# Value of Machine Learning and Cognition on Target Tracking

Sebastian D. Rodriguez

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Electrical Engineering

R. Michael Buehrer, Co-chair
Daniel J. Jakubisin, Co-chair
Mary Y. Lanzerotti

May 6, 2022
Blacksburg, Virginia

Keywords: Machine Learning, Cognitive Radar, Target tracking

# Value of Machine Learning and Cognition on Target Tracking

Sebastian D. Rodriguez

## ABSTRACT

In recent years previously restricted radio-frequency spectrum has been opened to civilian and industrial access in the United States. Because of this, high priority users such as the military and government need to develop systems that can adapt to the surrounding spectral environment which will suddenly be filled with new users. This thesis considers an environment with one tracking radar, a single target, and a communications system that can passively interfere with the radar system. Three separate agents, Sense and Avoid, Machine Learning, and "Optimal", are tasked with the channel selection problem for radar-communications coexistence. Each agent is evaluated based on their ability to detect and avoid the interferer while also tracking a target accurately. In particular, in this thesis, we are interested in the value that machine learning algorithms can provide over and above simple approaches. This value is assessed based on the conflicting requirements of avoiding interference yet using as much of the spectrum for tracking as possible.

# Value of Machine Learning and Cognition on Target Tracking

Sebastian D. Rodriguez

## GENERAL AUDIENCE ABSTRACT

With a newfound dependence on wireless transmission, the demand for electromagnetic spectrum allocations has vastly increased. In recent years the Federal Communications Commission has auctioned some previously restricted access frequency bands to public and commercial applications. While this enables the growth of faster and more widespread civilian communications, military radar systems which had been the priority users of those bands are now at risk of interference from new users. Current radar systems typically occupy fixed bands and are not yet well adjusted to sharing their allocated spectrum with other users. Cognitive radar systems have been proposed to monitor airwaves for potential interferences and autonomously manage band allocation to avoid the interferers.

In this thesis, we study a learning algorithm that enables a radar system to actively monitor and select its bandwidth to ensure proper target tracking. In particular, we are interested in the value this learning algorithm can provide over and above simple approaches. This value is assessed based on the conflicting requirements of avoiding interference yet using as much of the spectrum for tracking as possible.

*Dedicated to my parents*

# Acknowledgments

First and foremost, I would like to thank my advisor, Dr. R. Michael Buehrer, for introducing me to the amazing realm of wireless communications topics that I have been lucky enough to learn about through my graduate career. Dr. Buehrer's commitment to his students and his incredible ability to motivate and guide them to achieve their goals has shown me what an exceptional leader and role model he is and I have learned so much from him in my time at tech.

A very special thank you to Dr. Daniel Jakubisin whom I have been lucky enough to have had as a co-advisor. His commitment to my work and research has guided me incredibly throughout the entire process. I do not believe I could have made it this far with the thesis without his help.

My mom and dad have always been the most supportive of me and I wouldn't have made it here without them, I owe so much of my success to them. To my sister Ale, thank you for always matching my energy and being my best friend. Thanks to everyone from the Wireless@VT lab; especially Tanner, Sam, Charlie, Will, Don, Megan, and Benny for all the great times and laughs getting through grad school. To Smith and Sakib, thanks for supporting me and making me smile through some of the toughest times. To Anika!, thanks for putting up with me **always** staring at a computer while writing this and for matching my level of obsession with cats.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Executive Summary

The radio frequency spectrum is a valuable asset to many private sectors, with applications such as satellite communication, ground radar navigation, cellular, WiFi, weather radar, and landscape imaging. The recent spectrum auction [2] and reallocation has opened these once restricted-access frequencies to civilian and industrial applications. While this improves and eases the burden of cluttered frequency allocations for civilian technology, it drastically motivates the need for effective spectrum sharing technologies. There has been a recent push for *cognitive radio* systems which have the ability to sense the surrounding airspace for other users and actively adjust its parameters and settings to maximize functionality for its primary user while coexisting with other communications systems.

*Cognitive radar* has recently emerged as a useful solution to communications coexistence problems faced by radar systems in the recent years. Traditional radars were designed by assuming information about a broad range of expected targets rather than any one specific target. This was done to ensure that the radar system could detect most, if not all, targets within its operating range with some expected amount of accuracy. These assumptions were made to estimate range, signal-to-noise ratios, and target clutter characteristics. However, this design methodology led to static and fixed parameters that could not be easily adjusted (potentially new hardware would need to be swapped out to improve results). Because of

the wide variability of targets these parameters would never truly be specialized to any one specific target. Cognitive radar aims to circumvent these restrictions of traditional radar systems by having an intelligent agent that can actively adjust its parameters to perform optimally across all targets and scenes.

This work proposes evaluating the results of multiple perception-action agents in a target tracking and radar-communications coexistence problem. Here, each agent is provided with the same information received by a radar system and perceives their environment in their own way which is then used to take an action. Each action taken by the unique agents will have some effect on the target tracking performance which is then evaluated to formulate a new policy. As this process repeats continually the environment and its outcomes are recorded and studied to evaluate which agent generates the best results.

The "classical" Sense and Avoid (SAA) agent in this thesis uses a basic structure to choose actions when attempting to track a target. The SAA agent will only take the coexistence problem at face value and attempts to solve the problem by making the assumption that the action that is optimal now will be optimal in the next time step. This SAA agent also does not keep track of any of its actions taken and only generates actions based on the most recent observation.

The "optimal" agent studied in this thesis is assumed to be one that has had sufficient time to learn everything there is to learn about the communications interferer. Combining the reward structure and known probabilities of the interferer the Bellman Optimal policy can be found which dictates exactly what actions must be taken to maximize radar performances.

Finally the "ML" agent presented in this work uses an online reinforcement learning algorithm to learn actions which mitigate interference at the radar and minimize target tracking error using a specified reward function. The online aspect of the model is ideal for a radar

system due to the sequential nature of information at a radar receiver. This online optimization enables the radar with memory and adaptability to make smarter decisions as time goes on, as opposed to SAA which does not improve over time. Furthermore, the reward structure of the ML agent is flexible and allows for variability in the way the agent makes decisions, allowing a user to either emphasize interference avoidance or bandwidth maximization.

## 1.2 Thesis Overview

Chapter 2 discusses the fundamentals of radar physics, the radar wave equation, and the main functions of radar and its applications.

Chapter 3 presents the topics of machine learning in radar such as convolutional neural networks and reinforcement learning as well as their applications in recent years. The Bellman optimal agent is also discussed and expressed as a maximization of the reward and value function in reinforcement learning.

Chapter 4 begins the discussion of the system model and the setup of the radar environment. Radar parameters used in this thesis are described and the communication interferer state space is defined. Transition matrices and specific metrics for evaluating machine learning are also described such as the ”diagonality” and entropy values.

Chapter 5 presents the results of the research done in this thesis. Several experiments are demonstrated to support the value for machine learning such as: The Channel Selection Problem, Subbands with Varying Levels of Activity, Improper Thresholding, and Spatial Correlation. The results demonstrate that using a machine learning agent the radar system can, accurately select subbands given proper reward functions, avoid subbands that are highly occupied, detect signals that are incorrectly reported, and learn the relative location

of an interferer within an area.

# Chapter 2

# Background

## 2.1 Radar

**R**adio **D**etection **A**nd **R**anging also known as **RADAR** is a system that uses the transmission and reception of radio waves to estimate information about a target. Christian Huelsmeyer's 1904 patent for the Telemobiloscope was the first true radar system that could detect the presence of a metal object from a distance; however it was not capable of assessing range. This first radar design took advantage of a fundamental characteristic present in all waves seen in nature, reflection. Specifically Huelsmeyer transmitted an electromagnetic (EM) radio-frequency wave that was not fully absorbed by the metal object, allowing his receiver to detect this reflection and confirm that a target was present. [3]

All radar systems must be comprised of at least one antenna, one transmitter, and one receiver [4]. In a monostatic radar system both the transmission and reception of the radar waves will be handled by a single antenna. A bistatic radar system is compromised of individual receive and transmit antennas separated by a distance similar to their expected target distance. However beyond this, many more antenna elements may be used. In this case these may be placed at strategic locations whether in a colocated or sparse array configuration. All of these configurations, while advantageous in certain scenarios, come with some tradeoff costs and specific parameters must be tuned in an attempt to optimize these systems.

## 2.2 Physics and the Radar Range Equation

A wave in nature is defined as any propagating disturbance that oscillates either in time or distance. There are two main waves typically studied in physics, mechanical and electromagnetic. For the research conducted here only electromagnetic waves are considered for radar applications for several reasons including:

1. Disregard for a propagation medium (important when dealing with objects in space)
2. Extremely fast travel time ($c \approx 3 \times 10^8$ m/s in a vaccuum, $99.97\% * c$ in air)
3. Relatively low propagation losses
4. Low cost and high efficiency relative to certain systems such as loud speakers and mechanical oscillators.

An electromagnetic wave is comprised of a pair of perpendicular oscillating electric and magnetic fields which are given by the following expressions:

$$\vec{E}(x,t) = \hat{i}E_{max}(kx - \omega t + \phi) \tag{2.1}$$

$$\vec{B}(x,t) = \hat{j}B_{max}(kx - \omega t + \phi); \tag{2.2}$$

however, for analysis we only consider the electric field component. These two fields in conjunction travel through space together, and their direction is defined by the cross product of the two waves (Equation 2.1 and 2.2). In both of these equations $k$ is known as the wavenumber, defined by $k = \frac{2\pi}{\lambda}$. $\lambda$, or wavelength, is the distance traveled by a wave in a single period, and $\phi$ is the phase offset of the wave. $\hat{i}$ and $\hat{j}$ are the unit vectors in the direction of propagation of each wave; they are perpendicular to each other and generally are

represented by the X and Y-axes respectively in 3-dimensional space. Although wavelength is difficult to measure directly without the use of special equipment, the expression $V = \lambda f$ is a fundamental relationship that is used to estimate wavelength given information is known about the velocity and frequency of the wave. Since EM waves generally travel at or close to the speed of light (c) the equation simplifies to $\lambda = c/f$. Finally $\omega$, or angular frequency, is the rate at which the wave completes a full cycle defined as,

$$\omega = \frac{2\pi}{T}, \tag{2.3}$$

where $T$ is the period of the wave, $T = 1/f$.

The interaction of EM waves with background clutter and targets varies with frequency and power of the transmit signal [4]. It is imperative to this research to properly model how these EM waves propagate through free space and what losses may be incurred given the parameters of the wave. An equation that relates known wave, antenna, and target parameters to the power at the receive antenna is given by

$$P_R = \frac{P_t G_t G_R \lambda^2 \sigma}{(4\pi)^3 R^4 L_{total}}, \tag{2.4}$$

similar to the Friis Transmission Equation, which can be rearranged to derive the Radar Range Equation (RRE) [4]:

$$R_{max} = \sqrt[4]{\frac{P_t G_t G_R \lambda^2 \sigma}{(4\pi)^3 L_{total} S_{min}}} \tag{2.5}$$

where

$P_R$ = Received power

$P_t$ = Transmit power

$G_t$ = Transmit antenna gain

$G_R$ = Received antenna gain

$\lambda$ = Wavelength

$\sigma$ = Radar Cross Section

$R$ = Range to target

$L_{total}$ = Total losses due to issues air and its separate mediums.

In Equation 2.5 $P_r$ is exchanged for $S_{min}$, minimum detectable signal power, because for a maximum operating range it is only important to know that our signal is above the minimum noise floor of our radar system. This noise floor for typical RF frequencies and reasonable temperatures is defined as $P_{noise} = kTB$. In this equation $k$ is Boltzmann's constant ($k = 1.3806 \times 10^{-23}$), $T$ is temperature in Kelvin, and $B$ is the bandwidth of the receiver. As the bandwidth occupied by a radar system increases, its noise floor increases, increasing the necessary $P_r$ needed to properly detect a signal. $L_{total}$ is largely dependent on the propagation losses of waves through various mediums such as rain and clouds. Free space path loss, $L_{path}$, can be expressed separately as follows

$$L_{path} = \frac{P_r}{P_t} = \left(\frac{4\pi R^2}{\lambda}\right)^2 = \left(\frac{4\pi R^2 f}{c}\right)^2. \tag{2.6}$$

Equation 2.6 demonstrates that waves with greater frequencies will suffer from higher path losses, this is important in radar where frequencies can get very high (>10 GHz).

## 2.3 Radar Waveforms

Radar waveforms can be divided into one of two major classes, each with their own advantages and disadvantages. The first, although important to explain, is not used as the primary method in this research; it is known as Continuous-Wave (CW) radar. Continuous-Wave radar is advantageous due to the fact that it is, as the name implies, continuously transmitting and receiving. Because of this, a lower transmit power is used to prevent damage at the receiver. CW radar also interferes much less with wireless systems due to the low power; however, this also decreases the potential signal-to-noise ratio at the receiver. A CW radar enables simple computation of a target's velocity because the Doppler shift of the receive signal can be easily compared to the waveform that is still being transmitted. The faster a target is moving towards or away from the transmitter the greater the Doppler shift will be at the receiver. This effect is experienced by many people daily, with greater changes in frequency of an ambulance the faster they approach and drive by the listener. Due to the low power signal of a CW radar the range is greatly reduced (Equation 2.5). Another disadvantage to CW radar is the difficulty of determining a target's range. If the CW radar is only transmitting a single frequency then it is unable to differentiate when the transmit signal has reached the target. If a CW radar transmits a frequency sweep then range may be determined by comparing what time has passed since the received frequency was transmitted; however this is still challenging. The final disadvantage of CW radar is its inability to differentiate several targets in space from one another; the CW system will confuse them all for each other, and there is no simple way to overcome this.

The second radar class is Pulsed radar, in which EM waves are transmitted on timescales

ranging from milliseconds to fractions of nanoseconds. Pulsed radar works by transmitting these pulses and receiving the echos that are reflected from a target. Pulsed radar is used for most radar applications since it allows for velocity and range measurements. Pulsed radar is very modular, and measured parameters such as range resolution, maximum range, range ambiguity, and maximum number of targets can all be determined by adjusting certain parameters of the radar system. Transmit power, $P_t$, in pulsed radar can be significantly higher due to the low duty cycle of pulses. $P_{avg} = P_t * duty\%$, where duty% is the ratio of pulse time to PRI. This enables a much higher signal-to-noise ratio (SNR) at the receiver when compared to CW radar.

## 2.4 Measured Parameters

A few of the main problems that radar aims to solve are target detection and tracking, however to do so information must be acquired about a target. A typical radar system will be able to estimate these parameters about a target given it has knowledge on its waveform characteristics and can process the receive echos: [4]:

- Range
- Velocity
- Azimuth Angle $\theta$
- Elevation Angle $\phi$

A target's range is estimated from the propagation time between a transmitted pulse and its received echo. Since c is constant, calculating the range to a target is defined by

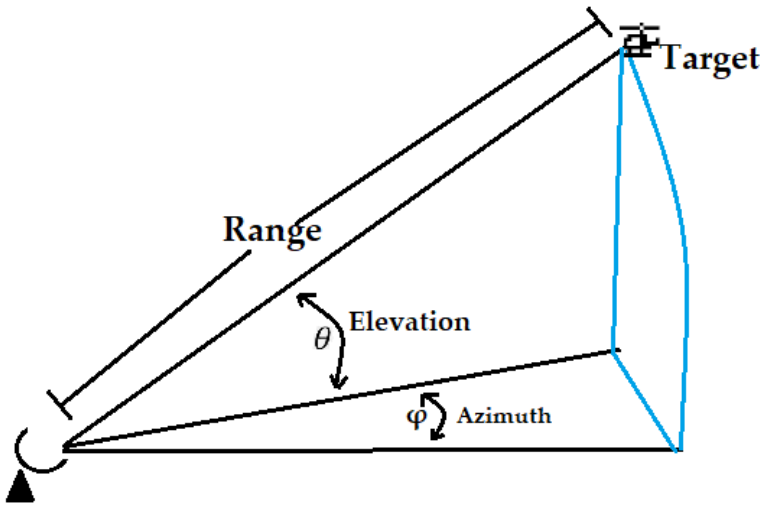


Figure 2.1: Angles of Elevation, Azimuth, and Range to the target being investigated

$$R = \frac{c\Delta T}{2}, \tag{2.7}$$

where $\Delta T$ is the measured time in seconds from when a pulse was transmitted to when an echo was received. In a radar system $\Delta T$ may be estimated by storing the time a pulse was transmitted and comparing that to the time a pulse is received.

A target's azimuth and elevation angles can be determined from the location of the antenna beam as it tracks the target. As previously discussed in Section 2.3, the Doppler effect is critical in determining the velocity of a target. Whether a target is moving towards or away from the radar, the return echo will incur some Doppler Shift ($f_d$) that can be measured. With knowledge of the wavelength of the transmit pulse and the angle between the target direction and radar antenna ($\delta$); the target's velocity ($v_t$) is given by

$$v_t = \frac{f_d \lambda}{2\cos\delta}. \tag{2.8}$$

In a scenario where several targets are positioned close to one another there may be a need to properly distinguish each target accurately. Parameters in the radar system can be adjusted to account for several targets located close together. Range resolution, or the minimum range between targets that a radar system can distinguish is defined by

$$S_r \geq \frac{c\tau}{2}, \tag{2.9}$$

where $\tau$ is known as the Pulse Width, the time it takes for a single pulse to be transmitted. Other important values to consider are the Pulse Repetition Frequency (PRF) and Pulse Repetition Interval (PRI), the former being how often the pulse is transmitted and latter being the total time from the start of one pulse to the start of the next. The relationship between the two is defined as

$$PRF = \frac{1}{PRI}. \tag{2.10}$$

PRI differs from Pulse Width in that it includes both the uptime of the pulse and the downtime given to receive an echo.

Unambiguous range for a radar, defined as

$$R_{ua} = \frac{c}{2 * PRF} = \frac{c * PRI}{2}, \tag{2.11}$$

is the maximum range at which a target can be located to guarantee that its echo corresponds to the most recent transmit pulse. Targets beyond this range will still be detectable; however they will appear closer to the radar system than they actually are because their return echo will have arrived after the allotted PRI time. Although this aliasing may occur, it is not impossible to generate an estimate of the target's range. With more pulse processing, a target who lies beyond the unambiguous range can have its range estimated.

The maximum detectable Doppler Shift,

$$f_{d_{max}} = \pm\frac{PRF}{2}, \tag{2.12}$$

is the highest frequency shift that a radar system will be able to detect. This is also a limit on the maximum velocity a radar system can detect. Targets that are travelling faster than this maximum velocity will be considered aliased due to the Nyquist Sampling Theorem, and the radar system is no longer able to properly interpret the frequency of the returned echo.

