due to the narrow FOV, long-range radars cannot detect objects in short distance and away from the radar's center of measurement.

Key Players and Contributors

The automotive industry is going in rapid developments. There are a lot of players promoting and advancing applications for comfort and safety. These developments take a global perspective to such an extent that almost every part of the automotive industry is involved. At the government level, the key player in Europe is Germany. Europe takes the lead in assessment efforts via Euro-NCAP and associated auto industry consortia. In North America, the USA takes the leading effort through the DoT, NHTSA, and various support groups from the auto industry. In Asia, Japan is at the forefront of advancing automotive applications through MLIT, JNCAP, among others. Governments provide the necessary regulatory framework on which all automotive technologies operate. It is also the governments that set standards for requirements that would otherwise be ignored by automotive industry such as pedestrian protection.

The auto industry is also very active since it implements these applications. Starting from Japan, Toyota, Honda, Mazda, Nissan, Subaru, and all other major automakers are involved in the development of advanced driver assistance systems (ADAS) that use radar technology in conjunction with other technologies. The applications range from simple blind spot detection to autonomous driving-oriented applications such as auto-parking. In other parts of the world, Daimler (Mercedes-Benz), Audi, Ford, GM, Tesla, and other international automakers are pushing technology to the limits. Of course, automakers cannot accomplish the task alone, so auto-suppliers

such Continental, Bosch, Denso, Fujitsu-Ten, Autoliv, and Valeo are making enormous effort to realize the goal of zero fatalities through innovative automotive applications. As stated above, pedestrian protection is one of the key elements of these applications as we move toward autonomous driving. Besides the auto industry and auto-suppliers, the IT and technology industry is becoming a powerful competitor in the field. Notable companies involved in recent research and development activities include Google's driverless car project, Uber's ride-sharing, Lyft working with GM on autonomous driving, Apple, Amazon, and robotics development entities. In addition, ISO, through its various working groups, such as ISO WG-14, is promoting the standardization efforts of these applications.

This chapter gives some important current and future automotive radar application. The main idea behind these applications is to sense the immediate condition, predict the future, and warn or assist the driver in mitigating dangerous situations. The applications dealt with in this chapter can be broadly divided into short-range and long-range categories. While long-range applications are now very well established in the automotive industry, short-range applications are still actively under research and development. In the following sections, accuracy and resolution of the measured parameter will be given where appropriate. The sensor resolution is defined as the minimum necessary condition that allows the discrimination of two adjacent objects of equal size and reflecting characteristic. The reflecting characteristics are normally defined by the radar cross section (RCS). On the other hand, accuracy refers to the standard deviation of the measured quantity relative to the expected value.

The next section summarizes some of the key applications and corresponding radar requirements. Due to the diversity of sensors provided by radar manufacturers, typical values for the radar parameters in data sheets will be given.

9.2 Short-Range Radar (SRR)

Rear Detection

Blind Spot Detection (BSD)

A side-looking radar sensor detects vehicles in the blind zone which is not visible from the rear-view mirror as depicted in Fig. 9.1 [1, 2]. Upon detection, a warning is provided to the driver to avoid collision, especially when making lane change maneuvers.

The radar requirements for BSD are shown in Table 9.1.

Rear Cross-traffic Alert (RCTA)

The technology is similar to blind spot detection except that it is applied to low-speed situations where the radar sensor detects vehicles at the rear zone as shown in Fig. 9.2 [3, 4]. This is especially important in parking areas where the possibility of collision with crossing vehicles is high.

The radar requirements for RCTA are shown in Table 9.2.

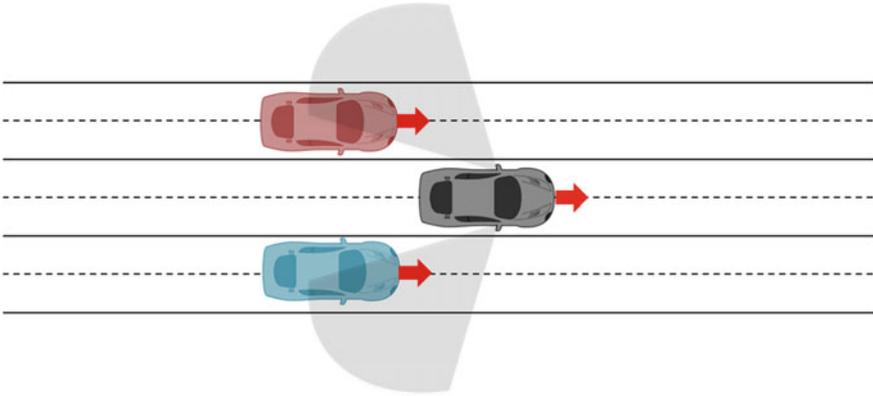


Fig. 9.1 Illustration of blind spot detection scenario

Table 9.1 Radar requirements for BSD

Radar parameter	Requirement
Range (m)	1–50
Range accuracy (m)	± 0.1
Range resolution (m)	0.75
Velocity (m/s)	–70 to +70
Velocity accuracy (m/s)	0.1
Velocity resolution (m/s)	0.25
Azimuth (deg)	± 75
Azimuth accuracy (deg)	± 5
Azimuth resolution (deg)	15
Elevation (deg)	± 6
Elevation accuracy (deg)	0.1
Elevation resolution (deg)	–

Note “–” denotes not required or optional

Front Detection

Front Cross-traffic Alert (FCTA)

The front cross-traffic uses the same concept as the rear cross-traffic except that radar sensors detect vehicles in the front zone [5]. An example is given in Fig. 9.3.

