DISSERTATION

TARGET TRACKING WITH DISTRIBUTED SENSING: INFORMATION-THEORETIC BOUNDS AND CLOSED-LOOP SCHEDULING FOR URBAN TERRAIN

Submitted by

Patricia de Rezende Barbosa

Department of Electrical and Computer Engineering

In partial fulfillment of the requirements

For the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Spring 2010

TK7872
D48
R495
2010

COLORADO STATE UNIVERSITY

April 5, 2010

WE HEREBY RECOMMEND THAT THE DISSERTATION PREPARED UNDER OUR SUPERVISION BY PATRICIA DE REZENDE BARBOSA ENTITLED TARGET TRACKING WITH DISTRIBUTED SENSING: INFORMATION-THEORETIC BOUNDS AND CLOSED-LOOP SCHEDULING FOR URBAN TERRAIN BE ACCEPTED AS FULFILLING IN PART REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY.

Committee on Graduate work

Louis L. Scharf

J. Rockey Luo

Chihoon Lee

Advisor: Edwin K. P. Chong

Department Head: Anthony A. Maciejewski

COLORADO STATE UNIV. LIBRARIES

ABSTRACT OF DISSERTATION

TARGET TRACKING WITH DISTRIBUTED SENSING: INFORMATION-THEORETIC BOUNDS AND CLOSED-LOOP SCHEDULING FOR URBAN TERRAIN

We address both theoretical and practical aspects of target tracking in a distributed sensing environment. First, we consider the problem of tracking a target that moves according to a Markov chain in a sensor network. We provide necessary and sufficient conditions on the number of queries per time step to track a target in three different scenarios: (1) the tracker is required to know the exact location of the target at each time step; (2) the tracker may lose track of the target at a given time step, but it is able to "catch-up", regaining up-to-date information about the target's track; and (3) tracking information is only known by the tracker after a delay of d time steps. We then address the problem of target tracking in urban terrain. Specifically, we investigate the integration of detection, signal processing, tracking, and scheduling, by simultaneously exploiting three diversity modes: (1) spatial diversity through the use of coordinated multistatic radars; (2) waveform diversity by adaptively scheduling the transmitted waveform; and (3) motion model diversity by using a bank of parallel filters matched to different motion models. A closed-loop active sensing system is presented, and Monte Carlo simulations demonstrate its effectiveness in urban terrain. Finally, we propose a scheduling scheme that adaptively selects the sequence of transmitters and waveforms that maximizes the overall tracking accuracy, while maintaining the sensing system's covertness in a hostile environment. We formulate this problem as a POMDP and use two distinct schedulers: (1) a myopic scheduler that updates waveforms at every radar scan; and (2) a non-myopic scheduler that activates a new set of transmitters if the overall tracking accuracy falls below a threshold or if a detection risk is exceeded. By simultaneously exploiting myopic and non-myopic scheduling schemes, we benefit from trading off short-term for long-term performance, while maintaining low computational costs. Monte Carlo simulations are used to evaluate the proposed scheduling scheme in a multitarget tracking setting.

Patricia de Rezende Barbosa
Department of Electrical and Computer Engineering
Colorado State University
Fort Collins, CO 80523
Spring 2010

ACKNOWLEDGEMENTS

I would like to thank Prof. Edwin Chong, my advisor, for his many suggestions and constant support during this research. His guidance was essential throughout my studies. I am also thankful to my research committee members, Prof. Chihoon Lee, Prof. Louis Scharf, and Prof. J. Rockey Luo.

I am especially grateful to my friends, for their understanding of the many times I needed to sacrifice our time together.

Last but not least, I thank my parents for their never-ending support, patience, and love. I am forever in your debt.

The work presented here was financially supported in part by the Shrake/Culler Scholarship and Sjostrom Scholarship.

Patricia de Rezende Barbosa

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

Introduction

This chapter starts with the motivation behind the proposed dissertation and revisits key historical aspects that led to this study, followed by a summary of contributions.

1.1 Motivation and Literature Review

It often happens in our everyday life that we receive information not all at once but bit by bit, so the complete information is a result of the accumulation of partial information. In many situations, however, it is not possible to wait until all information is gathered, and we are forced to make decisions and take actions under uncertainty. Few problems are more universal than that of decision-making under uncertainty; in this dissertation, we consider a few of its instances. Although the use of scheduling and decision-making in this work is motivated by applications in target tracking, the solution techniques presented here can be applied to a wide range of situations. In a world of limited resources — natural and man-made — it is not difficult to envision numerous other applications that would benefit from effective scheduling techniques.