It may be apparent that both of these parameters, while tunable by adjusting the Pulse Repetition Frequency, actually are inversely related with respect to the PRF. If a large PRF is selected then a high velocity resolution will be attainable; however this may only permit a low range unambiguous range estimation. A large pulse repetition frequency will give improper range measurements in targets at long distances. Conversely, a low PRF decreases potential velocity resolution but increases the maximum range at which a target can be detected. These are some of the first basic interactions where the selection of proper radar parameters will be imperative to problem that needs to be solved by the radar system.

## 2.5 Radar Detectors

### 2.5.1 Neyman Pearson

A Neyman Pearson (NP) detector is the simpler of the two basic radar detector models that can be applied. In this case a static threshold is employed to compare to our receiver data, this can be complex or real-valued. The two main criteria in an Neyman Pearson detector are:

- $P_{fa}$, Probability of False Alarm

- $P_d$, Probability of Detection

It may be clear that for a target detection problem a $P_{fa} = 0$ and $P_d = 100\%$ is an optimal solution; however, these specifications are unreasonable goals in real world radar where false alarms may always be present. The goal of the NP detector is to maximize the $P_d$ for a desired $P_{fa} = \alpha$, where alpha is the level of significance that is chosen for the expected $P_{fa}$. This begins by making an assumption of two hypotheses and testing the data received against both hypotheses to determine the truth.

$$H_0 = \tilde{\xi}(x) \tag{2.13}$$

$$H_1 = Y(x) + \tilde{\xi}(x) \tag{2.14}$$

Equation 2.13, the null hypothesis, states that the received signal is comprised of solely noise, $\tilde{\xi}(x)$, with no radar echo present. Conversely, Equation 2.14 is the alternative hypothesis that suggests that there may be a signal present. The fact that noise will always be present even in the alternative hypothesis should make it apparent that detection without false alarms is an unreasonable goal.

First, let X $= (x_1, x_2, ..., x_n)$ be random samples from the normal distribution $\mathcal{N}$. The conditional probabilities $p(x|H_0)$ and $p(x|H_1)$ are then defined by

$$p(x|H_0) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x}{\sigma}\right)^2} \tag{2.15}$$

$$p(x|H_1) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu_1}{\sigma}\right)^2}, \tag{2.16}$$

where $\mu_1$ represents the mean value of the radar echo. Suppose then a test is made for checking the null hypothesis $H_0 : \theta = \theta_0$ against $H_1 : \theta = \theta_1$ the alternative. There exists some subset $M$ of the sample space $X$ defined by $[M|x_m \leq m < \infty]$, such that

- $P((x_1, x_2, ..., x_n) \in M|H_0) = \alpha$
- $\frac{\mathcal{L}(\theta_0)}{\mathcal{L}(\theta_1)} \leq k$, for each point $(x_1, x_2, ..., x_n) \in M$
- $\frac{\mathcal{L}(\theta_0)}{\mathcal{L}(\theta_1)} \geq k$, for each point $(x_1, x_2, ..., x_n) \in M'$

where $\mathcal{L}(\theta)$ is the joint probability distribution function ($\mathcal{L}(\theta) = f(x_1|\theta)f(x_2|\theta)...f(x_n|\theta)$). Then it is said that $M$ is the best critical region of size $\alpha$ for testing $H_0$ versus $H_1$. This region $M$ maximizes the probability of detection and constrains the probability of false alarm to the desired size $\alpha$ such that

$$\max P_d = \int_M p(x|H_1)\, dx, \tag{2.17}$$

and

$$\min P_{fa} = \int_M p(x|H_0)\, dx = \alpha. \tag{2.18}$$

Once the region M is established, all following data collected that lies within M will be classified as a detection and everything below M will be denoted as noise. For a set of data with high Signal-to-Noise ratio there will be little overlap of the two distributions allowing for much smaller $\alpha$ values because the signal power is much greater than the noise. However, in a more realistic case, distributions as seen in Figure 2.2 will be observed.

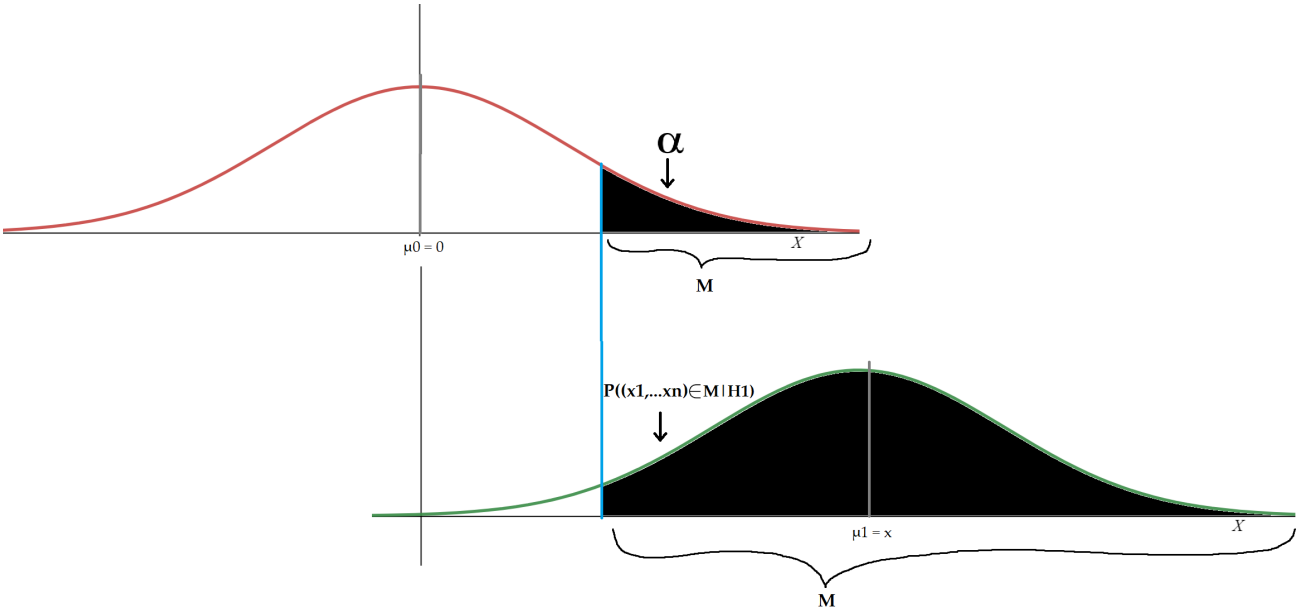


Figure 2.2: Conditional distributions under the null hypothesis $p(x|H_0)$ (red) and the alternative hypothesis $p(x|H_1)$ (green). The region M is the same for both distributions and continues to infinity.

### 2.5.2 CFAR

**C**onstant **F**alse **A**larm **R**ate, or CFAR is an algorithm similar to the Neyman Pearson detector, in that it also sets a threshold that must be passed to reject a noise hypothesis. However, the main difference between CFAR and NP is that the former is an adaptive algorithm that actively changes the threshold based on the surrounding noise floor. This avoids the issues when facing real world noise profiles that are not just additive white Gaussian noise, which are assumed in the NP detector.

The algorithm begins by assuming there is always a desire to keep the power required for detection to be high enough relative to the noise floor. This ratio of noise floor to threshold is known as $\alpha$, as seen in 2.19. The threshold factor then changes as noise varies, unlike the constant set M in the NP detector. This threshold factor allows the resulting probability of false alarm to be kept constant while also guaranteeing a strong probability of detection.

$$Threshold = \alpha P_n \tag{2.19}$$

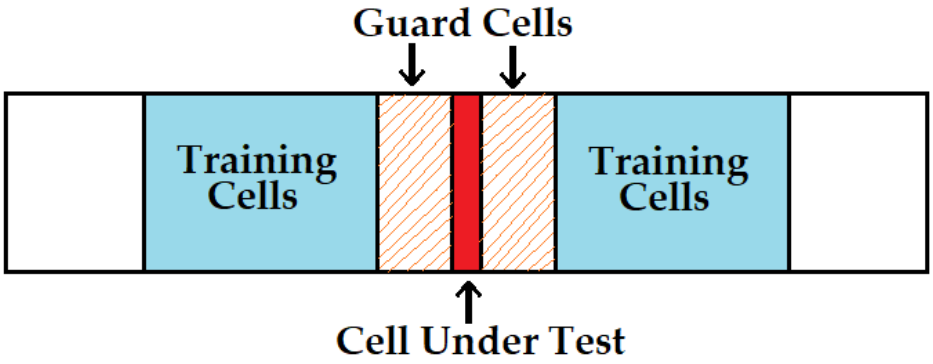


Figure 2.3: Sample figure for training cells and cell to be approximated. [1]

To determine a moving threshold a predetermined amount of training cells are isolated around the cell under test (CUT). Once the noise floor is determined the threshold of de-

tection is set and any values above the threshold will count as detections. This enables the radar system with a moving threshold that can detect a target regardless of noise profiles.

## 2.6 Radar Applications

Although radar is typically believed to be only a military application, there are several areas where radar can is applied. The following is a short list of applications [4].

1. Military Application

    - Search Radar: Detects and tracks targets.
    - Over-the-horizon Search Radar: Utilizes the advantage of the refractive effect in the ionosphere to detect targets at extremely long ranges, up to thousands of miles around the earth.
    - Ballistic Missile Defense Radar: Searches large angular volumes and capable of tracking low-RCS, fast-moving targets. However, debris and decoys create challenges for these types of radar.
    - Radar Seekers and Fire Control Radars: Many air-to-air and air-to-ground missile systems are designed to attack threats by detecting and tracking the target of interest. The missiles follow the reflected signal from a target that is being "illuminated" with RF signals. [4]

2. Commercial Applications

    - Process Control Radar: A method of measuring the amount/level of fluid inside of a tank or "dryness" of a product. Typically utilizing frequency modulated

continuous wave (FMCW) at higher frequencies (10 GHz) to measure the distance down to the top of the fluid.

- Airport Surveillance Radar: Detects and tracks commercial and general aviation aircraft. Typically these rotate mechanically in azimuth while employing wide elevation beamwidths. A beacon transponder on the aircraft will report the flight number and altitude back to the surveillance radar.
- Weather Radars: Measures the reflectivity of precipitation to obtain rainfall rate, uses Doppler techniques to obtain wind speed, and spectral width to measure turbulence. Some weather radars can use polarization characteristics of precipitation to discriminate between rain and hail, and others use Doppler techniques to measure wind shear, and rotating atmospheric events.
- Wake Vortex Detection: Large, heavy aircrafts generate wake vorticies and turbulence behind them that may persist for a long time, thus posing a threat to smaller, lighter aircrafts. Aircrafts taking off and landing are separated by certain amount of time to allow the turbulence to dissipate. Radars placed at the end of runways can sense this turbulence and generate a warning for dangerous conditions.
- Satellite Mapping Radars: These systems typically operate from satellites in orbit around 770 km altitude. Pulse compression and Synthetic Aperture Radar (SAR) techniques are employed to obtain good range resolutions. Satellites have the advantage of unobstructed views of the Earth, can operate at night and in poor weather conditions.
- Police Speed Radar: Utilizes continuous wave (CW) transmissions to measure the Doppler shift of a moving vehicle, which then is used to calculate velocity.
- Automotive Collision Avoidance Radar: Currently deployed in some cars; utilizes

millimeter wave radar to scan the road for targets that may pose a risk of collision.

- Radar Altimeter: Simple FMCW radars are employed to determine the height of an aircraft above ground level. A reflection will be received from the ground surface when the radar is pointed directly downward and the range to the ground is the altitude of the aircraft.

# Chapter 3

# Introduction to Cognitive Radar and Machine Learning

## 3.1 Machine Learning

Machine learning and machine intelligence was initially researched as early as the 1940's, in an attempt to draw connection between statistics, information theory, and the physical brain. It is understood that species on Earth have been able to evolve and become increasingly more adaptive to their surroundings. The pursuit to apply this adaptability and learning to data was key to progressing research. In 1949, Donald O. Hebb introduced his first theory of modeling brain cell interactions. Hebb described his theory as "When one cell repeatedly assists in firing another, the axon of the first cell develops synaptic knobs (or enlarges them if they already exist) in contact with the soma of the second cell" [5]. This interaction noted in Hebb's Theorem is believed by many to be the first iteration of neural networks, summed up by the quote "Cells that fire together wire together" [6].

The phrase "Machine Learning" is believed to have first been used by Arthur Samuel in his research conducted in the 1950's. At the time, while working for IBM, Samuel developed machine learning algorithms that were trained to play checkers against an opponent [7]. Since computers at the time were severely limited by their memory and speed, it was not

feasible to pre-compute all possible moves and board arrangements for checkers, therefore a new method had to be devised. The computer given only "the rules of the game, a sense of direction, and a redundant and incomplete list of parameters which are thought to have something to do with the game, but whose correct signs and relative weights are unknown and unspecified" [7] was able to play checkers at a level better than the creator of the program. This breakthrough in machine learning was one of the first applications of reinforcement learning, reward structures, and learning rate [8]. The ability to design models and train computers to perform everyday tasks has been studied extensively in the recent years leading to advancements in the ways technology is encompassed in every day life.

### 3.1.1 Convolutional Neural Networks

Convolutional Neural networks (CNN's or ConvNets) are one of the most widely used forms of machine learning in the field of image classification. CNN's, as a form of Supervised Learning [9], work by taking an input image and extracting features (much like a human perceives images) and then assigning weights to several aspects it believes is important to defining the images. Later, when a similar image is encountered, the model should be able to correlate a previously unseen image with a new one, identifying it as being the same or similar; or completely different [10]. Issues encountered with CNN's at times are that data sets generally need to contain a large amount of images to enable proper classification and problems that just do not have sufficient data may suffer from CNN approaches. CNN's, although proposed early in the years of machine perception and intelligence, took a serious stride with Graphical processing units being widely available in most computers in the early 2000's. Several researchers had shown early on that processing CNN's on GPU's had exceptional advantages over previous dedicated hardware [11, 12]. Although CNN's have much value in image classification there have been large strides in the field of wireless

communications where CNN's have been shown to solve many current issues.

In Martorella et al. the usage of a Neural Network is essential to the atuomatic target recognition problem (ATR) being tackled by the team [13]. In this paper they use a neural network to extract features from Inverse Synthetic Aperture Radar images to classify targets. The feature identification begins with a preprocessing of the ISAR data by extracting the brightest scatterers using a Pol-CLEAN algorithm (Polametric-CLEAN). ISAR is a radar technique similar to SAR except that the latter uses the movement of the radar system to generate a model of the target while the former generates a model from the movement of the target itself. ISAR is strategically used in maritime surveillance radar due to the constant movement of the water enabling the target to move consistently over time, alternatively this is difficult on land because the ground does not shift like the ocean does; SAR systems are preferred here. In the paper the authors study the ability of their model to learn features of their targets and classify them with background noise. The model is trained on few images of a small set of targets in an anechoic chamber using a horn antenna. While this paper shows moderate results to their work it falls short in common issues seen in CNN's, small datasets resulting in poor classification and severely over-fit data since they would not be able to classify any other target aside from the ones on which they had trained.

[14] Bentes and Velotto present a novel application of Deep Neural Networks to automatically classify oceanographic objects. The authors also utilize the approach of synthetic aperture radar imaging because the data collected is capable of providing detailed information about the ocean's surface and floating structures such as ships, icebergs, and oil rigs. Constant False Alarm Rate radar algorithms are generally employed to detect objects found in radar scans, and with proper thresholding objects can be successfully distinguished from background noise. The issue the authors aim to tackle is that once a target is detected by

CFAR algorithms, how can a system identify the target without human intervention. Here a CNN is built to classify SAR images of the targets detected out at sea. Prior to being used as training data all images are passed through a denoising autoencoder which is a special Neural Network tasked with recreating a sample input image when given a noisy version of that image. This autoencoder provides cleaner images to the main CNN which classifies the images by extracting features from the SAR images and pools them after every feature layer; this effectively reduces dimensionality and decreases computation time.

Molchanov P. et al. have developed a novel approach to classifying ground moving targets using a multilayer perceptron (MLP) [15]. Issues typically found with survelliance radar are low velocity or stationary targets of interest providing only micro-Doppler measurements which make it difficult to detect and classify a target. The author proposes a method to analyze these micro-Doppler signatures generated by a single human walking across vegetation clutter and extract features for object recognition and noise rejection. The received radar data is first preprocessed using a discrete cosine transform (DCT) to generate coefficients for the MLP. A DCT has similar applications in signal processing to a Discrete Fourier Transform, both of which decompose a finite length of signal samples into an array of their basis functions. The main difference between DCT and DFT is the latter specifically aims to map the input data to a set of complex exponential functions while the former uses real valued cosine functions. A DCT typically enables the representation of data with fewer coefficients than a DFT, as the author claims is a benefit for enabiling more efficient and faster processing. Once the coefficients are processed, features are extracted and selected from the data which is then split into training and testing sets to be passed to the MLP. The paper considers the approach of actual radar parameter modification, demonstrating that the classifier preforms better if longer integration time is allowed. This is an expected

result as radar without any machine learning may require a longer duration of time before being able to accurately detect a target. Finally the authors demonstrate their approach properly classifies up to three humans at a time walking through their clutter region with an accuracy of 86%.

Stroescu A. et al. proposed a solution to current issues in autonomous target detection for the automotive industry [16]. Problems when working with automotive radar typically arise from the fact that many objects are moving on the road all at once, this makes it exceptionally difficult to differentiate clutter from actual targets. Even if targets can be identified perfectly there still is a serious issue in automotive radar to be able to differentiate between pedestrians and vehicles, in rural areas this may be much simpler of a task but in a city environment this becomes exponentially more difficult. Automotive radar also does not have much time to deal with typical radar processing times that a stationary antenna system may have, especially with high priority targets such as pedestrians, decisions must be made quickly and accurately. The author begins by training the CNN on merged radar images of potential targets in a laboratory setting. This enables the classifier to be able to detect multiple targets in one pass rather than learning to pinpoint single targets at a time. The author demonstrates two separate detection algorithms, Faster R-CNN and SSD. While SSD was faster it actually detected small object significantly worse than Faster R-CNN, this trade off is overcome by the author deciding that classification accuracy is imperative to automotive radar even if the deep-CNN takes slightly longer to classify.

Shebert S. et al. presents a deep learning Convolutional Neural Network model tasked with classifying varying unknown wireless signals [17]. Advancements of this research are the ability for the classifier to identify signals without any synchronization with the unknown

signals, simply classifying them based off received In-phase and Quatrature samples. Shebert S. describes that the wireless standard classification have been less explored in recent years due to the extensive evolution of the complexity of wireless standards. This makes for great challenges when aiming to learn features about any signals because such high dimensionality would hinder computation time. Signals used to train and examine the classifier were 4G LTE downlink, 4G LTE uplink, 5G NR downlink, 5G NR uplink, IEEE 802.11ax (WiFi 6), and Bluetooth Low Energy (BLE) 5.0 signals. When generating signals to train the classifier MATLAB's wireless communications toolbox was used to generate and upsample all the signals to a similar rate, enabling for some similarity between all the signals. Signals were then also impared with Riciean or Rayleigh channels as well as additive white Gaussian noise between -10 dB and 20dB. The author's model contained three pairs of convolutional and maxpooling layers, followed by 2 pairs of dropout and dense layers. The classifier used in the paper was cappable of accurately classifying the signals up to 90% given they have a positive SNR value (using the Squeare-STFT feature extraction).