The radar requirements for FCTA are shown in Table 9.3.

Cut-in Warning (CIW)

In this case, the radar sensor detects high-speed vehicles that attempt to cut into own vehicle’s lane from both sides as shown in Fig. 9.4. The driver is warned if there is danger of collision due to cut-in maneuver.

The radar requirements for CIW are shown in Table 9.4.

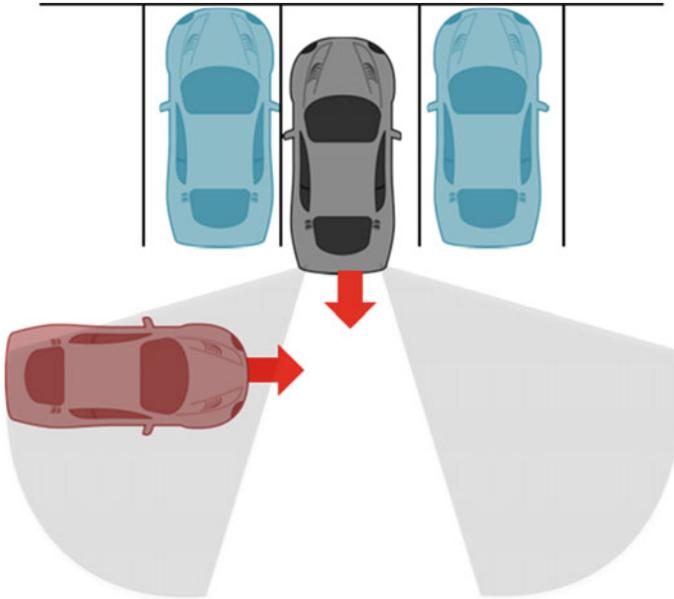


Fig. 9.2 Illustration of RCTA scenario

Table 9.2 Radar requirements for RCTA

Radar parameter	Requirement
Range (m)	2–60
Range accuracy (m)	±0.20
Range resolution (m)	1.0
Velocity (m/s)	-70 to +70
Velocity accuracy (m/s)	±0.1
Velocity resolution (m/s)	0.3
Azimuth (deg)	±40
Azimuth accuracy (deg)	±5
Azimuth resolution (deg)	15
Elevation (deg)	±10
Elevation accuracy (deg)	±5
Elevation resolution (deg)	-

Note “-” denotes not required or optional

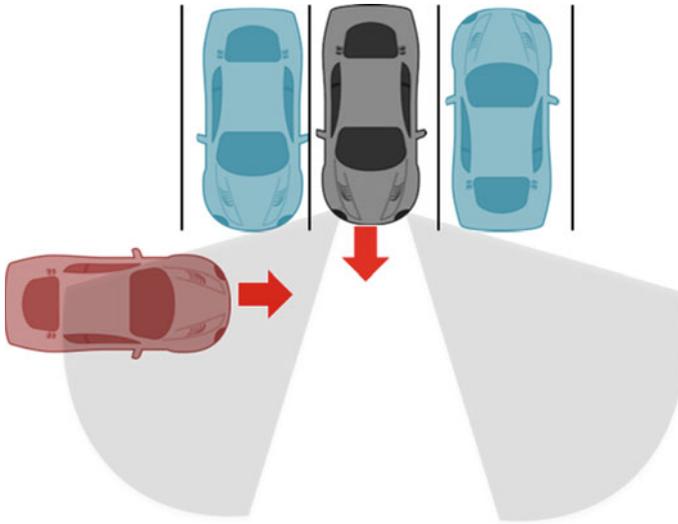


Fig. 9.3 Illustration of FCTA scenario

Table 9.3 Radar requirements for FCTA

Radar parameter	Requirement
Range (m)	2–60
Range accuracy (m)	± 0.20
Range resolution (m)	1.0
Velocity (m/s)	-70 to +70
Velocity accuracy (m/s)	± 0.1
Velocity resolution (m/s)	0.3
Azimuth (deg)	± 40
Azimuth accuracy (deg)	± 5
Azimuth resolution (deg)	15
Elevation (deg)	± 10
Elevation accuracy (deg)	± 5
Elevation resolution (deg)	-

Note “-” denotes not required or optional

Side Detection

Side Impact Warning (SIW)

Using the radar sensor to detect vehicle speed and distance, the driver can be warned of impending side impact from areas as shown in Fig. 9.5. High-range accuracy and resolution are required in order to achieve the desired goal [6].

The radar requirements for SIW are shown in Table 9.5.