1.1.1 Zero-Error Target Tracking Through Limited Querying of Binary Sensors

In Chapter 2, we examine the problem of target tracking through limited querying in a sensor network setting. We explore the similarities between this problem and Rényi-Ulam games [23, 73, 89], of which the game of "twenty-questions" can be considered a subclass. Our goal is to find theoretical bounds on the number of queries per time step a tracker is required to ask a set of sensors in order to track a target. We consider three different scenarios: (1) the tracker is required to know the exact location of the target at each time step; (2) the tracker may lose track of the target at a given time step, but it is able to "catch-up", regaining up-to-date information about the target's track; and (3) tracking information is only known by the tracker after a delay of d time steps.

Sensor networks have emerged as one of the most promising technologies in recent years. While much of the research done in this area explores networking issues like time synchronization [28, 22], sensor localization [35, 13], and routing [87, 29], additional communications problems such as data compression and message complexity have become increasingly important as the number of networked sensing devices continues to grow. For the majority of existing sensor networks, these small and inexpensive devices impose serious energy constraints affecting the network lifetime by having to transmit sensing information (over possibly long communication channels) to a remote monitoring station (tracker) [14]. Moreover, the reliability and the capacity of the channel available for communication with the tracker lead to restrictions on the amount of data sent over such networks. As a consequence, the tracker needs to make judicious decisions when selecting sensors to send data, so that communication with the sensor network is kept to a minimum. We assume a sensor model where sensors are capable of sending only one-bit messages to a tracker, which are used to gather tracking information about a target that moves according to a Markov chain.

In the literature, one-bit-message sensor networks are called binary sensor networks and have been previously considered for target tracking [3, 58, 59]. In [25], Evans et al. analyzed the problem of optimal sensor selection; however, their approach is to formulate the problem as a partially observed stochastic control problem, where sensors are not constrained to one-bit messages and the tracker also controls the channel data rate so that mean squared errors are bounded. Related problems also include those in the area of control under communication constraints. In particular, Tatikonda and Mitter [83, 85, 86] examined a control problem with a noiseless channel rate required to achieve different control objectives, namely, asymptotic observability and asymptotic stabilizability [84]. Sahai and Mitter [77] investigated the problem of tracking and controlling unstable processes over noisy channels and demonstrated that Shannon's classical notion of capacity was insufficient to characterize noisy channels for this purpose. Furthermore, they identified a novel characterizing quantity called anytime capacity and showed that it is both necessary and sufficient for channels to have a certain amount of anytime capacity such that unstable processes can be tracked and stabilized.

1.1.2 Closing the Loop on Target Tracking in Urban Terrain

In Chapters 3 and 4, we consider a more applied, though unconventional, approach to target tracking in a non-traditional setting. Specifically, we consider the design of a "closed-loop" active sensing platform for target tracking in urban terrain. Once again, we deal with uncertainty, though now from a different perspective. When tracking multiple targets in urban terrain, there is an inherent uncertainty caused by random variations of measurement data and modeling inaccuracies. Uncertainty can also be associated with the measurement origin in the presence of clutter and multipath, or when multiple targets are in the same neighborhood.

Motivated by the shift of battlefields — from open areas to urban canyons — we contribute to the experimental basis of the next generation of tracking and surveillance systems. Specifically, we investigate the integration of detection, signal processing, tracking, and scheduling by exploiting distinct levels of diversity: (1) spatial diversity through the use of coordinated multistatic radars; (2) waveform diversity by adaptively scheduling the transmitted radar waveform according to the scene conditions; and (3) motion model diversity by using a bank of parallel filters, each one matched to a different maneuvering model. We start by modeling the different elements of the urban environment in Chapter 3. Although intrinsically imperfect due to many simplifying assumptions, the models presented facilitate the analysis of the interplay among clutter, multipath, sensors, signals, and target motion, which ultimately affect the overall system performance. In Chapter 4, we design and simulate a closed-loop active sensing system for tracking multiple targets in urban terrain.

Traditionally, the approach to tracking systems design has been to treat sensing and tracking sub-systems as two completely separate entities. While many of the problems involved in the design of such sub-systems have been individually examined in the literature from a theoretical point-of-view, very little attention has been devoted to the challenges involved in the design of an active sensing platform that simultaneously addresses detection, signal processing, tracking, and scheduling. In addition, the problem of tracking under urban conditions remains relatively unexplored and the literature is particularly scarce on works that discuss closed-loop tracking systems. One of the few to tackle this problem, Sanders-Reed [78] examined integration issues of a multitarget tracking system using video sensors and sensor-pointing commands to close the feedback loop. Computational aspects of a closed-loop image-based tracker of airborne targets were considered by Robinson and Sasaki [75]. An overview of systems-level modeling for the performance evaluation of closed-loop tracking systems for naval applications was given by Beeton and Hall in [7].