Xiaoyu Liu, Diyu Yang, and Aly El Gamal present a problem similar to the last one, also attempted with CNN's. In this work the authors investigate the ability to identify and classify wireless signal modulation using a Convolutional Long Short-term Deep Neural Network [18]. The authors use previously generated modulation datasets including BPSK, QPSK, 8PSK, QAM16, QAM64, BFSK, CPFSK, and PAM4 digital modulation and FM, AM-DSB analog modulation. These samples were then impared in GNU radio rather than MATLAB to simulate real world transmission interference and fading. Samples were consistently attenuated between -20dB and 18dB. The CLDNN network, initially developed by T.N. Sainath at Google [19], is a hybrid combination of a CNN, and LSTM which enables entire sequence processing rather than just single point data. LSTM's are an example of recurrent neural

networks that are not composed of pooling or conv layers; instead they are stacked cells, input gates, output gates, and forget gates [19]. LSTM's have shown great strides in the fields of speech and writing recognition as well as Natural Language Processing. X. Liu et al. have developed their CNN with four convolutional layers after iteratively working through the optimization problem or choosing between 2 and 5 layers. The authors explain that at too few layers it was less likely for their network to converge and that at too deep of layers there were large losses during early stages of classification. The authors demonstrate the CLDNN architectures ability to properly classify their signals up to a 95% accuracy with the majority of errors arrising from QAM16 and QAM64 being misclassified for each other due to the similarities in their constellation diagrams. Wideband FM is also misclassified as AM-DSB; this issue is explained to be possibly caused by silences in the audio signals where only carrier tones exist, this is not an issue for higher degree modulation schemes that rely on constant transitions to modulate their signals.

### 3.1.2 Reinforcement Learning

"Reinforcement learning is learning what to do so as to maximize a numerical reward signal"[20]. In a reinforcement learning based algorithm there is no label to determine what action relates to which outcome as there is in supervised learning. There is also no attempt to generate correlations in hidden features found in unlabeled datasets, like what is done in unsupervised learning. Reinforcement learning simply aims to select actions and observe states to maximize a predetermined reward function. This is a very powerful learning method in the cases where a users is not expected to manually generate all the information about every single possible state. In these cases the state space of the problem may become intractably large and hidden or unexpected actions may result in the greatest reward [20]. In

recent years reinforcement learning has been the heart of many advancements of autonomous computer decision making.

DeepMind Engineers have previously developed a reinforcement learning algorithm that, given sufficient time and only rules of the game of Go, was capable of achieving superhuman performance and outperforming all other current state of the art models [21]. While this was a significant accomplishment on its own, they took the same reinforcement model and adjusted some aspects while also giving it the new rules of chess. After several hours of self-play AlphaGo Zero was able to again achieve superhuman levels of chess playing and outperformed all other existing chess playing algorithms. Adjustments made to train their model for chess included the removal of the binary win/loss conditions since players may tie or draw, also since Go is a significantly more complex game and rotations and reflection take massive parts in how Go is computed, symmetrical board playing was used for chess. AlphaStar, also developed by Google DeepMind Engineers was the first artificial intelligence system that was capable of defeating a professional human player at the virtual video game Starcraft II [22, 23]. Having been given only the rules of the game, live screenshots of the game to analyze, and a reward function that properly matched what the player is meant to do, AlphaStar was able to succesfully learn Starcraft at an expert level. One interesting point that was emphasized by the team was that professional players that went up against AplphaStar noted that it took untraditional approaches to playing the game. Many of the strategies that the AI employs actually would not be typically used by professional players. This shows that reinforcement learning does not pick the actions that are the most suitable to the user but instead attempts to optimize a reward function by any means possible, even if by some obscure exploitation. Berner et al. at OpenAI demonstrated a similar reinforcement learning approach to artificial intelligence in autonomous video game playing

[24]. Here OpenAI Five was tasked at defeating professional Dota 2 teams by means of self-play reinforcement learning with an AI system built on an "autoregressive action decoder, with an LSTM core to handle partial observability" [24].

### 3.1.3 Markov Decision Processes

A Markov Decision Process is a stochastic decision model that chooses from a set of available actions in a state given knowledge of the current and past states. MDP's are very popular in reinforcement learning problems because the agent starts with no prior knowledge and is provided a reward when an action is made. As a decision maker explores states and "learns" from previous actions it is able to converge to a policy which provides the decision maker a "prescription for choosing action(s) at a particular time" [25]. Generally it is favorable to *value* how good it is for the agent to be in a certain state. A value function, given a particular policy $\pi$ is modeled by [26]:

$$V^{\pi}(s) = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k \mathcal{R}_{t+k} \middle| \pi, s_t = s\right] \tag{3.1}$$

where $\gamma^k R_{t+k}$ is the reward at state $s_t$ discounted by $\gamma$, a discount factor which determines whether the agent favors immediate or long term reward. Beyond just valuing current actions an MDP must also consider the probabilities of moving from one state to a future state given any action is taken, this is known as the transition function. A transition function in literature is generally seen as:

$$\mathbb{P}(s_{t+1}|s_t, a_t) = \mathcal{T}(s_t, a, s_{t+1}), \tag{3.2}$$

where $s_t$ is the current state the agent is in, and $a$ is the action that the agent has taken.

Furthermore, the value function in equation 3.1 can be expanded upon and described in terms of the value function for the future state $s_{t+1}$, discount factor $\gamma$, and transition probabilities $\mathcal{T}$, this is known as the Bellman Equation:

$$V^{\pi}(s) = \sum_{s_{t+1}} \mathcal{T}(s, a, s_{t+1}) \left(\mathcal{R}(s, a, s_{t+1}) + \gamma V^{\pi}(s_{t+1})\right)\Big|_{a=\pi(s)}. \tag{3.3}$$

The ultimate realisation of the decision making agent would be when it has converged to a policy that results in the greatest overall reward, maximizing the value function over all policies. This optimization results in an optimal policy, $\pi^*$, its value function is known as the Bellman Optimality Equation [26, 27]:

$$V^{\pi^*}(s) = \arg\max_{a \in \mathbb{A}} \sum_{s_{t+1} \in \mathbb{S}} \mathcal{T}(s, a, s_{t+1}) \left(\mathcal{R}(s, a, ss_{t+1}) + \gamma V^{\pi}(s_{t+1})\right). \tag{3.4}$$

In section 5 the Bellman Optimal agent is derived for the radar channel selection problem investigated in this paper. The agent has full knowledge of the transition probabilities from one state to another and pre-computes all possible rewards for taking actions, which it then weighs against all possible transitions and chooses the actions that return the greatest reward.

Several papers have been published in the field of machine learning for communications that use MDP's as the main approach to a centralized centralized networkd. In [28], Howard W. et al. demonstrate the application of the M-ETC-Elim algorithm on a multi-player Multi-Arm Bandit (MMAB) problem for communications channel selection. A MMAB framework is similar to an MDP in that they both model the relationship between an agent selecting actions and incurring reward to converge to an optimal policy. In this case "multi-player" denotes several decision makers that are unaware of each other and only interact by record-

ing their reward from choosing actions [29]. The agents are tasked at selecting subbands uncooperatively while also avoiding a communications interferer, if an action is taken that interferes with either the communications channel or other agents then a reward of 0 is given to that agent. The author demonstrates the ability of the M-ETC-Elim algorithm to incur near logarithmic regret when applied to this channel selection problem, also demonstrating the ability of uncooperative agents to converge to a solution if rewards are structured well. Kalathil D. et al. tackle a very similar decentralized multi-armed bandit problem for communications channel selection with each agent blind to all others in [30].

## 3.2 Cognitive Radar

In recent years a vast majority of machine learning concepts have been investigated in their potential applications to radar systems and some have been shown to provide exceptional value when properly designed. Cognitive radar (CR) is the study of designing and implementing a model able to perform a variety of functions in a radar system and learn relevant information about targets and the background [31]. Some cognitive radar functions have been described in Section 3.1 such as target classification, target tracking, spectrum sensing, beamforming, and spectrum optimization. Prior to these advancements in machine learning, all the aforementioned functions would have been traditionally associated with individual dedicated radars [31]; however, in recent years radar systems have evolved to work on multiple functions at once with significantly wider frequency bands [32]. This brings significant issues with the current rise of civilian technologies that have a high demand for wider allocated spectrum, interfering with frequencies that may have previously been reserved for private or military usage. Martone A. et al. present a design for spectrum allocation in noncooperative radar coexistence through the applications of spectrum sensing and proper

channel selection algorithms [33]. This is one of many current strides in widening the availability of civilian communications services into broader and previously restricted access frequencies.

Burgin et al. propose a novel algorithm for estimating soil moisture from the dataset generated by NASA's Soil Moisture Active Passive (SMAP) environmentnal monitoring satellite [34]. The SMAP satellite collects data by orbiting the Earth in a Sun-synchronous orbit and uses a Synthetic Aperture Radar (SAR) to map out features of climate and weather on Earth's surface. The researchers derive a linear model relationship between the radar scattering detected by SMAP and previously collected data [34]. This linear function is then fitted with proper coefficients generated from past L-band radar data to accurately assess the soil moisture estimation in future datasets. Prior to the SMAP satellite providing the crucial data required to solve this problem, Burgin et al. had attempted to tackle the same issue [35]. Here they still arrive at a similar derivation and explain the hypothesis of radar scattering and its relation to soil moisture. The authors demonstrate a proof of concept and give a range of potential coefficients that may enable future estimations once data is retrieved.

Javed et al. analyze their design of an automatic target classifier for a Ground surveillance radar [36]. The proposed learning algorithms are logistic regression and linear discriminant analysis classifiers that can accurately (>92%) identify a moving target and differentiate whether the target is a pedestrian or vehicle, it can also identify a lack of target accurately classifies noise. The proposed algorithm takes the returned 'audio signal' of the radar and preforms the fast Fourier transform where features are extracted and passed to the classifiers. Target velocity is also quickly extracted from the Doppler frequency of the target and enables the classifier to pre-process the data and determine whether the target will be a pedestrian (low Doppler) or vehicle (higher Doppler).

Nikaein and Sheikhi propose a novel least angle regression (LAR) approach for target detection and localization in passive bistatic radars [37]. Targets are initially detected using a method of sparse estimation because with several targets and clutter in a range and no priori information on them it can be difficult to detect them individually. The targets and clutter are placed into a grid of discretized Doppler frequencies bounded by the maximum expected Doppler shift of any one target. The linear regression problem is then estimated with the LASSO method for regression analysis. The authors assess the issues of prior techniques which eliminated low Doppler portions of signals but inadvertently masked and negated some targets that may be present in the radar scene [38, 39]. The compressed sensing approach combines all clutter and targets before detection which enables the detection of targets with low Doppler frequencies within background noise.

Feng proposes a novel k-means clustering algorithm capable of sorting non-cooperative radar signals with no prior information [40]. The topic of radar sorting is extremely valuable in the field of adversarial radar, where there must be a quick and accurate way to acknowledge that an unknown signal is within the shared airspace, and the ability to do this without prior knowledge is crucial. The proposed algorithm preprocesses all received radar data with standard normalization and transformation, followed by noise removal from the data. Finally, data is sorted, and a standard k-means clustering algorithm [41] is applied to the data set once cluster centers are determined. The authors demonstrate the algorithms strong ability (>97% sorting accuracy) to properly identify and classify 4 separate radar signals, each with varying frequencies, PRI's, Directions of arrival, and noise levels.

Martone et al. investigate an updated algorithm for a previous study they had preformed

on k-Means algorithms for Moving Target Indication based radar systems [42]. Previously, Martone et al. had developed an MTI approach to detect and track targets inside of buildings with these clustering algorithms but this required human interaction to select parameters for the clustering algorithms [43]. This new approach [42] automated the process by applying a knee-point and recursive pixel finding algorithms [44]. The initial MTI algorithm proposed the target detection method by comparing the differences between two SAR images and recognizing artifacts that stand out behind clutter and noise; these are then passed to the clustering algorithm and tracked accordingly.

In [45], Martone et al. demonstrate the capabilities of cognitive nonlinear radars to adapt to cluttered radio frequency environments by selecting waveform parameters rather than only switching frequency bands. The waveforms managed by the CNR are optimized by modifying parameters based on EM interference, target likelihood, and permissible transmit frequencies. The radar receive pulses are processed with a priori target information that enables the CNR to adaptively change the waveform based on live target information. This is a reasonable assumption for cognitive radars because typically information will be known about a set of targets intended on being tracked before attempting to track them. Similarly, in [46], Stinco et al. developed a CR with the ability to sense the local spectrum of radio frequencies and adapt channel parameters to minimize interference between radar and communications interferers.

### 3.2.1 Online Learning and Thompson Sampling

The main machine learning approach investigated in this paper is a online contextual Thompson Sampling algorithm that aims to optimize a dynamic spectrum access problem between

a radar system and wireless communications interference [47]. An online learning approach to machine learning is a method employed with data being fed sequentially to the model in order to update it at each step. This has advantages over batch learning styles of ML (such as image classification) when a pattern needs to be recognized or adaptively adjusted for. Online learning also benefits when the data is simply generated over time rather than all at once, in the case of this paper a radar system is assumed to act over time to properly model a target moving through space.

In [48], Liu J. et al. demonstrate the exceptional performance of an online clustering algorithm for radar emitter classification. The goal of classifier was to determine the number of emitters in a given area and to classify the pulses according to the emitters from which they originate. This is enhanced by online learning because in a real world surveillance application, radar data is being collected live and having to wait for a batch of data to be collected before it can be processed may be slow and detrimental to the overall task of the CR. The authors develop their model based clustering algorithm using a Minimum Description Length (MDL) criterion where, given a data set and family of competing statistical models, the best model is the one that yields the minimum code length for the data. In other words, the overall model, given some set of data is attempting to create a balance between fit and complexity, this enables a very strong basis for online learning where complexity should be kept low to allow for real time analysis.

Thompson sampling (also known as posterior sampling) algorithm for online decision making "must balance between exploiting what is known to maximize immediate performance and investing to accumulate new information that may improve future performance" [49]. Prior to sampling any information about what needs to be learned, the TS algorithm is provided with a set of $K$ actions, or arms, that it can choose from each with a probability of $\theta_k$ of producing a reward greater than 0. The agent is unaware of the probabilities of receiving the

reward and must explore the arms to estimate it. While simply playing each arm once and recording what occurs may work, it is a very naive approach and the TS algorithm estimates these by establishing an independent prior over each $\theta_k$. First, an assumption is made that the likelihood of the reward given $\theta_k$ is defined by some known distribution and therefore a conjugate prior can be placed on $\theta$. The algorithm in this thesis uses a multivariate normal prior distribution, thus a multivariate normal posterior distribution is the ultimate result of the distributions $\theta$. This allows for an estimation on the posterior mean matrix ($\hat{\theta}_t = \mathbf{0}_d$) and covariance matrix ($\mathbf{B} = \mathbf{I}_d$) at each time step when an arm is chosen and reward is evaluated. At each time step the posterior distribution is then updated with the new context and arms are chosen that correlate to the highest expected reward [50]. This structure enables future information to be learned in the chance that the probabilities of $\theta_k$ were to change, creating a more robust algorithm that is adaptive in time [51].

**Algorithm 1** Cognitive Radar Thompson Sampling [47]

**procedure**
- $\mathbf{B} = \mathbf{I}_d$ ▷ Covariance Matrix of Posterior Distribution
- $\hat{\theta}_t = \mathbf{0}_d$ ▷ Mean of Posterior Distribution
- $\mathbf{f} = \mathbf{0}_d$
- **for** t = 2, ..., T **do**
  - Sense interferer, $Act_{Intf}(t)$
  - $\mathbf{x}_i(t) = \{\beta_1, ..., \beta_d\}$ ▷ Discounted context
  - Sample $\tilde{\theta}_t \sim N(\hat{\theta}_t, v^2\mathbf{B}^{-1})$
  - $A'_{radar}(t) = \{\mathbf{a} \in S_{radar} : \mathbb{E}[r_i(t)|\mathbf{x}_t = x] > \hat{r}\}$ ▷ Constrained action space
  - $Act_{radar}(t) = arg\max_i \mathbf{x}_i(t)^T\tilde{\theta}_t$ ▷ Select action with the highest expected reward
  - Observe actual reward
  - Update History
  - $\mathbf{B} = \mathbf{B} + \mathbf{x_{a(t)}}\mathbf{x_{a(t)}^T}$
  - $\mathbf{f} = \mathbf{f} + \mathbf{x_{a(t)}}(t)r_{\mathbf{a(t)}}(t)$
  - $\hat{\mu} = \mathbf{B}^{-1}f$
- **end for**

**end procedure**

# Chapter 4

# System Model and Radar Environment

## 4.1 Radar System Model Assumptions

In a real world radar system there are many parameters that must be considered such as specific target dimensions, background noise, clutter, ambient temperature, weather, etc. Instead of attempting to model every possible combination of radar specific parameters, assumptions are made that simplify the problem into just a few scenarios. When considering what target is being detected, Radar Cross Section (RCS) is an important factor that shows how well the target can be detected by radar. Targets with large RCS values (planes, tanks, ships) can be much more easily detected than ones with low RCS values (drones, civilians, birds). These target RCS values also vary depending on the radar signal's angle of arrival (AoA) and a complex model must be made to simulate each potential target. For this paper all targets are assumed to be point targets which scatter equally in all direction and whose RCS value is between 2 $m^2$ and 5 $m^2$. This assumption allows for a reasonably sized target simulation of a small to medium sized plane which covers a variety of target sizes that may be tracked by standard radar systems.

Typically an antenna will sweep across some area and use information gathered to decide whether a target is found within that range. Once a target is confirmed the radar may follow

this target and track it until it is out of the antenna's range. In this paper it is assumed that there is no time needed to find the target and the simulations only accounts for received signals when the radar antenna is pointed at the target. This assumption is made simply to reduce simulation time and to ensure that estimations on the target are only made when the target is actually in the line of sight. Since only point targets are simulated, it would be easy for any antenna with a narrow beamwidth to lose the target very quickly. In a true radar scenario this would not occur because a real target has physical X, Y, and Z dimensions that will induce responses at a majority of the beamwidth.

Finally, the interferer used for analysis in this paper is assumed to be a stationary communications system that can occupy and overpower the radar signal with any combination of the N number of channels. In most of the cases studied here it is also assumed that the interferer is location independent unless explicitly stated (Spatial Correlation, Chapter 5.5). Noise generated from an interferer when overlapping with the radar subbands is modeled as Additive White Gaussian Noise (AWGN) and is at a level sufficient enough to heavily impede on the target detection performance (i.e. Signal without Interferer, SNR $\approx$ 20, Signal with Interferer, SNR = [-6,-1] ). Background noise and clutter are both target independent and AWGN with powers high enough to marginally impede on the detection performance.