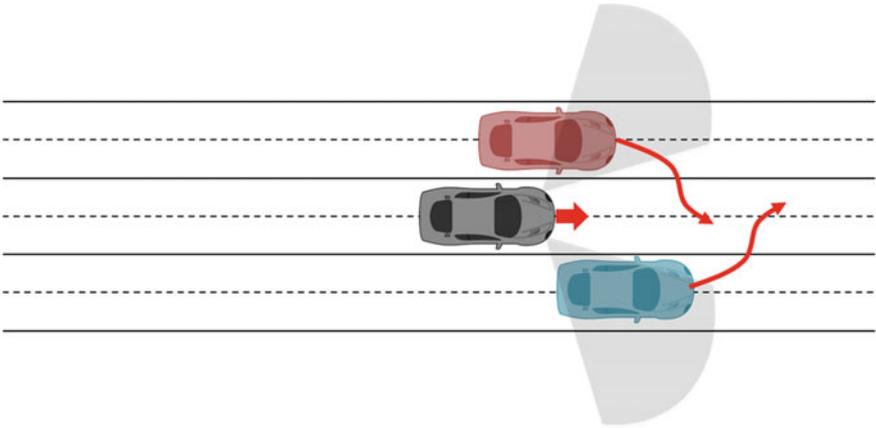


Fig. 9.4 Illustration of CIW scenario

Table 9.4 Radar requirements for CIW

Radar parameter	Requirement
Range (m)	50
Range accuracy (m)	0.02
Range resolution (m)	0.1
Velocity (m/s)	± 70
Velocity accuracy (m/s)	0.1
Velocity resolution (m/s)	0.6
Azimuth (deg)	± 80
Azimuth accuracy (deg)	0.1
Azimuth resolution (deg)	1.0
Elevation (deg)	± 10
Elevation accuracy (deg)	0.1
Elevation resolution (deg)	-

Note “-” denotes not required or optional

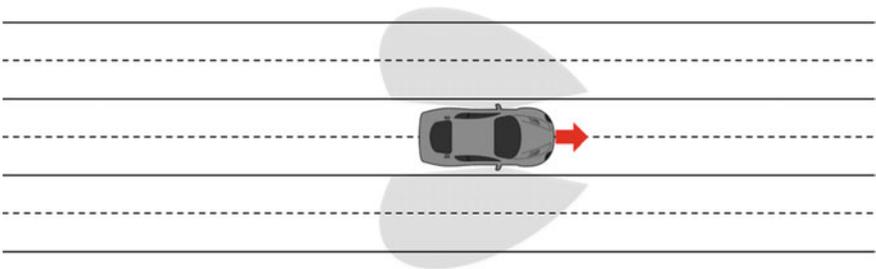


Fig. 9.5 Illustration of SIW scenario

Table 9.5 Radar requirements for SIW

Radar parameter	Requirement
Range (m)	20
Range accuracy (m)	0.02
Range resolution (m)	0.1
Velocity (m/s)	± 70
Velocity accuracy (m/s)	0.1
Velocity resolution (m/s)	0.6
Azimuth (deg)	± 45
Azimuth accuracy (deg)	0.5
Azimuth resolution (deg)	2
Elevation (deg)	± 10
Elevation accuracy (deg)	± 0.5
Elevation resolution (deg)	–

Note “–” denotes not required or optional

Cyclist Warning (CW)

Cyclists are generally classified as vulnerable road users (VRUs) [7–9]. The radar detects cyclists and warns the driver of their presence. Cyclists are not easily visible to the driver and appear unexpectedly in some driving situations. Figure 9.6 is an example of such a scenario.

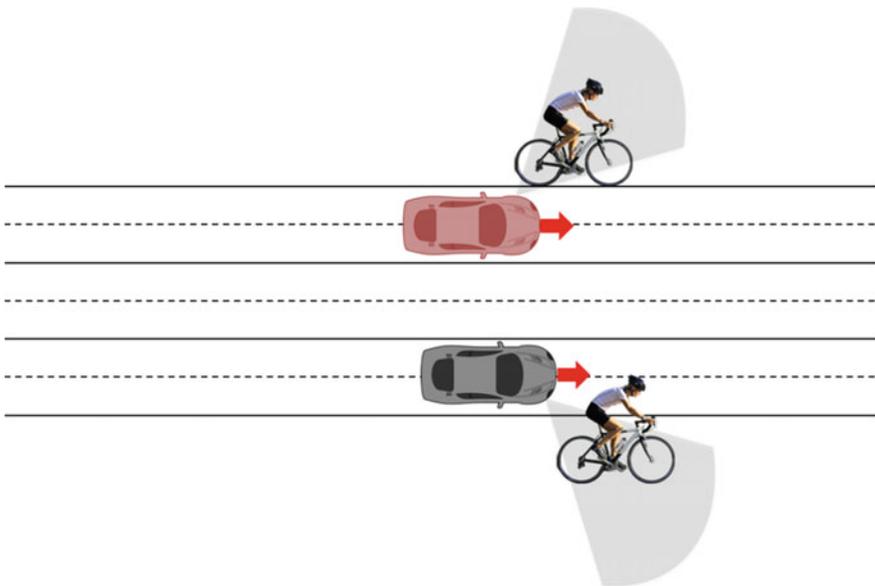


Fig. 9.6 Illustration of CW scenario

Table 9.6 Radar requirements for CW

Radar parameter	Requirement
Range (m)	0.5–50
Range accuracy (m)	0.3
Range resolution (m)	1.0
Velocity (m/s)	±40
Velocity accuracy (m/s)	±0.25
Velocity resolution (m/s)	0.15
Azimuth (deg)	±30
Azimuth accuracy (deg)	1.0
Azimuth resolution (deg)	6.0
Elevation (deg)	±6
Elevation accuracy (deg)	±1.0
Elevation resolution (deg)	–

Note “–” denotes not required or optional

The radar requirements for CW are shown in Table 9.6.