Note that waveform scheduling was not considered in any of these previous studies. However, the quality of radar measurements and, ultimately, target detection and the overall tracking performance depend on the transmitted waveform. Many modern radar systems are able to exploit waveform diversity by selecting waveforms from a programmable library, and changing the transmitted waveform in real time. This selection is done according to past responses from the surveillance area, which depend on both intra-pulse characteristics (e.g., carrier frequency and bandwidth) as well as on inter-pulse characteristics (e.g., pulse repetition frequency) of the transmitted waveform. The problem of one-step-ahead waveform scheduling for tracking systems was investigated by Kershaw and Evans in [38], while the multi-step-ahead case was considered by Suvorova et al. in [81]. An overview of emerging ideas in the area of waveform scheduling for active radar can be found in [18].

One of the few studies focusing on the urban environment is the recent case study of urban operations for counter-terrorism, which was analyzed using a probability of attack integrated into a multitarget tracking system, proposed by Sathyan et al. in [79]. Guerci and Baranoski [31] provided an overview of a knowledge-aided airborne adaptive radar system for tracking ground targets. A lookahead sensor scheduling approach was presented but, as the authors acknowledged, they were "merely scratching the surface" of a possible solution. A vehicular sensing platform that exploits recent advances in inter-vehicle communication and on-board sensor technology was considered in [51], where a new communication protocol was also proposed. Although not particularly tailored to target tracking applications, this mobile sensing approach has emerged as an attractive lightweight solution for opportunistic monitoring and proactive surveillance in urban terrain, given the abundant presence of sensor-equipped vehicles in such environments [50].

1.1.3 Two-Level Scheduling for Target Tracking in Covert Operations

In Chapter 5, we revisit the limitation on the number of sensor transmissions discussed in Chapter 2 and propose a novel adaptive sensing scheme for target tracking in a hostile environment. Specifically, the "two-level" scheduling scheme uses two distinct schedulers: (1) a myopic scheduler that updates waveforms at every radar scan; and (2) a non-myopic scheduler that activates a new set of transmitters if the overall tracking accuracy falls below a threshold or if a detection risk is exceeded. The two-level scheduling scheme is implemented by a central controller whose goal is to select the sequence of combinations of transmitters and waveforms that yields the most accurate tracking estimate according to a suitably chosen performance evaluation criterion. The complexity of the controller's operation is substantially increased when sensor covertness is involved. This detection risk constraint is modeled by a limit on the number of consecutive time steps a transmitter can stay activated.

In the computer science community, the term two-level scheduling describes a method of scheduling computer processes [82]. Similar to our proposed scheduling scheme, two different schedulers are used. At the lower level, the scheduler can only select processes already available in memory. In this case, processes are usually scheduled in a round-robin fashion for fast process switching. When the computer is low on memory, or when a process stored in disk needs to run, swapping processes in and out of memory is required. In this case, the upper level scheduler is involved, but since swapping processes between disk and memory is time-consuming, scheduling at the upper level happens much less often than at the lower level. A similar phenomenon occurs when scheduling radars for adaptive sensing. In general, the transition from idle mode to active mode is not instantaneous for radar transmitters, that is, they need some time to turn on and warm up until they become operational. Hence, measurements taken during start-up are often discarded in practice due to possible signal fluctuations that could cause large measurement errors [80]. Therefore, in our proposed scheme, switching transmitters on and off only happens if certain conditions are met, while switching waveforms from a programmable library in a transmitter that is already active is done in real time, hence waveform scheduling can be happen frequently.

In the literature, different solution approaches to the adaptive sensing problem have been proposed, in which the goal is to make decisions over time under uncertainty on the use of sensor resources to maximize future sensing performance. Adaptive sensing for single target tracking was previously investigated by He and Chong [33], where the sensor scheduling problem was formulated as a partially observable Markov decision process (POMDP). Multitarget results using the same solution framework were presented in [57]. A POMDP formulation was also used by Miller et al. to model the coordinated guidance of autonomous aerial vehicles (UAVs) [63]. In [31], Guerci and Baranoski provided an overview of a knowledge-aided airborne adaptive radar system for tracking ground targets. Reinforcement learning methods were used to schedule sensors to most quickly and effectively detect and track smart targets in [45]. In [92], Wu and Cameron studied the general mathematical framework for applying Bayesian decision theory to optimal sensor placement. Fuzzy logic and genetic algorithms were two of the solution approaches for scheduling prioritized tasks in sensor networks studied in [20].