## 4.2 Radar Environment

The radar specific parameters used in this paper such as center frequency, coherent processing interval (CPI), and PRI are as follows:

- Max Bandwidth = 100 MHz
- Carrier Frequency = 2.5 GHz
- Pulse Repetition Interval (PRI) = 0.1024 ms
- Pulse Repetition Frequency (PRF) = 9.765 kHz
- Coherent Processing Interval (CPI) = 0.48 s
- Pulses per CPI = 128
- Maximum Detectable Velocity = 292.9 m/s
- Maximum Unambiguous Range = 15.37 km
- Transmission Type = Linear Frequency Modulated Pulses (LFM)
- Number of Subbands (N) = [5,10]

The interferer subband space is defined by a set of possible unique states $S_{intf}$,

$$S_{intf} = [\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3, ..., \mathbf{a}_k] \tag{4.1}$$

where $k$ is the total number of combinations of the desired subbands ($k = 2^N$). These are then observed by the radar as a time sequence of interference states known as $Act_{intf}$,

$$Act_{intf} = [\mathbf{a}_{10}, \mathbf{a}_5, \mathbf{a}_{22}, \mathbf{a}_{16}, ...] \tag{4.2}$$

which can be any integer multiple length of the number of pulses per CPI. Each of the states $\mathbf{a}_i$ is a 1 $\times$ N vector defined as

$$\mathbf{a}_i = [\mu_1, \mu_2, \mu_3, ..., \mu_N]^\top \tag{4.3}$$

where $\mu_i$ is a binary 0 or 1 which represents whether the subband is occupied by the interferer or not (ex. [0,1,1,1,0] as a possible interference state).

Similarly, the centralized radar decision maker's subband space is defined by a set of possible states $S_{radar}$,

$$S_{radar} = \left[\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3, ..., \mathbf{b}_{\frac{N(N+1)}{2}}\right] \tag{4.4}$$

unlike the interferer subband space however, the radar subband space has fewer actions it may take because it must always select at least 1 subband and can only select contiguous subbands. Therefore, the maximum number of subband choices that the radar may take is limited to $\frac{N(N+1)}{2}$. The radar system then runs through every action in $A_{intf}$ and makes a decision based on the observed interferer state. These decisions are then stored in $Act_{radar}$,

$$Act_{radar} = [\mathbf{b}_{11}, \mathbf{b}_9, \mathbf{b}_{17}, \mathbf{b}_4, ...] \tag{4.5}$$

which is defined as a vector of size length($Act_{intf}$). Similar to $a_i$ seen in equation 4.4, $\mathbf{b}_i$ is a 1 $\times$ N vector defined as:

$$\mathbf{b}_i = [\mu_1, \mu_2, \mu_3, ..., \mu_N]^\top \tag{4.6}$$

which represents whether a specific subband is occupied or not. For example, $\mathbf{b}_i$ = [0, 1, 1,

1, 0] means there are 3 of 5 possible bands being occupied by the radar system. Although the state space $S_{radar}$ contains all uniquely possible binary arrangements, only those that contain contiguous sets of bands are considered true actions in $Act_{radar}$. Once the radar action space is generated, each $\mathbf{b}_i$ is translated into the LFM waveform to be used in the detection and tracking simulation.

## 4.3 User/Jammer Spectrum Transition Probabilities

Although it would be simple to select interferer states randomly, for the case of evaluating machine learning for radar applications, there must be a pre-defined state space that can be learned. It is important to understand that if there is something to learn, how well can a machine learning enabled radar decision maker learn it. Also, if it can perfectly learn it how much does this actually help in making a decision.

To properly generate an interferer set $S_{intf}$ that can be estimated by some learner, a probability transition matrix ($\mathcal{T}$) must first be created that maps all previous interference states to all next interference states. The matrix $\mathcal{T}$ is a k $\times$ k matrix whose rows and columns describe the probability of a specific interference state occurring (column), given that any one interference configuration has been observed (rows). This is a good metric to evaluate the ability to learn transition probabilities due to the fact that if all next observable states are equally likely to occur, then there is nothing truly of value to learn about our interferer and no action has a greater reward than any other. Figure 4.1 represents a sample transition matrix of size [32 $\times$ 32] where every row represents the integer value plus one of the possible interferer subband selection.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.2343 | 0.4447 | 0.2647 | 0.0499 | 0.0065 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0.0561 | 0.2614 | 0.4200 | 0.2119 | 0.0469 | 0.0037 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0.0064 | 0.0831 | 0.2912 | 0.3916 | 0.1882 | 0.0368 | 0.0027 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 2.9818e-04 | 0.0101 | 0.0948 | 0.3063 | 0.3854 | 0.1694 | 0.0317 | 0.0020 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 8.0841e-05 | 5.6589e-04 | 0.0105 | 0.0968 | 0.3222 | 0.3618 | 0.1734 | 0.0323 | 0.0022 | 0 | 0 | 0 | 0 | 0 |
| 6 | 0 | 0 | 8.1976e-04 | 0.0126 | 0.1083 | 0.3088 | 0.3651 | 0.1664 | 0.0363 | 0.0017 | 0 | 0 | 0 | 0 |
| 7 | 0 | 5.9168e-05 | 0 | 9.4669e-04 | 0.0130 | 0.1019 | 0.3255 | 0.3654 | 0.1616 | 0.0302 | 9.7627e-04 | 3.5501e-04 | 0 | 0 |
| 8 | 0 | 0 | 0 | 0 | 0.0012 | 0.0163 | 0.1105 | 0.3123 | 0.3588 | 0.1691 | 0.0284 | 0.0034 | 0 | 0 |
| 9 | 0 | 0 | 0 | 0 | 0 | 7.0190e-04 | 0.0151 | 0.1183 | 0.3262 | 0.3495 | 0.1567 | 0.0326 | 9.1247e-04 | 0 |
| 10 | 0 | 0 | 0 | 0 | 0 | 1.2704e-04 | 5.9286e-04 | 0.0132 | 0.1199 | 0.3316 | 0.3547 | 0.1514 | 0.0267 | 0.0018 |
| 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.0011 | 0.0181 | 0.1127 | 0.3187 | 0.3629 | 0.1646 | 0.0203 |
| 12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3.1976e-04 | 0.0162 | 0.1165 | 0.3360 | 0.3485 | 0.1549 |
| 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.0012 | 0.0197 | 0.1232 | 0.3201 | 0.3442 |
| 14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.0013 | 0.0210 | 0.1219 | 0.3195 |
| 15 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.0012 | 0.0209 | 0.1252 |
| 16 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.0012 | 0.0159 |
| 17 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1.6505e-04 | 0.0011 |

Figure 4.1: A sample Transition Matrix, $\mathcal{T}$, of size $[32 \times 32]$, representing an interferer with 5 subbands and thus $2^5 = 32$ states.

### 4.3.1 Transition Matrix R-Values

One metric used to assess the transition matrices are their diagonality, referred to as their R-value. Since a transition matrix relates all observed states to all possible next states, the diagonal values of this matrix represents the probabilities that no change will occur between the observed time $t$ and next state at time $t + 1$. The more likely an observed interferer is to not change within the next time step, the larger the probability found at $\mathcal{T}(b_t, b_t)$ will be. R-values are a single number between $[0, 1]$ that is computed once a $\mathcal{T}$ is generated. An R-value correlated with the diagonality of a matrix tells how strongly deterministic and consistent our interferer is over time, and an R-value of 1 would represent an interferer that is perfectly static and never changes its subband selection. Alternatively, an R-value of 0 represents an interferer that is highly dynamic and where the next state is guaranteed to be different from the current observed state. Since all $\mathcal{T}$'s are square matrices of size k $\times$ k then their R-values are calculated as such,

$$j = \mathbf{1}^{(1\times k)}$$

$$r = (1, 2, ..., k)$$

$$r_2 = (1^2, 2^2, ..., k^2)$$

$$n = j\mathcal{T}j^T$$

$$\sum x = r\mathcal{T}j^T$$

$$\sum y = j\mathcal{T}r^T$$

$$\sum x^2 = r_2\mathcal{T}j^T$$

$$\sum y^2 = j\mathcal{T}r_2^T$$

$$\sum xy = r\mathcal{T}r^T$$

$$R = \frac{n\sum xy - \sum x \sum y}{\sqrt{n\sum x^2 - (\sum x)^2}\sqrt{n\sum y^2 - (\sum y)^2}} \tag{4.7}$$

This can then be applied to a square transition matrix of any size to roughly evaluate how deterministic it may be. Figure 4.1 is visibly deterministic, and a human could generally predict what next state the interferer will be in with high accuracy. As expected, this transition matrices R-value is very high (0.9846). Matrices with low R-values cannot be accurately predicted by simple selection and do not provide much information to the decision maker. It should be noted that a transition matrix can be deterministic but have a low R-value, an example being a triangle frequency sweep where each next observed state is predictable but does not lie along the diagonal. To evaluate the case where a matrix is deterministic but not diagonal entropy is used to measure the *randomness* of the matrix.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.0153 | 0.0297 | 0.0426 | 0.0556 | 0.0574 | 0.0644 | 0.0761 | 0.0741 | 0.0769 | 0.0690 | 0.0639 | 0.0549 | 0.0546 | 0.0531 |
| 2 | 0.0148 | 0.0277 | 0.0388 | 0.0504 | 0.0639 | 0.0618 | 0.0661 | 0.0628 | 0.0841 | 0.0669 | 0.0652 | 0.0541 | 0.0534 | 0.0496 |
| 3 | 0.0125 | 0.0258 | 0.0347 | 0.0464 | 0.0557 | 0.0630 | 0.0646 | 0.0693 | 0.0671 | 0.0657 | 0.0687 | 0.0632 | 0.0602 | 0.0449 |
| 4 | 0.0116 | 0.0217 | 0.0321 | 0.0414 | 0.0527 | 0.0558 | 0.0647 | 0.0652 | 0.0696 | 0.0715 | 0.0638 | 0.0634 | 0.0526 | 0.0613 |
| 5 | 0.0105 | 0.0184 | 0.0313 | 0.0380 | 0.0466 | 0.0517 | 0.0533 | 0.0552 | 0.0670 | 0.0678 | 0.0680 | 0.0670 | 0.0499 | 0.0465 |
| 6 | 0.0084 | 0.0187 | 0.0253 | 0.0308 | 0.0408 | 0.0422 | 0.0490 | 0.0612 | 0.0645 | 0.0738 | 0.0582 | 0.0661 | 0.0521 | 0.0490 |
| 7 | 0.0075 | 0.0154 | 0.0201 | 0.0269 | 0.0363 | 0.0420 | 0.0435 | 0.0498 | 0.0538 | 0.0504 | 0.0587 | 0.0599 | 0.0495 | 0.0633 |
| 8 | 0.0063 | 0.0113 | 0.0169 | 0.0235 | 0.0316 | 0.0405 | 0.0472 | 0.0443 | 0.0472 | 0.0575 | 0.0584 | 0.0609 | 0.0603 | 0.0591 |
| 9 | 0.0052 | 0.0109 | 0.0141 | 0.0205 | 0.0266 | 0.0294 | 0.0352 | 0.0446 | 0.0486 | 0.0503 | 0.0566 | 0.0604 | 0.0632 | 0.0555 |
| 10 | 0.0040 | 0.0085 | 0.0127 | 0.0150 | 0.0207 | 0.0269 | 0.0311 | 0.0363 | 0.0417 | 0.0444 | 0.0508 | 0.0552 | 0.0484 | 0.0566 |
| 11 | 0.0031 | 0.0063 | 0.0103 | 0.0136 | 0.0171 | 0.0235 | 0.0257 | 0.0317 | 0.0327 | 0.0410 | 0.0451 | 0.0469 | 0.0525 | 0.0589 |
| 12 | 0.0028 | 0.0048 | 0.0082 | 0.0106 | 0.0136 | 0.0192 | 0.0234 | 0.0273 | 0.0328 | 0.0334 | 0.0377 | 0.0506 | 0.0576 | 0.0569 |
| 13 | 0.0023 | 0.0037 | 0.0065 | 0.0104 | 0.0118 | 0.0165 | 0.0198 | 0.0196 | 0.0296 | 0.0349 | 0.0388 | 0.0455 | 0.0488 | 0.0556 |
| 14 | 0.0018 | 0.0027 | 0.0050 | 0.0072 | 0.0112 | 0.0129 | 0.0153 | 0.0221 | 0.0250 | 0.0316 | 0.0308 | 0.0390 | 0.0477 | 0.0514 |
| 15 | 0.0012 | 0.0024 | 0.0039 | 0.0051 | 0.0081 | 0.0111 | 0.0138 | 0.0167 | 0.0219 | 0.0266 | 0.0312 | 0.0361 | 0.0389 | 0.0446 |
| 16 | 9.9688e-04 | 0.0023 | 0.0032 | 0.0042 | 0.0079 | 0.0099 | 0.0142 | 0.0158 | 0.0178 | 0.0253 | 0.0280 | 0.0312 | 0.0303 | 0.0407 |
| 17 | 0.0012 | 0.0015 | 0.0032 | 0.0035 | 0.0057 | 0.0069 | 0.0104 | 0.0122 | 0.0159 | 0.0203 | 0.0205 | 0.0291 | 0.0324 | 0.0366 |

Figure 4.2: A sample transition matrix with low R-value (R = 0.5790)

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| 2 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| 3 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| 4 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| 5 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| 6 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| 7 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| 8 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| 9 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| 10 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| 11 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| 12 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| 13 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| 14 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| 15 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| 16 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |
| 17 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 | 0.0313 |

Figure 4.3: A completely uniform transition matrix with an R-value of exactly 0. This example transition matrix would provide nothing to learn to a centralized learner and would cause serious challenges to Sense and Avoid.

### 4.3.2 Transition Matrix Entropy

As described in 4.3.1 a state transition matrix may be deterministic but not diagonal. In these cases the R-value does not provide much insight into how the interferer is moving between states because it only tells us how likely it is that the next action taken by the interferer is the same as the previous one. For this task Shannon entropy [52] is calculated across every row of possible observed states and then averaged to get a total entropy of the transition matrix $\mathcal{T}$. The entropy (H) of a transition matrix, $\mathcal{T}$ is defined by

$$\mathcal{H}(\mathcal{T}) = \frac{-\sum_{i=1}^{2^N} \mathcal{T}(1,i)\log(\mathcal{T}(1,i)) - \sum_{i=1}^{2^N} \mathcal{T}(2,i)\log(\mathcal{T}(2,i)) - ... - \sum_{i=1}^{2^N} \mathcal{T}(2^N,i)\log(\mathcal{T}(2^N,i))}{2^N} \tag{4.8}$$

where $\log(\mathcal{T}(x,y))$ is evaluated in base $e$.

## 4.4 Experiment Details

The core of the problem that is being tackled in this paper is the ability to detect and track a target using radar; however, the main factor in a radar's ability to perform these tasks is bandwidth. A radar system's range resolution is limited only by its bandwidth (B), defined as

$$\Delta R = \frac{c}{2B}, \tag{4.9}$$

where a larger bandwidth reduces the range resolution and increases the accuracy of our target tracking. In an ideal scenario with no interferer or system limitations it would be favorable to select as large of a bandwidth as possible to detect the target perfectly. However, in this paper and in real world applications this is just not a feasible approach. The tradeoff typically made is that at further distances a coarser range resolution is acceptable since the difference of a few meters at distances of several kilometers is almost negligible. However, as the target approaches the radar system, this range resolution must become much finer to truly be certain of the target's location.

It should become apparent now that the purpose of these learning and classical algorithms

are to optimize bandwidth allocation when attempting to coexist with an interferer using the same bands as the radar. Although these algorithms all perform their tasks with verifiable results there will always be mistakes and no one algorithm is perfect. In the instance where a subband configuration is selected that collides with an interferer there will be a significant drop in the SINR (Signal to Interference plus Noise Ratio). Without any interferer and coherent processing a typical SINR level will be between 15dB and 20dB, as seen in Table 4.1, signifying that the radar signal is strongly identifiable over the background and system noise. For the simulations in this thesis a signal-to-interference ratio (SIR) of roughly 20 is used to properly hide the target, as seen in Table 4.1. Once an interferer collides with the radar subband selection SINR may drop to 2dB and even below 0dB, see Table 4.1. At these levels, the radar will easily lose the target and may not be able to identify the radar signal over any of the interference. This is highly undesirable for a tracking radar that may need constant surveillance over a high priority target.

Table 4.1: List of sample SINR values with and without interferences.

| **SINR** with **no** interference | **SINR with** interference |
|---|---|
| 15.1853 dB | -8.5135 dB |
| 18.9964 dB | 0.26594 dB |
| 19.3653 dB | -3.5156 dB |
| 14.9739 dB | -2.9228 dB |
| 14.4886 dB | -5.5811 dB |
| 18.559 dB | -0.13321 dB |

A simple solution for this may just be to avoid the interferer subbands entirely and to simply transmit in unused frequencies. However, with the growing usage of spectrum by industrial communications systems, this just may not be practical. The point of the algorithms studied in this thesis is to solve an optimization problem where the radar seeks to avoid interference while maximizing bandwidth to ensure proper tracking of a target. Using too few subbands

may lead to large tracking errors and using too many subbands may lead to too many collisions and an insurmountable drop in SINR.

The final stages of the radar environment is the target information that can be inferred by the radar system. Two key elements that the user needs to know about a target are its range and velocity.

In a coherent radar system information about transmitted pulses must be retained such as, frequency of the pulse, phase of the pulse when transmitted, and timing of when the pulses were transmitted. With timing alone a target's range may be calculated simply by using Equation 2.7. A more difficult issue is the velocity of the target which requires extremely accurate phase information about the transmitted and received signal. If a radar system can track this perfectly then Equation 2.8 describes how the target's velocity is computed. In this paper it is assumed that the system has perfect knowledge of the pulse travel time and phase changes brought on by the target, this allows an accurate simulation of a target that is traveling in any direction relative to the radar antenna.

Figures 4.4 and 4.5 demonstrate a range Doppler map that is used in processing the range of a sample target. Although it may be simple to detect a target by glancing at the range Doppler map only a small section is provided to the radar system for analysis. This cutout section, known as the "validation region", is an an area where the target is expected to be over time and allows for simple computation of range errors and limits maximum estimation error by restricting the search area. The agent then selects the cell with the highest value (brightest white in the figures) and returns the range and velocity it believes the target is at. This is then compared to the true range of the target and recorded for each channel selection agent. As noted in equation 4.9, with a higher bandwidth range resolution decreases allowing

for a narrower target estimation. Figure 4.4 shows a target that is tracked at 350 meters with full bandwidth utilization, providing a narrow range of possible target locations. However, figure 4.5 shows the same target tracked with lower bandwidth utilization, increasing the range resolution and giving less accurate results about where the target lies. Figure 4.6 shows an example of when an interference occurs at the receiver. In this range Doppler map the target is completely hidden within noise and no accurate reading can be evaluated.