9.3 Long-Range Radar (LRR)

Rear Detection

Rear-End Collision Warning (RCW)

The radar detects and initiates warning to drivers of the possibility of collision, such as during lane change maneuvers or decelerating while other vehicles are approaching from behind as shown in Fig. 9.7. RCW is considered as a part of pre-crash systems [10, 11].

The radar requirements for RCW are shown in Table 9.7.

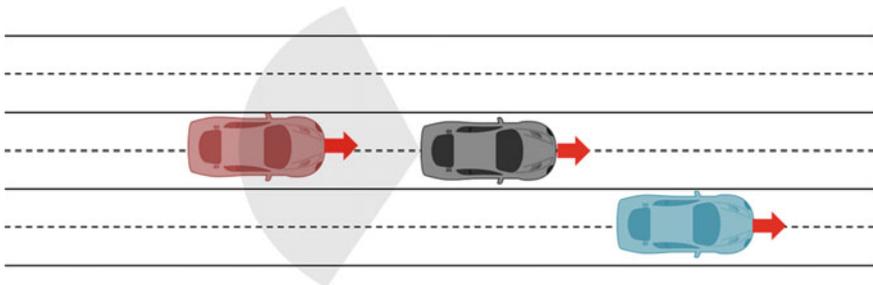


Fig. 9.7 Illustration of RCW scenario

Table 9.7 Radar requirements for RCW

Radar parameter	Requirement
Range (m)	1–60
Range accuracy (m)	± 0.1
Range resolution (m)	0.5
Velocity (m/s)	0 to ± 70
Velocity accuracy (m/s)	± 0.3
Velocity resolution (m/s)	3
Azimuth (deg)	± 40
Azimuth accuracy (deg)	0.1
Azimuth resolution (deg)	5
Elevation (deg)	± 5
Elevation accuracy (deg)	0.1
Elevation resolution (deg)	–

Note “–” denotes not required or optional

Front Detection

Automatic Cruise Control (ACC)

The radar detects a range of preceding vehicles and automatically adjusts the inter-vehicle distance for comfortable driving as depicted in Fig. 9.8. Automatic emergency braking can be incorporated into the ACC function [12].

The radar requirements for ACC are shown in Table 9.8.

Pedestrian Collision Warning (PCW)

The radar detects the presence of pedestrians in the vicinity of the vehicle and initiates warning to the driver as depicted in Fig. 9.9. Pedestrian protection is a key component of autonomous driving and has received a lot of attention from both regulators and manufacturers [13, 14].

The radar requirements for PCW are shown in Table 9.9.

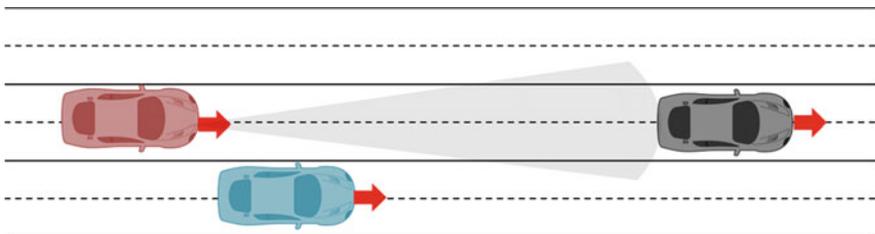
**Fig. 9.8** Illustration of ACC scenario

Table 9.8 Radar requirements for ACC

Radar parameter	Requirement
Range (m)	10–250
Range accuracy (m)	±0.1
Range resolution (m)	0.5
Velocity (m/s)	–70 to +70
Velocity accuracy (m/s)	±0.3
Velocity resolution (m/s)	2.25
Azimuth (deg)	±8 to ±10
Azimuth accuracy (deg)	±0.5
Azimuth resolution (deg)	5
Elevation (deg)	4
Elevation accuracy (deg)	0.1
Elevation resolution (deg)	–