Resource allocation strategies can be classified as myopic or non-myopic. In the myopic case, the objective is to maximize the immediate reward, which is usually translated to minimizing the instantaneous estimation error in target tracking applications. Myopic strategies do not explicitly consider long-term performance, hence are less computationally expensive than their non-myopic counterparts, and have been extensively studied in the adaptive sensing literature [6, 94, 47, 65, 38, 41]. Non-myopic strategies are capable of trading off short-term performance for long-term performance by taking into account the future benefit of current actions via lookahead [81, 48, 49, 46]. However, optimal non-myopic strategies present prohibitive computational costs to most adaptive sensing applications, and therefore approximate solution techniques or heuristics are often adopted [17, 33]. While waveform agility implemented in a myopic fashion can significantly improve the tracking estimate [6, 38, 81], the need for non-myopic decision making in target tracking is greatly accentuated in a hostile environment, where maintaining covertness is imperative to protect deployed assets. Hence, in addition to waveform diversity, adaptive activation of sensors based on a measure of the risk incurred by this activation is needed. The concept of risk to sensors was previously considered as the threat imposed by targets, and calculated as the probability of attack based on target kinematics [79]. Due to their active sensing behavior, radar transmitters can be especially susceptible to detection by hostiles. By limiting the number of consecutive scans during which each transmitter remains active, and using a non-myopic scheduling strategy, we are able to reduce the detection risk and still maintain the desired long-term performance.

1.2 Summary of Contributions

The contributions from this work are the following:

- In Chapter 2 we determine the minimum number of queries per time step such that a target is trackable under three different tracking definitions. In particular, we show that: (1) the minimum number of queries per time step necessary (and sufficient) to follow a target is $\max_{i \in \mathcal{X}} \log |E_i|$, where E_i denotes the set of neighboring sensors of sensor i and binary search is a following strategy; (2) the minimum number of queries per time step necessary to track a target is at least equal to H, the entropy rate of the underlying Markov chain, and $H + 1$ queries per time step are sufficient to track; (3) if a fixed delay of $d \geq 1$ time steps is tolerable, $H + \frac{1}{d}$ queries per time step are sufficient for d-tracking, while at least H queries per time step are necessary in this case as well. Both tracking and d-tracking strategies are based on Huffman coding.

- Another important and innovative aspect of the work described in Chapter 2 is related to the similarities between the target tracking problem in a sensor network setting and Rényi-Ulam games. To the best of our knowledge, this study is the first to explore the duality between these two problems, providing the notable simplicity of our tracking scheme; simplicity always sought for, but never really achieved, in previous works [24, 59, 60, 91].

- In Chapter 4, we demonstrate through Monte Carlo simulations a closed-loop active sensing system that simultaneously: (1) increases the number of confirmed (true) tracks by approximately 15%; and (2) reduces the position root mean square error (RMSE) for a tracked target by several tens of % when compared to more traditional systems. We achieve these results by exploiting distinct levels of diversity: (1) spatial diversity through the use of coordinated multistatic radars; (2) waveform diversity by adaptively scheduling the transmitted radar waveform according to the scene conditions; and (3) motion model diversity by using a bank of parallel filters, each one matched to a different maneuvering model.

- From a systems engineering point-of-view, the work presented in Chapters 3 and 4 exposes the challenges and difficulties of integrating detection, signal processing, tracking, and scheduling in a single closed-loop platform for target tracking in an urban environment. To the best of our knowledge, this work is the first to design and analyze a specific tracking system that addresses all of these issues at once.

- Chapter 5 describes the novel two-level scheduling scheme for management of waveform-agile radars for target tracking in a hostile environment. We formulate the proposed adaptive sensing problem using the POMDP framework, and illustrate the performance gains of the order of several tens of % in position (RMSE) that can be achieved by the novel two-level scheduling scheme with only moderate increase in computational complexity when compared to a fully-myopic scheduler. Moreover, we show that the myopic scheduling of waveforms at each radar scan improves on non-myopic actions taken in the past approximately 8% of the time.

Chapter 2

Zero-Error Target Tracking Through Limited Querying of Binary Sensors

We consider the problem of tracking a target that moves according to a Markov chain using a tracker that queries a set of sensors to obtain tracking information. We are interested in finding the minimum number of queries per time step such that a target is trackable. Three scenarios are analyzed. First we investigate the case where the tracker is required to know the exact location of the target at each time step. We then relax our requirements and explore the case where the tracker may lose track of the target at a given time step, but it is able to "catch-up," regaining up-to-date information about the target's track. Finally, we consider the case where tracking information is only known after a delay of d time steps. We provide necessary and sufficient conditions on the number of queries per time step to track in the above three scenarios. These conditions are stated in terms of the entropy rate of the target's Markov chain.