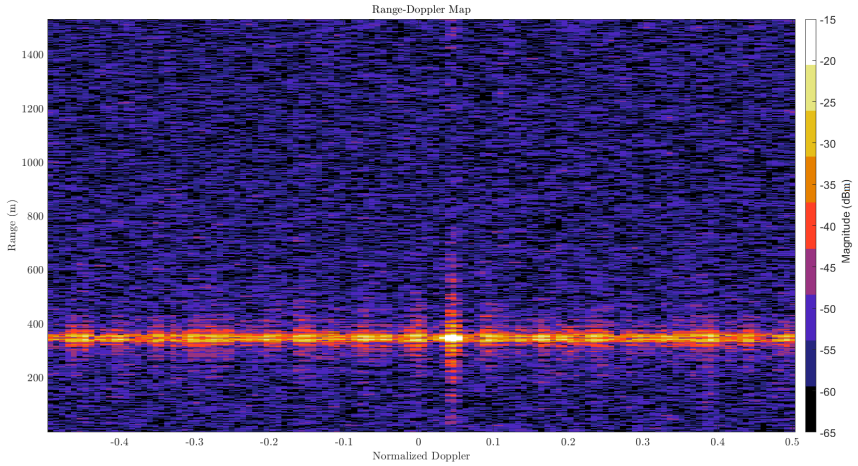


Figure 4.4: A target is seen in this example Range Doppler Map with a narrow range resolution we can be certain that our target is within just a few meters of the predicted range.

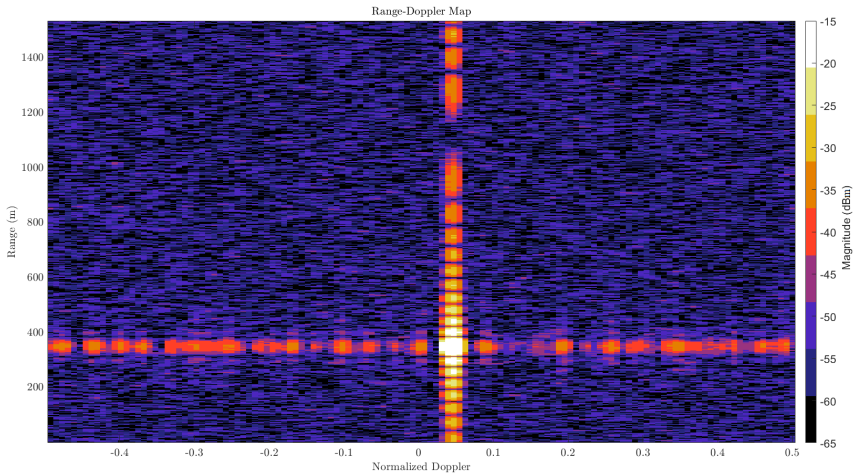


Figure 4.5: In this figure a lower bandwidth is selected and the range resolution is increased leading to larger uncertainty about exactly where the target may be. Noise is also affecting the computations a little more than in figure 4.4

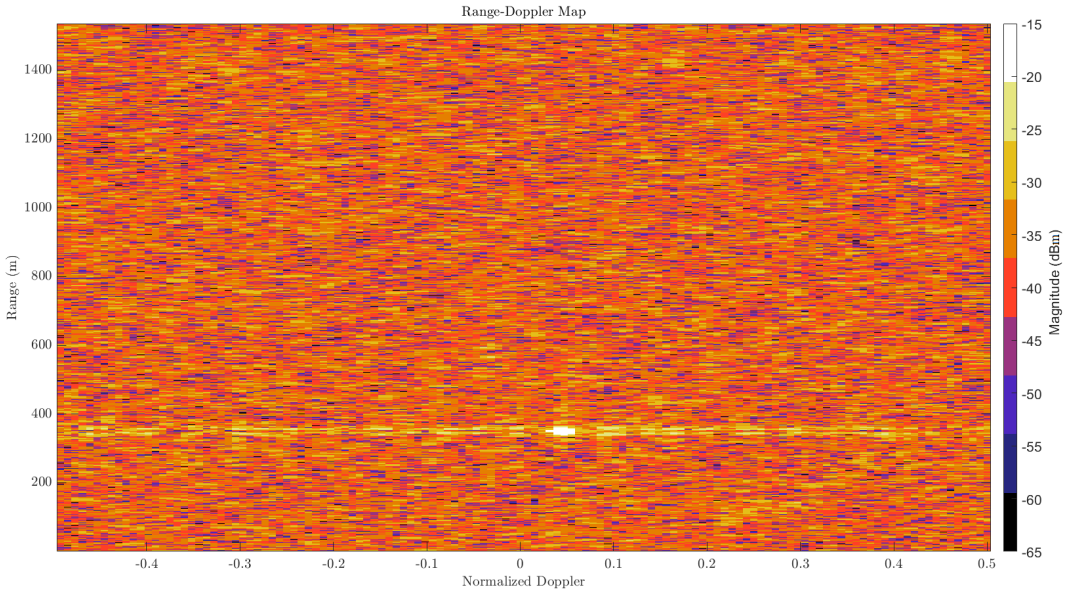


Figure 4.6: In this figure an interferer has collided with our radar subband selection and Interference has masked the target within the interference noise. In this Range Doppler Map the target is well hidden; although still visible for demonstration, in reality no information can be discerned about the target when an interference occur.

# Chapter 5

# Experimental Results and Analysis

## 5.1 Intuition

Before presenting results of the experiments conducted in this research it is important to provide some intuition about the various experiments. This helps provide a sanity check for the results seen in simpler experiments and provides confidence in the more complicated scenarios. Since the essence of this paper lies in the value of machine learning on a set of tasks, a good first approach would be to consider scenarios where classical sense and avoid preforms optimally (or at least very well) when compared to machine learning.

1. The first trivial scenario would be when an interferer is highly deterministic (R-value $> 0.97$), with a low transition matrix entropy, and does not change states for long periods of time. In this case any observed state will likely give a lot of information into the next expected state, this is the essence of sense and avoid which simply assumes that the next state will be identical to the current state. Although sense and avoid would be near optimal in this case it is assumed that a machine learning model should be able to adapt and replicate sense and avoid, however, regret will be incurred during the learning phase and so machine learning is not needed. As a transition matrix for an interferer becomes less deterministic and diagonal there will be greater benefits to

using a learning model.

2. Alternatively, a scenario where a communication interferer is exceptionally not deterministic (R-value 0.00) with a near maximum transition matrix entropy it should be clear that there is nothing to learn. In these scenarios any observed state gives absolutely no information into the next expected state, as they are all equally likely to occur. Here, neither machine learning nor sense and avoid can really be considered optimal because tracking performance will be poor regardless of action taken. Although both models will preform poorly, a machine learning approach may consider selecting fewer subbands, or even just a single subband, that it believes have provided it with the greatest reward over its learning time. Sense and avoid, however, is not adaptive and will not attempt to improve in any way. In other words, machine learning can at least learn what is best on average while sense and avoid cannot.

3. Some cases where a learning model should be expected to perform better than sense and avoid are when the interferer transition matrix can be modeled as very deterministic but not diagonal (low entropy, low R-value). Examples of these interferer types are triangle frequency sweep, sawtooth frequency sweep, binary counters, and other simple patterns. A machine learning model is expected to learn these simple patterns and optimally select the correct subbands within some reasonable learning time while a sense and avoid radar system will be incorrect at every time step.

4. In a real world radar scenario timing must be a serious consideration when deciding on a learning algorithm. In the cases where a radar system can detect an interferer within some band and switch its own subband selection faster than the interference, sense and avoid can be considered optimal. Here learning is not really necessary because relative to the timesteps of the radar system, the interferer is very deterministic and regret will only be incurred when the interferer switches subbands.

5. Finally, a core component of all machine learning algorithms explored in Ch.3 is their reward functions. Each model is attempting to maximize their reward for a given problem and if the reward functions do not correlate to good radar detection and tracking performance then they cannot be considered a strong replacement for sense and avoid.

## 5.2 Channel Selection Problem

The following results show the performance of the 3 main algorithms investigated in this paper Sense and Avoid, Thompson Sampling, and the Bellman Optimal Policy. In this section they are gauged solely on their ability to avoid an interferer and properly select optimal subbands when facing an interferer. Although no actual radar emulation is preformed, this provides a good first view of a machine learning model when tasked with the channel selection problem. This should hopefully directly relate to radar performance and provide some insight into the value of machine learning. In each of these figures a comparison between online Thompson Sampling, Sense and Avoid, and "Optimal" is recorded for a wide range of calculated R and H values. As described in Ch. 3, when an "optimal" channel selection agent is referenced in this paper it is assumed to be the Bellman optimal policy. This optimal agent does not know what state will occur next but it instead it is given the exact probability that a state will occur given that a state was observed, then applies weights to all possible outcomes relative to the maximum reward for a given number of subbands. This optimal agent then simply selects the action that returns the highest reward, which over the long run should provide the best possible performance (assuming the reward is structured properly).

The reward function used to evaluate the optimal reward is described by,

$$r(t)_i = \begin{cases} 0, & \text{for } N_c > 0 \\ \eta_1/(\eta_2 N_{mo}), & \text{for } N_c = 0, N_{mo} > 0 \\ 1, & \text{for } N_c = 0, N_{mo} = 0 \end{cases}, \tag{5.1}$$

where $\eta_1$ and $\eta_2$ are tunable parameters that enable the user to balance how the machine learning model attempts to tackle the channel selection problem. In this structure $N_{mo}$ represents the number missed opportunities incurred when a subband selection is made. An example case for an interferer who has selected the subband configuration $b_i = [1, 0, 0, 0, 1]$, the optimal action here is of course to select subbands two, three, and four ([0,1,1,1,0]). However, if any of our channel selection agents choose fewer than what is considered optimal, while still not selecting an interfering band, this is counted as a missed opportunity. It should be apparent then that the machine learning's ability to decide between higher bandwidth utilization or lower interferences is heavily influenced by $\eta_1/\eta_2$. If $\eta_1$ were set to 0 then this learning model will only incur reward when the absolutely optimal decision is made, this of course would be negatively impact the radar system as it will rarely ever be perfectly certain of what subbands the interferer will choose next. When $\eta_1/\eta_2$ is large, the model will be much looser when attempting to take up as much bandwidth as possible, because the penalty for missed opportunities is decreased. Alternatively for a small $\eta_1/\eta_2$, the model will attempt to fill as much bandwidth as possible while still avoiding interference.

As described earlier, a Bellman optimal policy is computed by taking the weights of optimal decisions and evaluating it given the known state transition probabilities, see Algorithm 1.

First, a set of solutions to all possible interferer actions is computed where the largest set of contiguous subbands is selected; each optimal action provides maximum reward if selected given that associated interferer state is present during that action.

| Interferer State | Optimal Action |
|---|---|
| [0,0,0,0,0] | [1,1,1,1,1] |
| [0,0,0,0,1] | [1,1,1,1,0] |
| [0,0,0,1,0] | [1,1,1,0,0] |
| [0,0,0,1,1] | [1,1,1,0,0] |
| [0,0,1,0,0] | [1,1,0,(0,0)] |
| [0,0,1,0,1] | [1,1,0,0,0] |
| [0,0,1,1,0] | [1,1,0,0,0] |
| [0,0,1,1,1] | [1,1,0,0,0] |
| [0,1,0,0,0] | [0,0,1,1,1] |
| [0,1,0,0,1] | [0,0,1,1,0] |
| [0,1,0,1,0] | [(0),0,1,0,(0)] |
| [0,1,0,1,1] | [(0),0,1,0,0] |
| [0,1,1,0,0] | [0,0,0,1,1] |
| [0,1,1,0,1] | [(0),0,0,1,0] |
| [0,1,1,1,0] | [(0),0,0,0,1] |
| [0,1,1,1,1] | [1,0,0,0,0] |
| [1,0,0,0,0] | [0,1,1,1,1] |
| [1,0,0,0,1] | [0,1,1,1,0] |
| [1,0,0,1,0] | [0,1,1,0,0] |
| [1,0,0,1,1] | [0,1,1,0,0] |
| [1,0,1,0,0] | [0,0,0,1,1] |
| [1,0,1,0,1] | [0,(0),0,1,0] |
| [1,0,1,1,0] | [0,(0),0,0,1] |
| [1,0,1,1,1] | [0,1,0,0,0] |
| [1,1,0,0,0] | [0,0,1,1,1] |
| [1,1,0,0,1] | [0,0,1,1,0] |
| [1,1,0,1,0] | [0,0,1,0,(0)] |
| [1,1,0,1,1] | [0,0,1,0,0] |
| [1,1,1,0,0] | [0,0,0,1,1] |
| [1,1,1,0,1] | [0,0,0,1,0] |
| [1,1,1,1,0] | [0,0,0,0,1] |
| [1,1,1,1,1] | [0,0,0,0,0] |

Table 5.1: Sample list of optimal decisions used in the optimal decision maker for a 5 subband case. Subbands in () denote that either can be selected to provide an optimal solution.

Table 5.1 shows the optimal decisions that are precomputed and stored in a $2^k \times k$ array to be recalled later.

**Algorithm 2** Generate Bellman Optimal Matrix

```
1:  procedure
2:      Intf(k) = [2^k × k]                          ▷ Array of all interference states
3:      Opt(k) = [2^k × k]                  ▷ Array of all optimal decisions for Intf(k)
4:      Tmat = [2^k × 2^k]                          ▷ Matrix of transition probabilities
5:      i,m,j,n = 1
6:
7:      for i ≤ k do
8:          intf = Intf(i,:)
9:          for m ≤ k do
10:             act = Opt(m,:)
11:             Nc = sum(and(intf,act))                  ▷ Calculate Number of Collisions
12:             Nmo = sum(not(or(intf,act)))  ▷ Calculate Number of Missed Opportunities
13:
14:             rewards(i,m) = CalculateReward(Nc, Nmo)                ▷ Equation 5.1
15:
16:         end for
17:     end for
18:
19:     j,m,n = 1
20:     for j ≤ k do
21:         for n ≤ k do
22:             for n ≤ k do
23:                 weightedReward(j,m,n) = Tmat(j,n) * rewards(n,m)
24:             end for
25:         end for
26:     end for
27:
28:     bellOpt = sum(weightedReward,3)
29:
30: end procedure
```

Once generated, the optimal subband selection is made by choosing the state which returns the highest reward. Finally, this is generated over all R and H values being investigated and compiled into values that calculate average reward, average collisions, and average number of subbands selected.

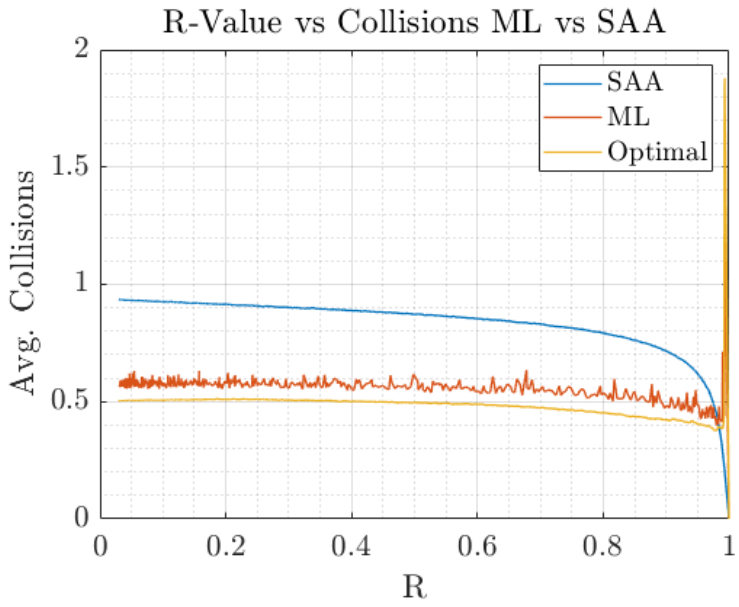


Figure 5.1: SAA, TS, and Optimal Sub-band selection algorithms are tested against a range of R-values for average collisions

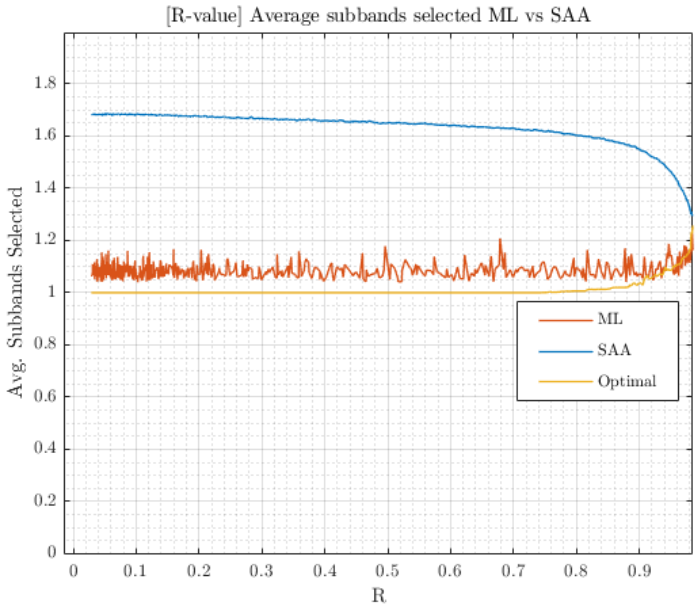


Figure 5.2: SAA, TS, and Optimal Sub-band selection algorithms are tested against a range of R-values for average number of Sub-bands selected

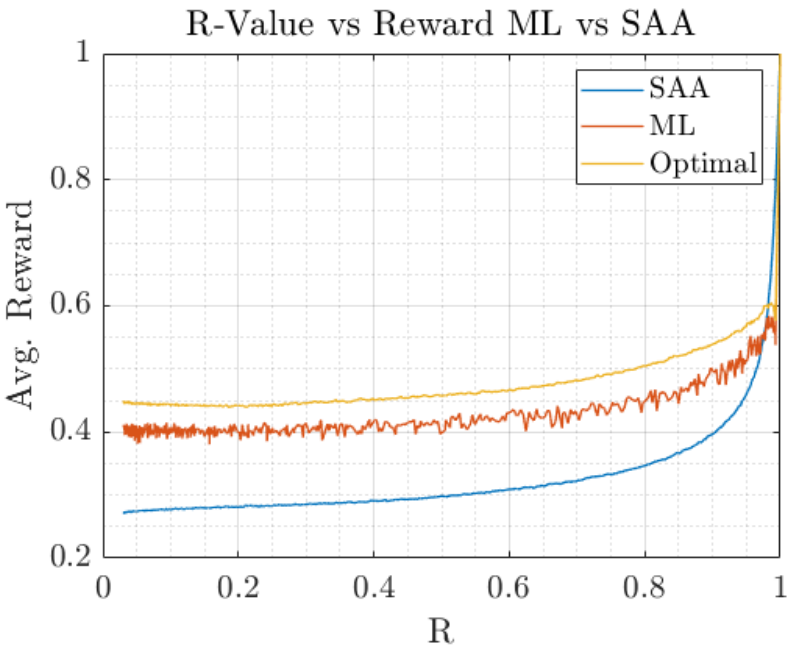


Figure 5.3: Testing R-values versus Average Reward. At the far right (>.95) optimal and SAA act roughly the same.