Note “–” denotes not required or optional

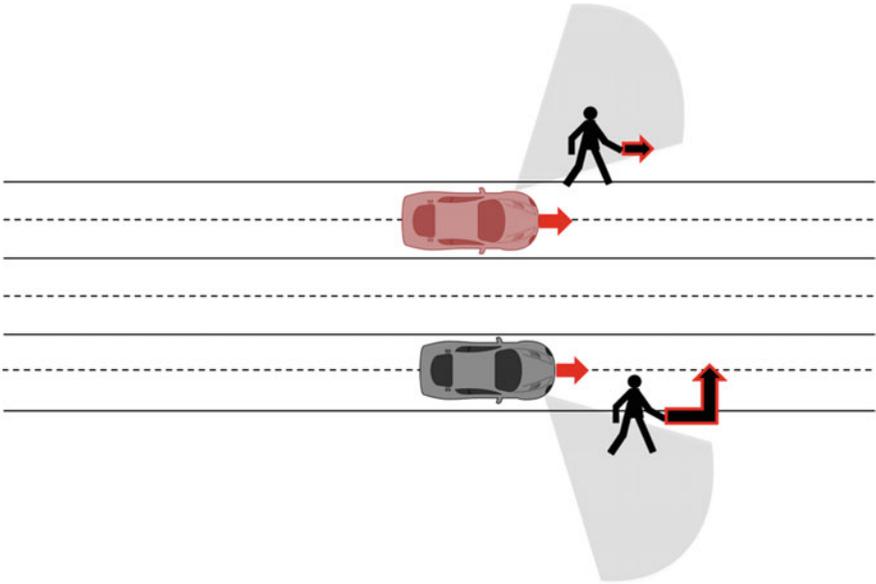


Fig. 9.9 Illustration of PCW scenario

Table 9.9 Radar requirements for PCW

Radar parameter	Requirement
Range (m)	0.75–60
Range accuracy (m)	± 0.2
Range resolution (m)	0.15–0.75
Velocity (m/s)	–70 to 70
Velocity accuracy (m/s)	± 0.3
Velocity resolution (m/s)	0.06 to 0.15
Azimuth (deg)	± 70
Azimuth accuracy (deg)	0.5–2
Azimuth resolution (deg)	–
Elevation (deg)	4–8
Elevation accuracy (deg)	± 5
Elevation resolution (deg)	–

Note “–” denotes not required or optional

Forward Collision Warning (FCW)

The radar detects an imminent frontal crash and initiates safety systems to reduce the severity of the crash, warning the driver in the process. Such situations arise if the preceding vehicle suddenly decelerates. An accurate estimation of the range is important for the effectiveness of this function [15]. Figure 9.10 shows an example of one such scenario.

The radar requirements for FCW are shown in Table 9.10.

The range and sophistication of automotive radar applications are increasing with every new vehicle model. Therefore, the above applications can be considered as part of the basic applications available on the market. The list should not be considered as exhaustive.

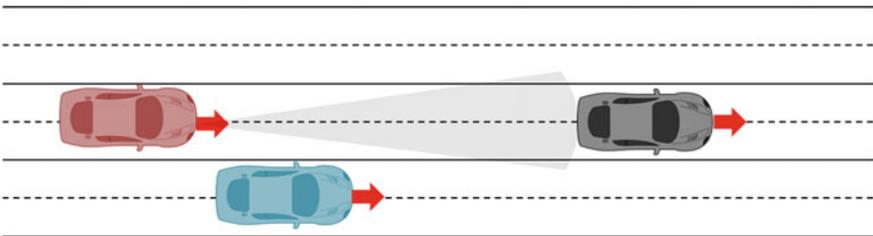
**Fig. 9.10** Illustration of FCW scenario

Table 9.10 Radar requirements for FCW

Radar parameter	Requirement
Range (m)	0.75–200
Range accuracy (m)	±1.0
Range resolution (m)	0.75
Velocity (m/s)	–70 to 70
Velocity accuracy (m/s)	±0.25
Velocity resolution (m/s)	–
Azimuth (deg)	±10
Azimuth accuracy (deg)	0.1
Azimuth resolution (deg)	1.5
Elevation (deg)	4–8
Elevation accuracy (deg)	±5
Elevation resolution (deg)	–

Note “–” denotes not required or optional

9.4 Trends in Automotive Applications

9.4.1 Future Roadmaps Automotive Applications

All involved governments have their own visions for the future of automotive applications. However, the actual implementation will depend on the speed at which automakers, and in turn auto-supplier, can produce the necessary supporting sensor technology. For example, lane detection systems would not be realized without the necessary image acquisition technologies. Even with the sensor technology available, the reliability of the processed sensor data is critical. A system that produces a lot of false alarms will be unacceptable to most drivers.

The general consensus now is to move from ADAS to autonomous driving in the next 10–20 years. As outlined in Chap. 1, Society of Automotive Engineers (SAE) defines six levels of automation, including no automation, which are now seen as the best way to advance this technology. As a quick recap, these are Level 0 (No Automation) where the human driver is in control all the time. Level 1 applies collision mitigation braking that automatically brakes if a collision is imminent. In Level 2 automatic acceleration/deceleration, braking and steering assistance is possible while Level 3 driving system executes steering and acceleration/deceleration operation including monitoring of the driving environment. The human driver takes control of the system’s request. Level 4 extends the autonomous capabilities and can handle all driving responsibilities, even if a human driver fails to respond appropriately to request for intervention. Levels 1–4 apply to some driving modes, but Level 5 takes full control of the vehicle for all driving modes. A human driver is not required.

Roadmaps: Japan, Europe, USA

With the above levels of automation as a general guideline, the roadmaps for Japan [16], Europe [3], and USA [4] are shown in Figs. 9.11, 9.12 and 9.13.

As shown in the roadmaps, full automation can be expected on the roads in the 2030s.

Roadmaps by Automakers: Japan, Europe, USA

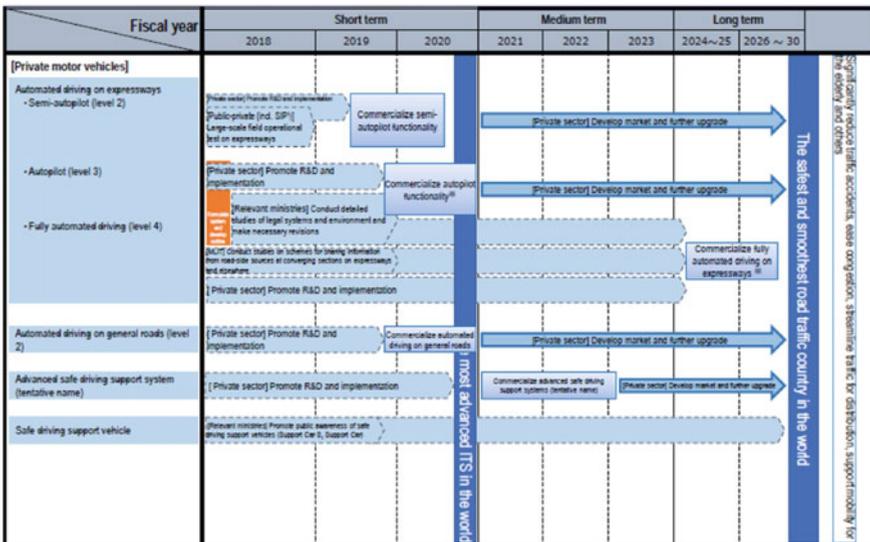
With the vision outlined by respective governments, automakers and suppliers are also racing to make autonomous driving a reality.