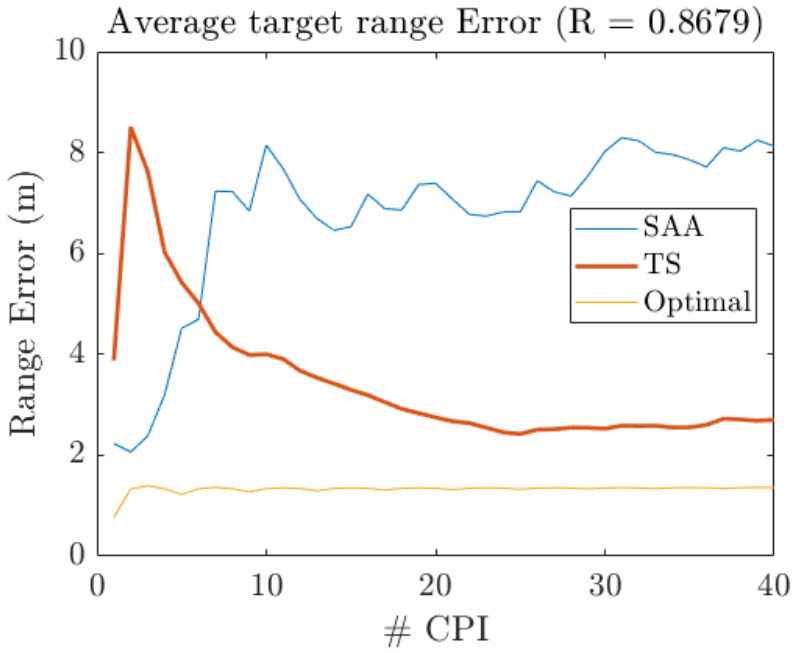


Figure 5.4: Average tracking error for a sample transition matrix with R = 0.8679, roughly where the average reward between optimal and ML differ the greatest.

Figure 5.1 demonstrates the average number of collisions incurred by each of the three sub-band selection algorithms as the R-value varies. Figure 5.4 shows that the Thompson sampling algorithm investigated in this paper has good tracking error over the sense and avoid agent and approaches Bellman Optimal when run sufficiently long. However, Figure 5.2 shows that the actions taken by the Optimal agent actually selects fewer subbands on average. In the realm of interference avoidance this is extremely favorable because a tighter range may be needed to absolutely avoid interference. Figure 5.3 demonstrates that both the optimal and ML agent average similar long term reward while sense and avoid falls short since it is unable to learn anything. The overall lower reward seen by the ML agent is believed to occur due to its inability to perfectly learn the state transitions of the interferer within the given time (300 CPI's). It is believed that if given an indefinite amount of time to learn it may eventually converge to optimal at every R-value. In these simulations the optimal agent is aiming at making the greatest long term actions based on a perfect set of interferer information while the machine learning model is strictly following the rules defined in the reward structure while having to learn the interferer over time. It is important to note that the improved reward values of the optimal agent over ML (Figure 4.4) does correspond to improved target tracking (i.e. lower range estimation error in Figure 5.4). Figure 5.5 shows the entropy of the transition matrix compared to average collisions during the entire simulation. As the interferer becomes more deterministic (R-values near 1 and H-values near 0) Sense and Avoid preforms better in the channel selection problem and is shown to be near optimal. Figure 5.6 demonstrates similar results for entropy of the transition matrices versus average reward. We note that in near-deterministic cases the ML agent preforms worse than both optimal and sense and avoid, this may be due to very few observed state transitions during the simulation time period. However, since the optimal agent can preform better in these cases there should be a different ML model that can learn how to act optimally in these scenarios.

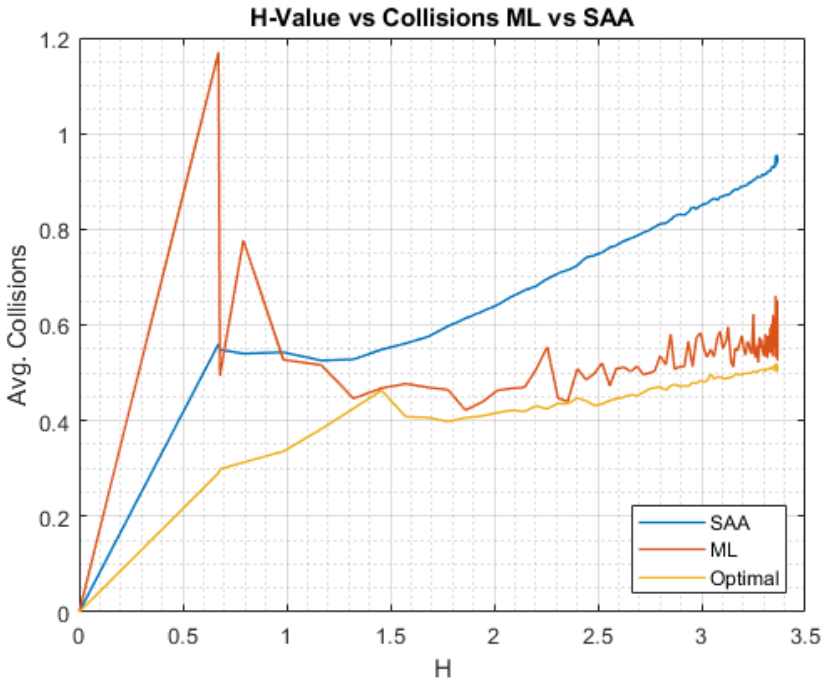


Figure 5.5: Collisions vs. Entropy

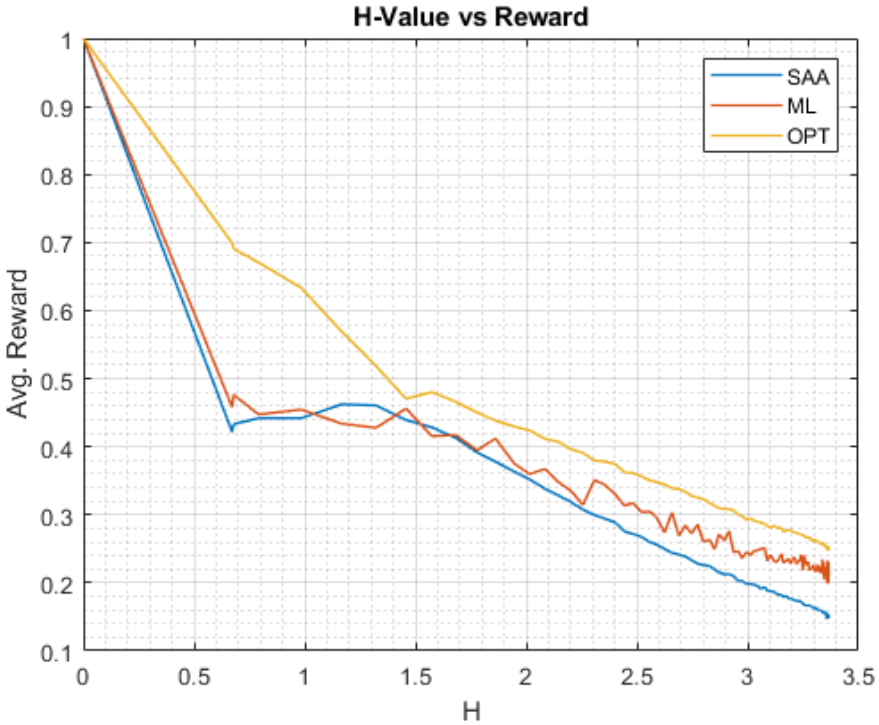


Figure 5.6: Reward vs. Entropy

## 5.3 Subbands with varying levels of activity

The case of subbands with varying levesl of activity is similar to the basic subband selection problem, except that certain subbands are always available to the agent at repeating and deterministic intervals. This was chosen to be studied as a potentially real-world scenario where one interferer in the system occupies most but not all subbands that the radar system is attempting to use. Although the interferer may not be using these subbands at all, the problem would be trivial if only a contiguous subset of bands was left open at every single state. Therefore, to actually ensure that there is something to learn, a pattern is chosen. In the two cases studied here, two and three contiguous subbands are made available to the channel selection agent seven consecutive times. These subbands are then fully occupied for the next three time steps. This allows something for the machine learning agent to actually attempt to fit itself to and it also dissaudes the optimal agent from only selecting the open bands every time.

Unlike the previous channel selection problem, R-values and entropy are not useful metrics since the observable pattern is longer than one observation interval. However, all other interference subbands that are not defined by this pattern are still generated by their R-values which are given below. Figure 5.7 demonstrates a seven subband case with three lower activity channels added in subbands one, two, and three. Each row represents a subband selection of the interferer over time and the columns are the individual bands selected by the interferer at that time step.

| | $\mu_1$ | $\mu_2$ | $\mu_3$ | $\mu_4$ | $\mu_5$ | $\mu_6$ | $\mu_7$ |
|---|---|---|---|---|---|---|---|
| t = 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| t = 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |
| t = 2 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
| ⋮ | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| | 1 | 1 | 1 | 1 | 0 | 1 | 0 |
| | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| | 1 | 1 | 1 | 1 | 0 | 1 | 1 |
| | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| | 0 | 0 | 0 | 1 | 0 | 1 | 0 |

Figure 5.7: Example of varying level of activity in predetermined interferer bands (seven subband case with the first three subbands of lower activity). A clear pattern of 3 occupied subbands and seven open subbands (green) is seen repeating here in bands one, two, and three.

Figure 5.8 demonstrates the average collisions of all three models in the first ten CPI's. Here it is obvious that the machine learning radar agent has not had sufficient time to investigate the subband states and on average makes many mistakes when compared to sense and avoid and optimal. The only exception here is at the five total subband case; it is noted that thompson sampling and sense and avoid collide an equal number of times during the first 10 CPI, however TS still selects a much lower average number of subbands as seen in Figure 5.9. Figure 5.16 now shows that TS performs slightly worse at the lowest subband case, mostly due to the increased complexity of the problem. However 5.17 still shows a lower average number of subbands selected even though more subbands are readily available. These large errors and low selections are due in part to the greedy-exploration algorithms implemented in TS where there is an attempt to learn as much information about the state space by taking many actions as quickly as possible. However, once a pattern is recognized in what actions it must take it begins to properly adjust its selections.

Figures 5.10 and 5.18 demonstrate the average collisions of the machine learning agent when given more time to adjust to the interferer. In both cases the machine learning agent selects a lower number of subbands as it did in the first 10 CPI. The machine learning model begins to recognize the pattern of lower activity subbands and attempts to exploit these openings, however, it still appears to make a high number of mistakes because collisions remain high throughout. For the lower 2 subband cases, channel selections numbers quickly approach optimal showing that with lower complexity problems convergence time is greatly diminished. However, with the large subband cases, more time is still needed to converge to a policy that provides greater results than even sense and avoid.

Figures 5.12 and 5.20 demonstrate how the machine learning model has improved when given a reasonable amount of time (80 CPI's) to learn from the interferer. At this point

the machine learning radar agent has selected proper channels that enable lower collisions than sense and avoid across all number of subbands except the largest one. It is still noted that Figures 5.13 and 5.21 show a lower number of average subbands selected by the machine learning agent. This again shows that the learning may not be making "smarter" decisions and possibly favoring safer decisions when compared to the bellman optimal agent which demonstrates the ability to avoid the interferer better than machine learning while still choosing a higher number of subbands overall.

Finally, Figures 5.14 and 5.22 demonstrate a long run of the simulation where the machine learning agent is given a significant amount of time to learn about the channel state space. At this point, the machine learning agent does not improve much unless given a vast amount of time and mostly repeats its actions indefinitely. This is due to the value of what has already been seen by the machine learning agent, as more information is fed to the model it would need significantly more to overcome its previous decisions on how to act next. In these cases the machine learning agent has successfully improved on collisions when compared to sense and avoid, successfully reducing collisions at all levels of complexity across both degrees of channel activity. However, it is still noted in both 5.15 and 5.23 that the machine learning agent is constantly selecting fewer than optimal subbands and colliding more times than optimal. While this does not significantly devalue the machine learning agent, it should show that there is still more to be desired when considering the machine learning agent, possibly better reward structures or varying greedy-exploitation algorithms. There may also be no way to refactor machine learning algorithms to make better decisions and safer decisions may just always be the outcome of channel selection problems when specifically looking at radar due to the inherent need to avoid collisions to properly track a target. The bellman optimal agent is demonstrated to track a target with higher accuracy than the

machine learning and sense and avoid agents, however, given sufficient time it is possible that the ML agent may converge much closer to optimal.

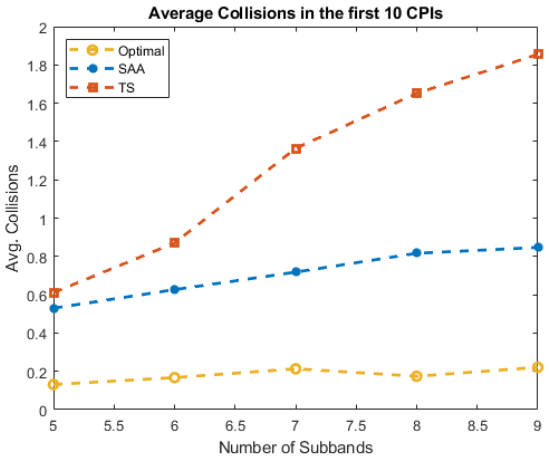


Figure 5.8: Average collisions of the first 10 CPI with varying subband lengths and 2 subbands of low activity

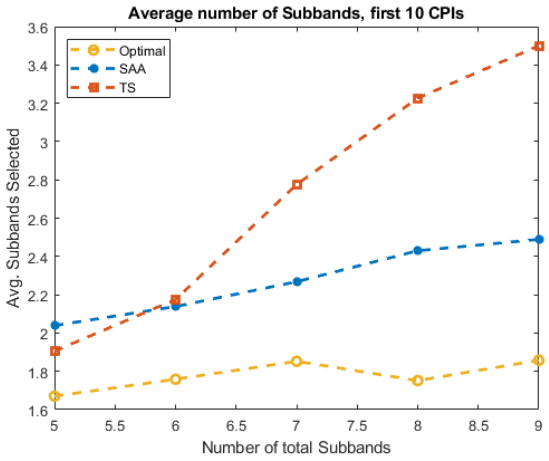


Figure 5.9: Average number of subbands selected in the first 10 CPI's with varying subband lengths and 2 subbands of low activity

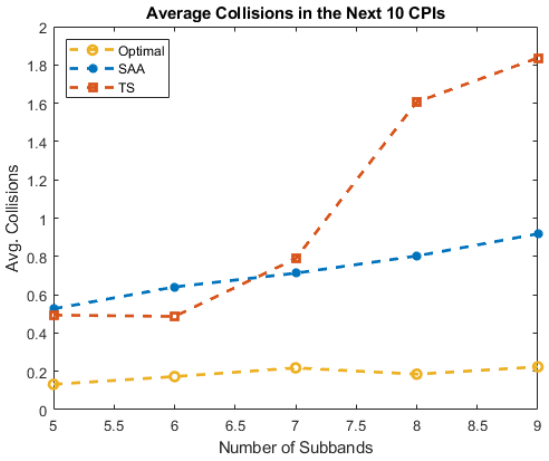


Figure 5.10: Average collisions of the next 10 CPI's with varying subband lengths and 2 subbands with low activity

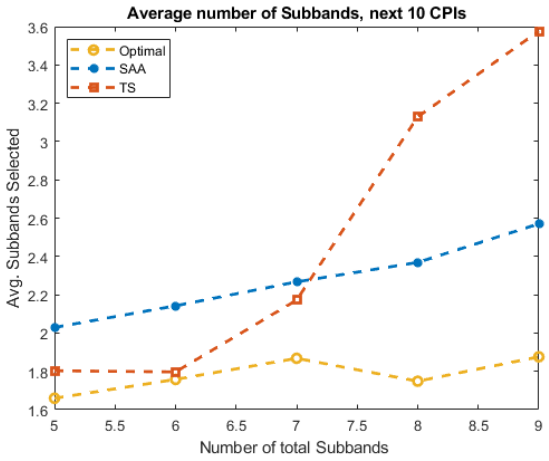


Figure 5.11: Average number of subbands selected in the next 10 CPI's with varying subband lengths and 2 subbands with low activity

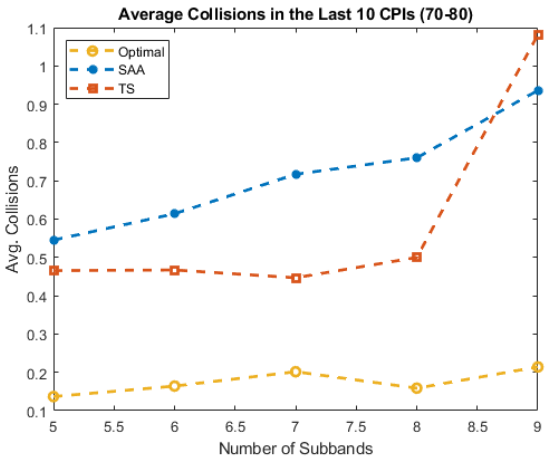


Figure 5.12: Average collisions of the last 10 CPI's with varying subband lengths and 2 subbands with low activity

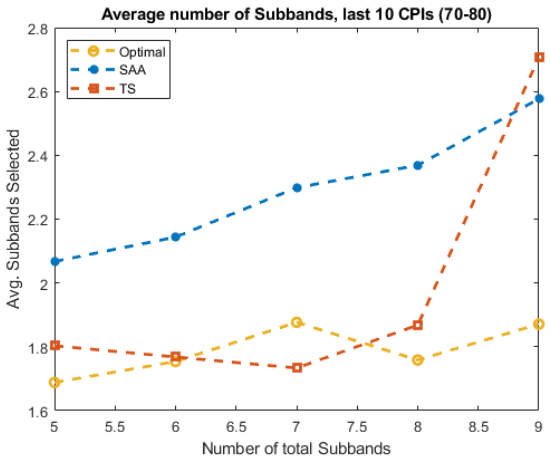


Figure 5.13: Average number of subbands selected in the last 10 CPI's with varying subband lengths and 2 subbands with low activity

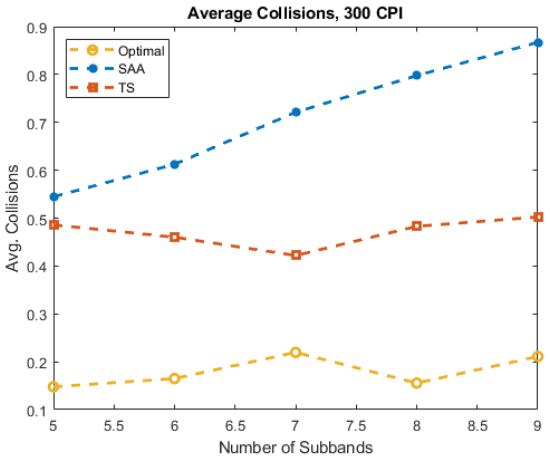


Figure 5.14: Average collisions between CPI 290 and CPI 300 with varying subband lengths and 2 subbands with low activity

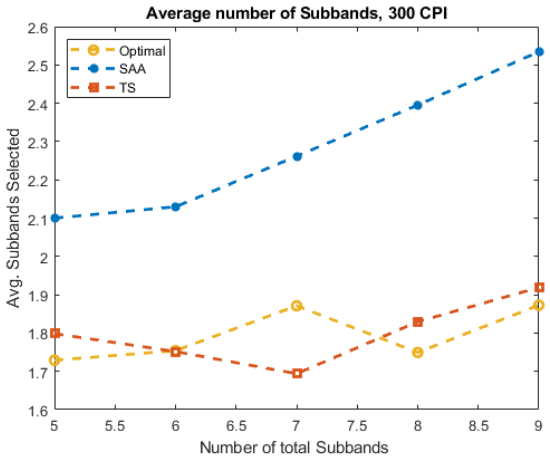


Figure 5.15: Average number of subbands between CPI 290 and CPI 300 with varying subband lengths and 2 subbands with low activity

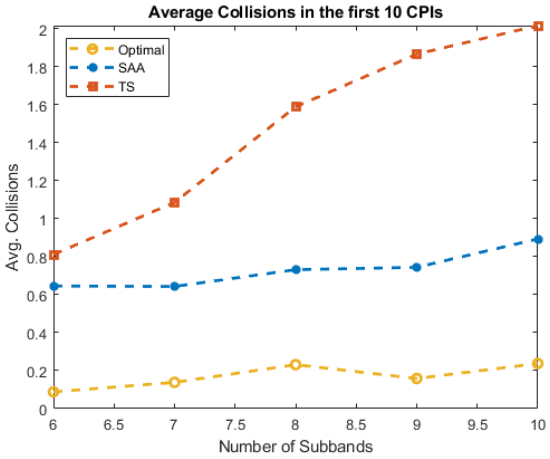


Figure 5.16: Average collisions of the first 10 CPI with varying subband lengths and 3 subbands of low activity

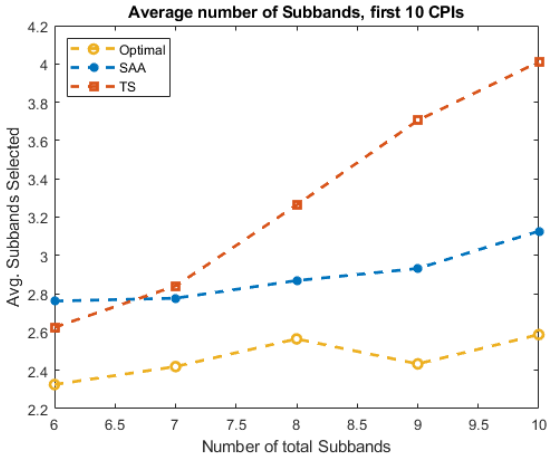


Figure 5.17: Average number of subbands selected in the first 10 CPI's with varying subband lengths and 2 subbands of low activity

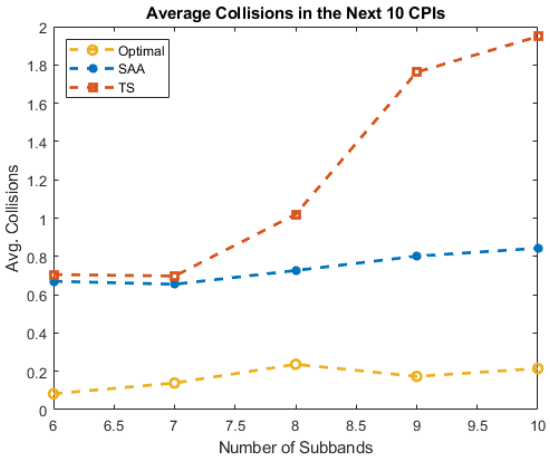


Figure 5.18: Average collisions of the next 10 CPI's with varying subband lengths and 3 subbands with low activity

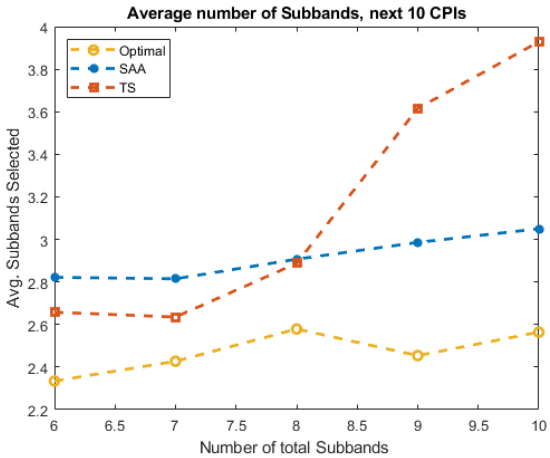


Figure 5.19: Average number of subbands selected in the next 10 CPI's with varying subband lengths and 3 subbands with low activity

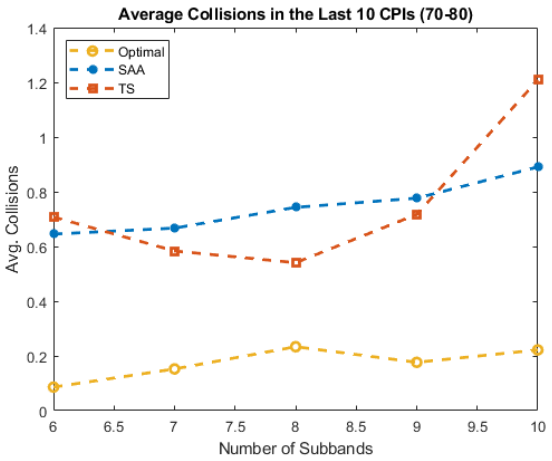


Figure 5.20: Average collisions of the last 10 CPI with varying subband lengths and 3 subbands of low activity

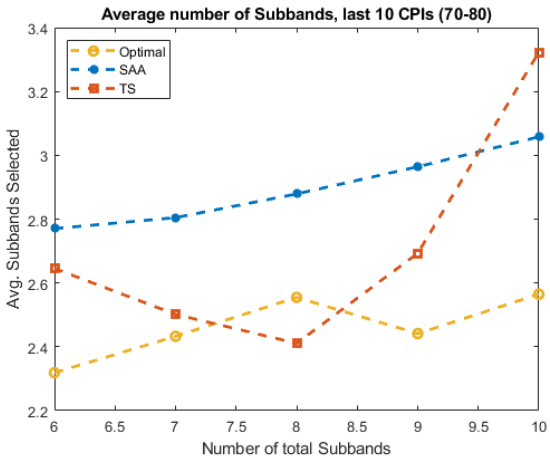


Figure 5.21: Average number of subbands selected in the last 10 CPI's with varying subband lengths and 2 subbands with low activity

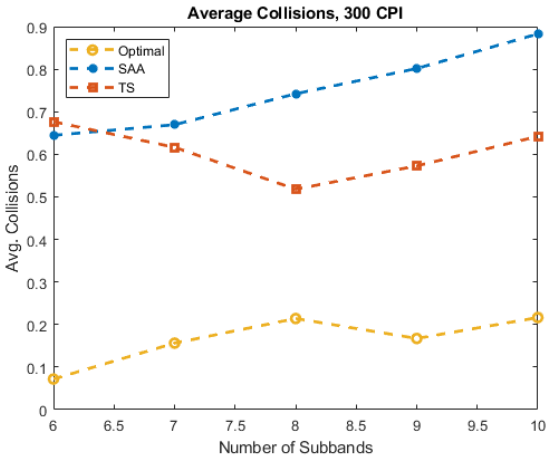


Figure 5.22: Average collisions between CPI 290 and CPI 300 with varying subband lengths and 3 subbands with low activity

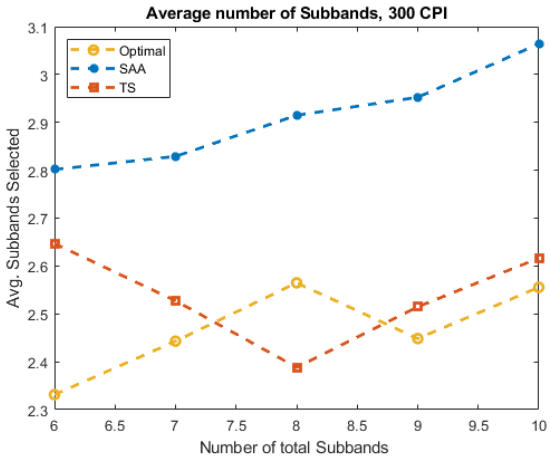


Figure 5.23: Average number of subbands between CPI 290 and CPI 300 with varying subband lengths and 3 subbands with low activity

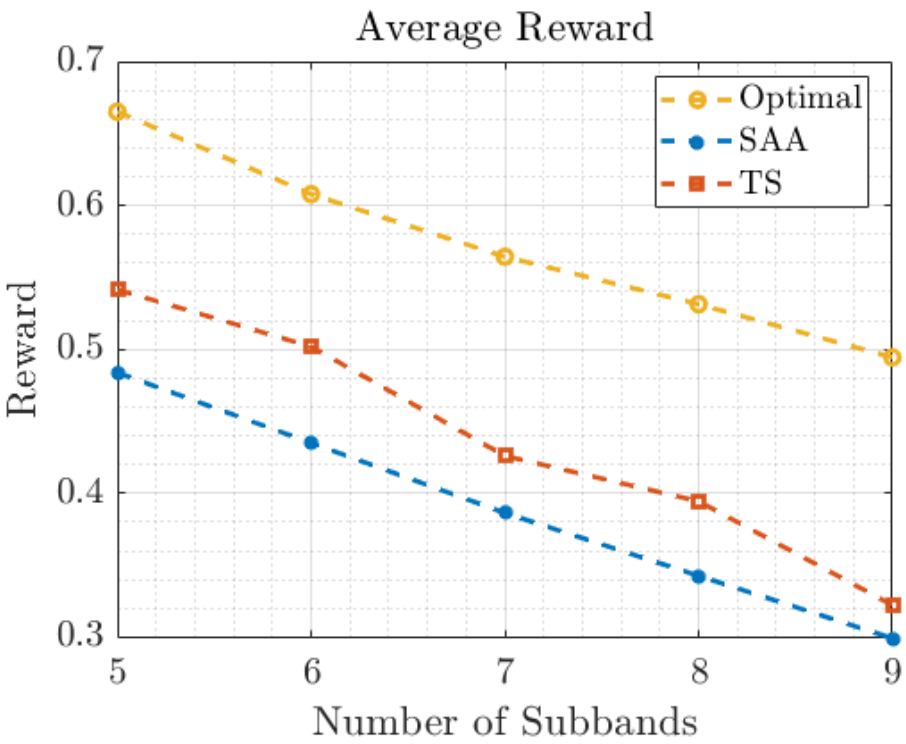


Figure 5.24: Average reward of all three agents with varying subband lengths and 3 subbands of low activity

## 5.4 Improper Thresholding

Although in simulation it is easy to tune parameters by adjusting constants and variables, a real world radar system may not be so easy to modify with possible changes to physical hardware required. In the experiment of thresholding effects on target tracking it is assumed that there are some parameters of the radar system that fall short and improperly report an interferer's subband selection. An example would be if a Neyman-Pearson detector's threshold is set too low, the radar system is too sensitive to noise and false alarms will be reported at the receiver although no actual interferer is present. Alternatively, if the same threshold is set too high, and the radar system is not sensitive enough to the received signal then interference may go unreported and a target could be lost without any explanation.

In the following experiment transition matrices and interferer state spaces are generated as usual, however an 80% "dropout rate" was applied to subband two regardless of the number of subbands. If the threshold was set too low then any time a "0" was encountered in subband two there was an 80% chance of it changing to a "1" and incorrectly reporting an occupied band. On the other hand if the threshold was set too high then the opposite would occur, incorrectly reporting no interference when a band was actually occupied.

Intuition serves as a sanity check to again understand what the expected outcomes should resemble. It is known that sense and avoid does not take interferer subbands at greater than face value and simply aims to avoid the current observation at the next time step. If we assume that what is currently observed occurs in the next time step then when the radar system reports an interference when there is none sense and avoid will always have at least one missed opportunity. Also, when the radar system reports no interference due to improper thresholding sense and avoid will always collide in subband two if the states do not change. This should give the impression that sense and avoid will rarely perform well

in this case if the current interferer is the same as the previous, however, if the subbands are constantly switching "on" and "off" then sense and avoid may get lucky at times. Both optimal and machine learning weigh their decisions based on the results of collisions and missed opportunities because it is important for them to investigate deeper into how their actions impact tracking results. Although the ML model is given the same interference subbands that sense and avoid is given, when reward is calculated it should notice that at times collisions occurs although it believes it has picked the optimal decision. The ML model should then be expected to recognize that it receives lower reward when it attempts to select subband two and either attempt to learn a pattern of when it can choose subband two or attempt to avoid it entirely. On the other hand, the Bellman optimal agent was given perfect knowledge of the threshold discrepancies to set an absolute best case where the ML agent is able to learn about this issue.

Figure 5.25 demonstrates the results of a radar threshold being set too low and ML's ability to detect false alarms and learn from this issue. Overall there is clear greater performance from the machine learning agent over sense and avoid, however optimal shows that there is still more to be learned from how the thresholding impacts the subbands. Figure 5.26 shows the case where a radar threshold is set too high and reporting no interference when there is one. In this case it is again seen that ML performs well when compared to sense and avoid and can learn to avoid the interferer even if it does not have perfect information about the subbands that it occupies.

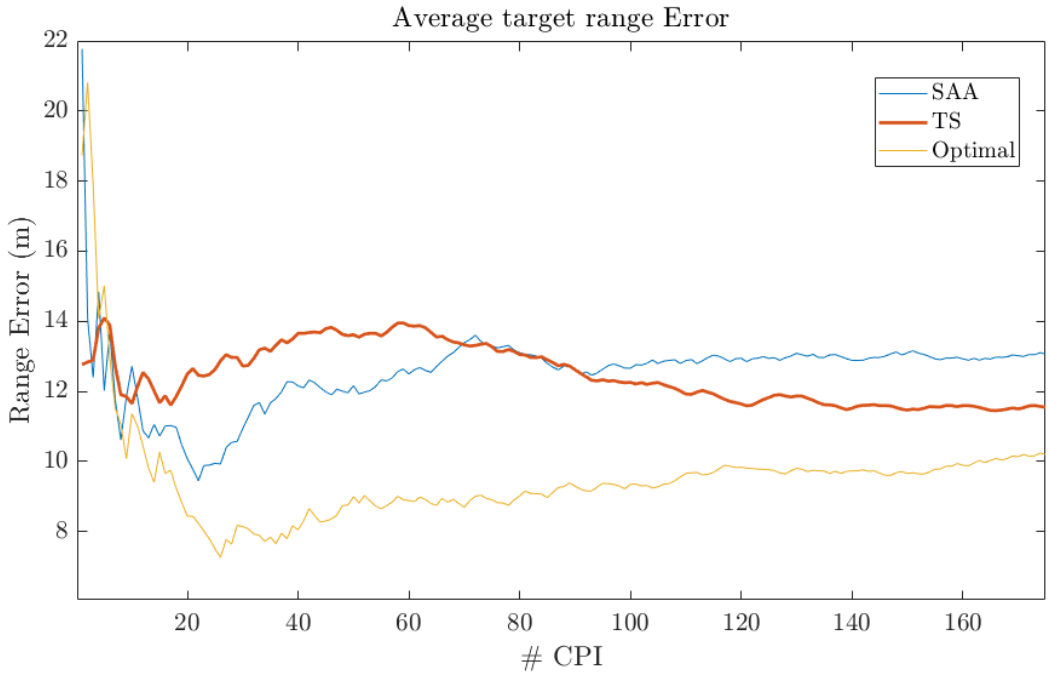


Figure 5.25: Average range error with a threshold set too low, incorrectly reporting subband 2.


Figure 5.26: Average range error with a threshold set too high, incorrectly reporting subband 2.

## 5.5 Spatial Correlation

In the previous scenarios an interferer that is encountered by the radar environment is always on and never allows the agent to select the entire bandwidth. In a realistic scenario an interferer may only collide with the radar system for a specific region of the tracking range. This study tackles the issue of the machine learning models' ability to track a target that is being jammed by a passive communications system only in a given region along a specific look-angle. This study is split into two separate cases.

In the first case the antenna only encounters the interferer once and the agents must choose subbands to avoid it. To simulate this, the interferer only begins transmitting after a specified number of CPI's pass. Here the target is passing across the radar system range at 22 m/s at a range of 350 meters. After 50 CPI the radar system begins to encounter the interferer and each of the three algorithms is tasked at dealing with the channel selection problem. At this point the interferer is not completely on but has a $P_i = 40\%$. After CPI = 70 the interferer is completely on and constantly transmitting in the same subbands as the radar system. Figure 5.27 demonstrates the target tracking error incurred by all three agents when tasked with avoiding the partial interferer after the previously specified amount of time, this data has been normalized and filtered to properly convey the running averages of the tracking error. It is clear that as the interferer first begins sense and avoid is actually preforming slightly better than machine learning as it still has to learn the pattern that is being observed. Once the interferer is fully "on" and always transmitting through the subbands, machine learning very quickly takes over and preforms much better than sense and avoid in the target tracking problem. However, it should still be noted that there is more to be desired from the machine learning model when the bellman optimal agent is able to track the target at a higher accuracy than the ML model. The bellman optimal agent is actually nearly perfect the entire run until the $90^{th}$ CPI where the interferer gets into more

complicated subband decisions.

In the second case, shown in 5.28, the simulated target passes over the interference region several times. This is a good scenario for the machine learning model as it demonstrates its ability to learn where the interference zones are and recall this when the radar system enters it again. In this case the context of the thompson sampling model has to be updated with the location of the target. The target is defined by its range from the radar system and angle $\theta$ made with the antenna. In this case optimal is no longer considered the bellman optimal since it is not given context into how the target is moving along the radar range. Instead, a perfectly optimal agent is used to evaluate an absolute best case where the channel selection agent chooses exactly perfect contiguous subbands while avoiding the interferer.