Table 9.11 shows the general known directions of some key automakers worldwide. Details can be found in [5].

Levels	Technologies Expected to be Realized	Expected Time of Commercialization	(For Reference) Targeted Time in Europe and Other Regions **
Level 2	• Follow-up and tracking systems	Mid 2010s	2013 - 2015
	• Steering for collision avoidance		2017 - 2018
	• Self-driving on multiple lanes, etc.	In 2017	2016
Level 3	• Automated merging, etc.	First half of 2020s	2020
Level 4	• Full automated driving	Second half of 2020s*	2025 – 2028 (expressways) 2027 – 2030 (urban areas)

* Level 4 (full Automated Driving Systems) is expected to have a trial period. The expected time will be reviewed as necessary due to many uncertainties, based on the examination of commercialization in Japan and abroad as well as the status of various efforts.
 **: The targeted time of commercialization in Europe and other regions was investigated by the Cabinet Secretariat through such channels as the iMobility Forum.

(a) A general road map according to the level of automation.



(b) A detailed roadmap up to year 2030.

Fig. 9.11 Japan autonomous driving roadmap

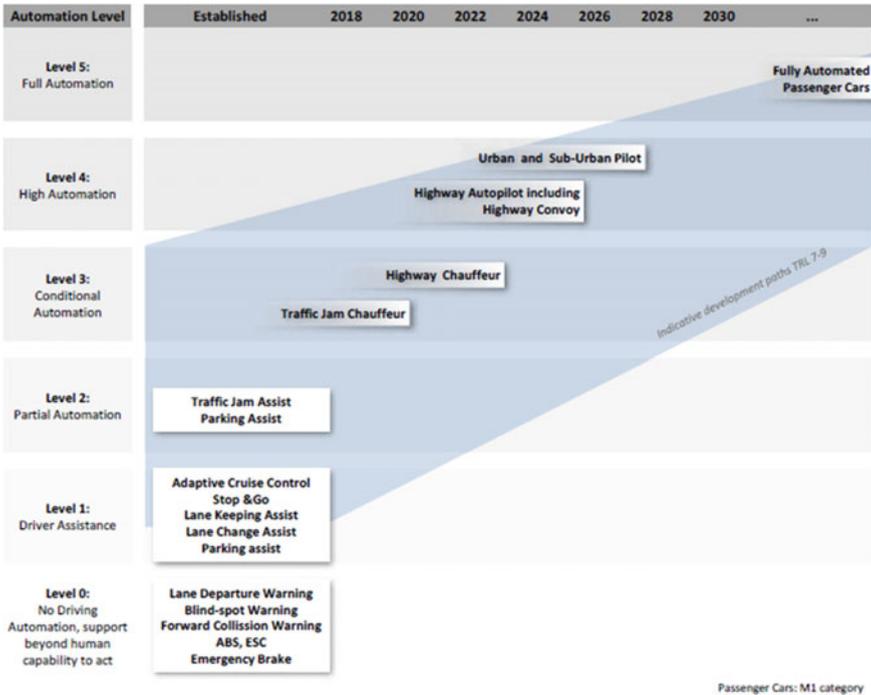


Fig. 9.12 Europe’s autonomous driving roadmap

In Japan, Toyota is promoting the Highway Teammate concept; Honda the Honda Sensing concept, Nissan the Mobility concept, Mazda the Intelligent Transport System, and Subaru is advancing the Eyesight concept.

9.4.2 Future Contributions of Automotive Applications

Benefits of Autonomous Driving to Society

Autonomous vehicles will improve our lives in many ways. Here, some of the benefits of autonomous vehicles often cited as justification for putting resources in this field are given. Although the contents outlined here are not exhaustive, they give an insight into why research and development in autonomous driving is important to the future generation. Like when transportation changed from horses to cars, there will always be winners and losers, but society, in general, stands to benefit very greatly from autonomous vehicles.