The radar system simulated in this scenario sweeps between $\frac{\pi}{4}$ and $\frac{3\pi}{4}$ and the interferer is located and fully on centered at $\frac{\pi}{2}$ with a range of $\frac{\pi}{20}$. Given the velocity of the target and range that is covered in the interference zone this allows for 11 CPI's in each pass over the interference region. 5.28 demonstrates the 4 unique passes of the radar system over the interference region while tracking the simulated target passing by. Starting from the darkest red, it is seen that the machine learning model preforms only slightly better than sense and avoid on any single run over the interference zone. However, after just 2 passes over the interference region it is seen that the machine learning model is able to quickly adapt to and predict the interference region, successfully reducing estimation error. Finally, in the $4^{th}$ pass of the target over the interference region (bright red), the machine learning agent can successfully predict the region of interference and acts near optimal.

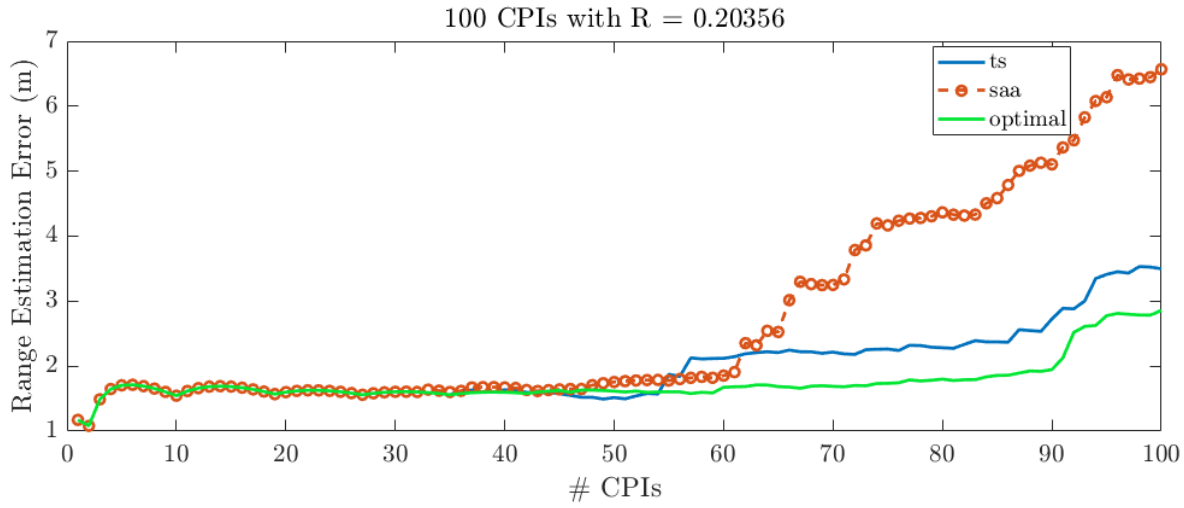


Figure 5.27: Range estimation error as the antenna begins to move into an area that contains an interferer.

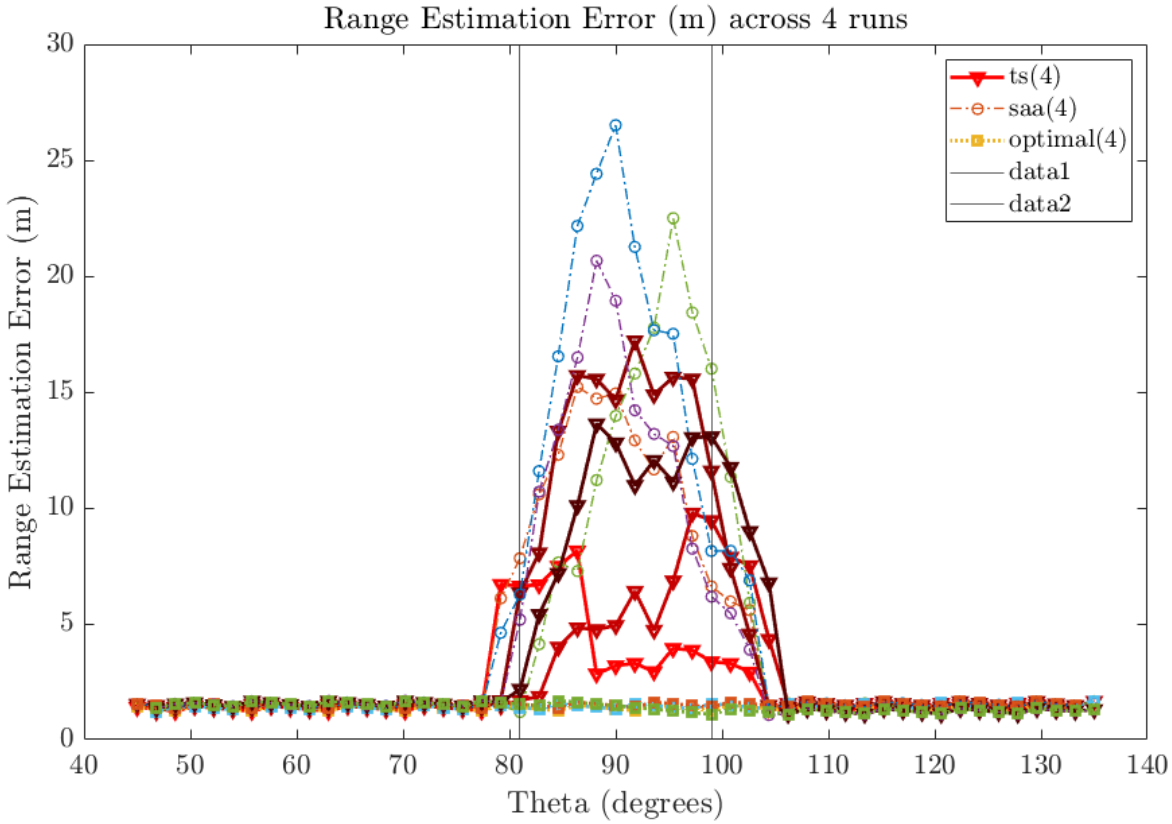


Figure 5.28: 4 continuous passes over a region that contains an interferer. This figure demonstrates ML's ability to learn where an interferer lies and anticipate and avoid it to properly track a target.

# Chapter 6

# Conclusion

This work has demonstrated the value of machine learning when applied to the radar communications coexistence problem. The results have demonstrated that a machine learning agent is able to learn from and adapt to the interference behavior of a communication system and adjust a radar subband usage to successfully track a target. Through several experiments the machine learning agent has shown that it is advantageous over a classical sense and avoid technique and capable of converging towards a Bellman optimal solution if given sufficient time. Through the reinforcement learning implementation the radar system is now enabled with memory and adaptability over classical sense and avoid techniques. Without these adaptations a radar system would have to be manually developed to select actions state by state for each potential interferer, instead now this can be learned by a cognitive radar relatively quick and implemented with good accuracy. The proof of concept for online learning in radar scenarios is strengthened here as it is shown that these models do not need a batch of data before being able to make decisions, they can be made on the go and improved as data arrives at radar.

The results in this thesis have indicated that

1. An ML agent is very well equipped to properly select contiguous subbands when compared to sense and avoid particularly when interferers are very non-deterministic.

2. When a communications interferer is only occupying some of the subbands that the

radar has access to, the ML agent can learn a pattern of unoccupied bands and accurately select bands that maximize bandwidth and reward.

3. If information about a subband that contains an interferer is improperly reported at the radar system, the ML agent can properly identify the scenario and adjust accordingly. This is shown to be true in both the case where an interferer exists but goes unreported (missed detection) and where an intereferer is reported but does not exist (false alarm). Alternatively, classical sense and avoid techniques are not capable of identifying and adjusting for this misinformation.

4. The ML agent is able to learn and expect a spatially consistent interferer in a given region once it is provided with information about the current target location and angle.

Future work will involve revisiting assumptions made in this thesis and studying the effects of each. Specifically this could entail studying

1. Evaluating the time to converge to an optimal policy for an ML algorithm when faced with a high priority, fast moving target such as a missile.

2. Modeling an actual communications interferer such as LTE, WiFi, 3GPP, etc;

3. Actual hardware experiments for a cognitive radar testbed;

4. Modelling an intelligent interferer;

5. Effects of real world environmental impacts and how ML will converge with varying reward functions in these scenarios.

Additional future work should study optimizing reward functions to generate optimal policies for these scenarios. Also a study that used a much larger bandwidth or number of subbands

might demonstrate the time complexity of an ML algorithm convergence and how it might function in that scenario. Other applications may be adapting the actual radar simulation section where noise and clutter are applied to the received signals. Target dependent clutter is discarded in this thesis because of the assumption of a point target model, however, a complex target model should be investigated in future work. Waveform selection and optimization can also be studied, such as NLFM and phase coded, as only an LFM is investigated here.

# Bibliography


[1] K. B. R. Cafe, "Radar techniques - primer principles, https://www.rfcafe.com/references/qst/radar-techniques-primer-principles-apr-1945-qst.htm."

[2] "Federal communications commission, https://www.fcc.gov/auction/107," Apr 2022.

[3] Hollmann, "Christian huelsmeyer, the inventor https://www.radarworld.org/huelsmeyer.html," Radar World, 2007.

[4] M.A.Richards and W. A. Holm, *Principles of Modern Radar*. Institution of Engineering and Technology, 2010.

[5] D. Hebb, *The Organization of Behavior*. Wiley, 1949.

[6] G. V. Keysers, Christian, "Hebbian learning and predictive mirror neurons for actions, sensations and emotions," *PubMed*, 2014 Apr 28.

[7] A. L. Samuel, "Some studies in machine learning using the game of checkers," *IBM Journal of Research and Development*, vol. 3, no. 3, pp. 210–229, 1959.

[8] A. L. Samuel, "Some studies in machine learning using the game of checkers. ii—recent progress," *IBM Journal of Research and Development*, vol. 11, no. 6, pp. 601–617, 1967.

[9] T. M. Mitchell, *Machine learning*. McGraw-Hill, 1997.

[10] K. P. Murphy, *Machine learning: A probabilistic perspective*. MIT Press, 2021.

[11] D. Steinkraus, I. Buck, and P. Simard, "Using gpus for machine learning algorithms," in *Eighth International Conference on Document Analysis and Recognition (ICDAR'05)*, pp. 1115–1120 Vol. 2, 2005.

[12] K.-S. Oh and K. Jung, "GPU implementation of neural networks," *Pattern Recognition*, vol. 37, pp. 1311–1314, Jan. 2004.

[13] M. Martorella, E. Giusti, A. Capria, F. Berizzi, and B. Bates, "Automatic target recognition by means of polarimetric isar images and neural networks," in *IGARSS 2008 - 2008 IEEE International Geoscience and Remote Sensing Symposium*, vol. 4, pp. IV – 1249–IV – 1252, 2008.

[14] C. Bentes, D. Velotto, and S. Lehner, "Target classification in oceanographic sar images with deep neural networks: Architecture and initial results," in *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 3703–3706, 2015.

[15] P. Molchanov, J. Astola, K. Egiazarian, and A. Totsky, "Ground moving target classification by using dct coefficients extracted from micro-doppler radar signatures and artificial neuron network," in *2011 MICROWAVES, RADAR AND REMOTE SENSING SYMPOSIUM*, pp. 173–176, 2011.

[16] A. Stroescu, L. Daniel, D. Phippen, M. Cherniakov, and M. Gashinova, "Object detection on radar imagery for autonomous driving using deep neural networks," in *2020 17th European Radar Conference (EuRAD)*, pp. 120–123, 2021.

[17] S. R. Shebert, A. F. Martone, and R. M. Buehrer, "Wireless standard classification using convolutional neural networks," in *2021 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, 2021.

[18] X. Liu, D. Yang, and A. E. Gamal, "Deep neural network architectures for modulation classification," arXiv, 2017.

[19] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4580–4584, 2015.

[20] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. The MIT Press, 2020.

[21] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan, and D. Hassabis, "Mastering chess and shogi by self-play with a general reinforcement learning algorithm," 2017.

[22] K. Arulkumaran, A. Cully, and J. Togelius, "AlphaStar: An Evolutionary Computation Perspective," in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, ACM, jul 2019.

[23] "Alphastar: Mastering the real-time strategy game starcraft ii," DeepMind, 2019.

[24] OpenAI, :, C. Berner, G. Brockman, B. Chan, V. Cheung, P. Dębiak, C. Dennison, D. Farhi, Q. Fischer, S. Hashme, C. Hesse, R. Józefowicz, S. Gray, C. Olsson, J. Pachocki, M. Petrov, H. P. d. O. Pinto, J. Raiman, T. Salimans, J. Schlatter, J. Schneider, S. Sidor, I. Sutskever, J. Tang, F. Wolski, and S. Zhang, "Dota 2 with large scale deep reinforcement learning," arXiv, 2019.

[25] M. L. Puterman, *Markov decision processes: Discrete stochastic dynamic programming*. Wiley, 2005.

[26] M. van Otterlo and M. Wiering, *Reinforcement Learning and Markov Decision Processes*, pp. 3–42. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012.

[27] A. Slivkins, "Introduction to multi-armed bandits," arXiv, 2019.

[28] W. W. Howard, C. E. Thornton, A. F. Martone, and R. Michael Buehrer, "Multi-player bandits for distributed cognitive radar," in *2021 IEEE Radar Conference (RadarConf21)*, pp. 1–6, 2021.

[29] S. Kang and C. Joo, "Combinatorial multi-armed bandits in cognitive radio networks: A brief overview," in *2017 International Conference on Information and Communication Technology Convergence (ICTC)*, pp. 1086–1088, 2017.

[30] D. Kalathil, N. Nayyar, and R. Jain, "Multi-player multi-armed bandits: Decentralized learning with iid rewards," in *2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 853–860, 2012.

[31] M. S. Greco, F. Gini, P. Stinco, and K. Bell, "Cognitive radars: On the road to reality: Progress thus far and possibilities for the future," *IEEE Signal Processing Magazine*, vol. 35, no. 4, pp. 112–125, 2018.

[32] S. Haykin, "Cognitive radar: a way of the future," *IEEE Signal Processing Magazine*, vol. 23, no. 1, pp. 30–40, 2006.

[33] A. F. Martone, K. I. Ranney, K. Sherbondy, K. A. Gallagher, and S. D. Blunt, "Spectrum allocation for noncooperative radar coexistence," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 1, pp. 90–105, 2018.

[34] M. S. Burgin, L. Mandrake, G. B. Doran, B. D. Bue, and J. J. Van Zyl, "Soil moisture estimation by linear regression from smap polarimetric radar data with aquarius de-

rived coefficients," in *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, pp. 1816–1817, 2018.

[35] M. S. Burgin and J. J. van Zyl, "Analysis of polarimetric radar data and soil moisture from aquarius: Towards a regression-based soil moisture estimation algorithm," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 9, no. 8, pp. 3497–3504, 2016.

[36] A. Javed, A. Ejaz, S. Liaqat, A. Ashraf, and M. B. Ihsan, "Automatic target classifier for a ground surveillance radar using linear discriminant analysis and logistic regression," in *2012 9th European Radar Conference*, pp. 302–305, 2012.

[37] H. Nikaein, A. Sheikhi, and S. Gazor, "Target detection in passive radar sensors using least angle regression," *IEEE Sensors Journal*, vol. 21, no. 4, pp. 4533–4542, 2021.

[38] Y. Liu, J. Yi, X. Wan, X. Zhang, and H. Ke, "Evaluation of clutter suppression in cp-ofdm-based passive radar," *IEEE Sensors Journal*, vol. 19, no. 14, pp. 5572–5586, 2019.

[39] C. Palmarini, T. Martelli, F. Colone, and P. Lombardo, "Disturbance removal in passive radar via sliding extensive cancellation algorithm (eca-s)," in *2015 IEEE Radar Conference*, pp. 162–167, 2015.

[40] X. Feng, X. Hu, and Y. Liu, "Radar signal sorting algorithm of k-means clustering based on data field," in *2017 3rd IEEE International Conference on Computer and Communications (ICCC)*, pp. 2262–2266, 2017.

[41] J. Macqueen, "Some methods for classification and analysis of multivariate observations," in *In 5-th Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281–297, 1967.

[42] A. Martone, R. Innocenti, and K. Ranney, "An analysis of clustering tools for moving target indication," Army Research Lab Adelphi MD, Defense Technical Information Center, 2009.

[43] A. Martone, K. Ranney, and R. Innocenti, "Through-the-wall detection of slow-moving personnel," in *Radar Sensor Technology XIII* (K. I. Ranney and A. W. Doerry, eds.), vol. 7308, pp. 225 – 236, International Society for Optics and Photonics, SPIE, 2009.

[44] A. K. Jain, *Fundamentals of Digital Image Processing*. Prentice-Hall of India, 2006.

[45] A. Martone, K. Ranney, A. Hedden, G. Mazzaro, and D. McNamara, "Cognitive processing for nonlinear radar," in *Radar Sensor Technology XVII* (K. I. Ranney and A. Doerry, eds.), vol. 8714, pp. 144 – 153, International Society for Optics and Photonics, SPIE, 2013.

[46] P. Stinco, M. Greco, F. Gini, and B. Himed, "Spectrum sensing and sharing for cognitive radar systems," Dipartimento di Ingegneria dell'Informazione, University of Pisa, 2016.

[47] C. E. Thornton, R. M. Buehrer, and A. F. Martone, "Efficient online learning for cognitive radar-cellular coexistence via contextual thompson sampling," in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, pp. 1–6, 2020.

[48] J. Liu, J. Lee, L. Li, Z.-Q. Luo, and K. Wong, "Online clustering algorithms for radar emitter classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1185–1196, 2005.

[49] D. Russo, "A tutorial on thompson sampling," arxiv, 2020.

[50] M. Kozy, J. Yu, R. M. Buehrer, A. Martone, and K. Sherbondy, "Applying deep-q networks to target tracking to improve cognitive radar," in *2019 IEEE Radar Conference (RadarConf)*, pp. 1–6, 2019.

[51] O. Chapelle and L. Li, "An empirical evaluation of thompson sampling," in *Advances in Neural Information Processing Systems* (J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, eds.), vol. 24, Curran Associates, Inc., 2011.

[52] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[53] C. Thornton, "Learning schemes for adaptive spectrum sharing radar," Virginia Polytechnic Institute and State University, 2020.

[54] E. Selvi, "Cognitive radar applied to target tracking using markov decision processes," Virginia Polytechnic Institute and State University, 2018.

[55] M. Kozy, "Creation of a cognitive radar with machine learning: Simulation and implementation," Virginia Polytechnic Institute and State University, 2019.

[56] C. E. Thornton, R. M. Buehrer, and A. F. Martone, "Constrained contextual bandit learning for adaptive radar waveform selection," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 58, no. 2, pp. 1133–1148, 2022.