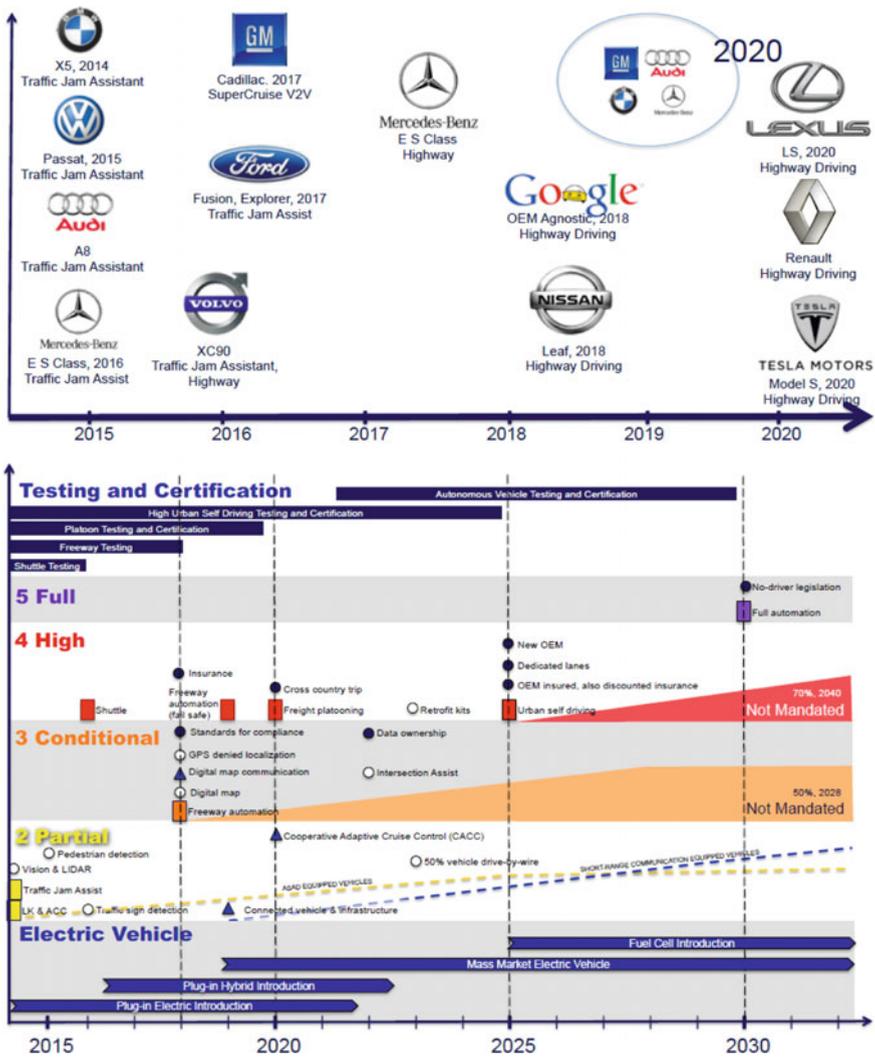


Fig. 9.13 USA autonomous driving roadmap

1. Roads will become safer

By reducing the number of accidents through active sensing, thousands of lives can be saved. In the USA, it is estimated at 90% autonomy level for vehicles on the road, and accidents can be reduced by 80% resulting in 66% reduction of fatalities [17]. Introduction of autonomous vehicles will automatically eliminate drunk and drugged drivers off the steering wheel. Moreover, sensory technology could potentially perceive the environment better than human senses, seeing farther ahead, better

Table 9.11 Estimated targets for autonomous driving levels by the marker

Maker	Level 2	Level 3	Level 4
Audi	2016	2018/2020	2020
BMW	2016		2021
Ford	2019		
Honda	2016	2020	
Mercedes-Benz	2016		
Nissan	2016	2018	2020
Tesla	2015		2018

in poor visibility, detecting smaller and more subtle obstacles, are more reasons for less traffic accidents [18].

Difficult vehicle maneuvers such as parking and lane change would be simplified making them less stressful and require no special skills. Voice instructions could be used to command to the car.

2. *Improvement in traffic and fuel efficiency*

Automated vehicles will result in less probability of accidents caused by human error, leading to less traffic congestion. In general, an increase in autonomous service such that offer Uber will help increase the number of self-driving taxis and in turn help decrease the total number of cars on the road, alleviating the overall traffic. Since autonomous vehicles are designed to optimize efficiency in acceleration and braking, they will also help improve fuel efficiency and reduce carbon emissions. It is thought that the adoption of autonomous cars could reduce CO₂ emissions produced by cars by as much as 300 million tons per year.

3. *More free time will be available*

It can be imagined that, with cars doing most or all of the driving, drivers will be free to make better use of their time spent in the vehicle. Instead of spending all the time being vigilant about the vehicle’s surrounding, if at all, drivers will only take control of only when necessary or when they choose to do so.

Moreover, due to the expected decrease in traffic congestion, it will likely take less time to get to the intended destination, which will lead to the creation of more valuable time for other things besides commuting. In the long run, commuters worldwide could save a combined 1 billion hours every day once autonomous vehicles become the main means of transport.

Among other benefits that come with autonomous vehicles, it is possible to sleep during long journeys and thereby reducing fatigue due to driving. Driverless vehicles will eliminate arguments between drivers on the road resulting in the reduction of road rage incidents. The list of such minor benefits that sum to the overall well-being of society is endless.

4. *Increased speed limits*

With well-planned and sophisticated autonomous driving infrastructure available, speed limits on major highways can be increased to reflect the safer driving and thereby shortening journey times.

5. *Improved way of life*

People who traditionally have difficulties with driving, such as disabled persons, older citizens, as well as the very young people, would be able to experience the freedom of car travel. No assistance will be needed for people with physical disabilities such as vision. The requirement to take drivers' licenses before using vehicles or driving tests will be a thing of past. This will lead to reduce time required to use vehicles and hence improved lifestyle.

6. *Reduction in insurance premiums*

Autonomous vehicles could bring about a massive reduction in insurance premiums for car owners resulting in great savings.

7. *Self-aware cars would lead to a reduction in car theft*

Thieves will have no incentive to steal automated vehicles since they can be easily tracked.

8. *New job opportunities for all sectors of society*

With autonomous vehicles, new job opportunities will be created in the software design and engineering. More players can also join the automotive industry leading to competition and reduction in the cost of cars.

With all the above benefits and others, some downsides all exist with autonomous vehicles. In the beginning, driverless cars would likely be out of the price range of most ordinary people but with mass production, prices will go down in the coming 20 years. Truck drivers and taxi drivers will lose their jobs, as autonomous vehicles take over but they can take the role of managing vehicles. A computer fault or software bug could cause a severe crash as was the case with Tesla where the autonomous system failed to recognize a crossing truck at an uncontrolled intersection. Invasion of privacy is another issue since cars would rely on the collection of location and user information. There is also the possibility that hackers could get access to the vehicle's software and controls the vehicle operation remotely. These issues are currently being addressed through the use of secure protocols.

Although there are still areas to be addressed as outlined above, society stands to benefit greatly from the introduction of autonomous vehicles. Like every new technology, challenges will always exist but the benefits by far outweigh the disadvantages. Investment in sensing technology, road infrastructure, and security will be important looking in the future.

9.4.3 Future Directions and Conclusion

As outlined in this section, the automotive industry will experience rapid growth in the coming 20–30 years. Some of the advanced technologies are already on the market, although in limited driving situations, such as highways. One of the key aspects of this growth will be to introduce sensing technology that will both save lives and enhance driving comfort. The growth will result in more complex radar signal processing application, especially for object detection, tracking, and recognition systems for autonomous driving. The ability to sense and predict the dynamics of the immediate surrounding of the vehicle will be crucial in all automated driving systems.

References

1. Schiementz, M.: Postprocessing Architecture for an Automotive Radar Network. Cuvillier Verlag (2005)
2. Continental SRR-20X Industrial Sensor Data Sheet, <https://www.continental-automotive.com/getattachment/e98ad36c-49d6-400f-8b99-32ff1ef4e45b/SRR20X-Datasheet-EN.pdf.pdf> (2012). Accessed 17 Mar 2019
3. ERTRAC Task Force: Automated Driving Roadmap: Connectivity and Automated Driving. ERTRAC, pp. 1–16 (2017)
4. Underwood, S.: Automated, Connected, and Electric Vehicle Systems, Expert Forecast and Roadmap for Sustainable Transportation. Institute for Advanced Vehicle Systems, University of Michigan, pp. 75–97, (2014)
5. NanoRadar CAR70 Radar Sensor Data Sheet: <http://www.nanoradar.cn/english/pdf/car70/car70bps.pdf> (2017). Accessed 17 Mar 2019
6. Hasch, J., et al.: Millimeter-wave technology for automotive radar sensors in the 77 GHz frequency band. *IEEE Trans. Microw. Theor. Tech.* **60**(3), 845–860 (2012)
7. Rohling, H.: HProject Final Report, ATRAC: Advanced Radar TRacking and Classification for enhanced road safety, <https://trimis.ec.europa.eu/sites/default/files/project/documents/9730/final1-artrac-final-report-final.pdf> (2014). Accessed 17 Mar 2019
8. Schubert, E., et al.: Target modeling and deduction of automotive radar resolution requirements for pedestrian classification. *Int. J. Microw. Wirel. Tech.* **7**, 433–441 (2015)
9. Stolz, M., et al.: Multi-target reflection point model of cyclists for automotive radar. In: European Radar Conference (EURAD), Nuremberg (2017)
10. Lübbert, U.: Target Position Estimation with a Continuous Wave Radar Network. Cuvillier Verlag (2005)
11. Tu, N.: Automotive radar system architecture and evaluation with EEs of solutions. In: Automotive Radar Seminar, Taiwan (2016)
12. Fölster, F., Rohling, H.: Signal processing structure for automotive radar. *Frequenz* **60**(2), 20–24 (2006)
13. Wu, T.: Pedestrian and Vehicle Recognition Based on Radar for Autonomous Emergency Braking. SAE Technical Paper, No. 2017-01-1405 (2017)
14. Hamdane, H., et al.: Issues and challenges for pedestrian active safety systems based on real world accidents. In: Accident Analysis and Prevention, pp. 53–60. Elsevier (2015)

15. Zador, P.L., Krawchuk, S.A., Voas, R.B.: Final Report—Automotive Collision Avoidance System (ACAS) Program. National Highway Traffic Safety Administration (NHTSA), Aug 2000
16. Kantei: Public-Private ITS Initiative/Roadmaps 2018. https://japan.kantei.go.jp/policy/it/2018/2018_roadmaps.pdf, pp. 74–82, 15 June 2018. Accessed 17 Mar 2019
17. Bertocello, M., Wee, D.: Ten Ways Autonomous Driving Could Redefine the Automotive World. <https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/ten-ways-autonomous-driving-could-redefine-the-automotive-world> (2015). Accessed 17 Mar 2019
18. Goodman, P.: Advantages and Disadvantages of Driverless Cars. <https://axleaddict.com/safety/Advantages-and-Disadvantages-of-Driverless-Cars> (2019). Accessed 17 Mar 2019