2.1 Introduction

The problem of searching by asking questions has been the subject of extensive research for many years. Its origins can be traced back to Ulam and Rényi, who introduced variations of the famous "twenty questions problem." Since then, several other formulations of this two-person game have been considered in the literature [70, 21, 34, 1]. In this work, our goal is to find theoretical bounds on the number of queries per time step a tracker is required to ask a set of sensors to track a target.

The problem of target tracking through limited querying is of particular interest in the sensor network setting. Sensor networks have emerged as one of the most promising technologies in recent years. For the majority of existing sensor networks, sensors are small and inexpensive devices, which impose serious energy constraints affecting the network lifetime by having to transmit sensing information (over possibly long communication channels) to a remote monitoring station (tracker). Moreover, the reliability and the capacity of the channel available for communication with the tracker lead to restrictions on the amount of data sent over such networks. As a consequence, the tracker needs to make judicious decisions when selecting sensors to send data, so that communication with the sensor network is kept to a minimum. It is within this setting that we propose a sensor model in which sensors are capable of sending only one-bit messages to a tracker. These messages are used to gather tracking information about a moving target. Specifically, we provide necessary and sufficient conditions on the number of queries per time step to track a target in three different scenarios: (1) the tracker is required to know the exact location of the target at each time step; (2) the tracker may lose track of the target at a given time step, but it is able to "catch-up", regaining up-to-date information about the target's track; and (3) tracking information is only known by the tracker after a delay of d time steps.

The remainder of this chapter is organized as follows. Section 2.2 formalizes the tracking problem under three different definitions. According to each of these definitions, in Sections 2.3, 2.4 and 2.5 we state and prove theorems that relate the number of queries per time step necessary (and sufficient) to track and the entropy rate of the target's Markov chain. Finally, Section 2.6 concludes this chapter with summary remarks. This chapter is joint work with Hua Li.

2.2 Problem Formulation

We shall now formalize the problem of tracking a target that moves according to a Markov chain. The structure of the motion model is described by a directed graph $G = (\mathcal{X}, E)$, where the set of nodes $\mathcal{X}$, with finite cardinality, represents target locations, and the set of edges E describes each neighborhood, that is, possible target motion. If there exists an edge $(i, j) \in E$ $(i, j \in \mathcal{X})$, with associated transition probability $p_{ij} > 0$, connecting node $i \in \mathcal{X}$ to $j \in \mathcal{X}$, then the target moves from node i to j with probability p_{ij}. Associated with each target location is a sensor, which can sense if the target is at its location. Throughout this chapter we use the terms network node, sensor, and target location as synonyms.

A discrete-time, homogeneous and ergodic Markov chain $\{X_t : t \in \mathbb{N}\}$ on a probability space, with measure P and state space $\mathcal{X}$, models the random motion in time of a target. Each node identifies a state of the chain, and the target mobility law is specified by the (one-step) transition matrix $[p_{ij}]$, where $p_{ij} = \mathrm{P}\{X_{t+1} = j | X_t = i\}$, $i, j \in \mathcal{X}$.

Given the initial target location $X_1 = x_1 \in \mathcal{X}$, the history of Markov chain states visited by a target up to time step t is called a track.

Definition 2.2.1. The target track at time $t \geq 1$ is defined as the following finite random sequence of states:

$$X_{[1:t]} = X_1 X_2 \ldots X_t. \tag{2.2.1}$$

The main goal of target tracking is to estimate tracks over time. We use the following notation for such estimates: at time step $\tau \geq t$, the estimate of $X_{[1:t]}$ is denoted by $\hat{X}^{\tau}_{[1:t]}$; likewise, at time step $\tau \geq t$, the estimate of state X_t is given by $\hat{X}^{\tau}_t$.

A tracker estimates target tracks by querying subsets of sensors. At each time step t, the tracker may query the sensors a number of times, say K_t times. We denote the kth query at time t by $q_{t,k}$, $1 \leq k \leq K_t$. Furthermore, each query consists of a number of "questions", each of which addresses a particular sensor with a timestamp. Therefore, the query $q_{t,k}$ is characterized by a set of sensor-timestamp pairs:

$$q_{t,k} = \left\{ \left(s^j_{t,k}, \tau^j_{t,k}\right) : 1 \leq j \leq J_{t,k}, s^j_{t,k} \in \mathcal{X}, \tau^j_{t,k} \leq t \right\}, \tag{2.2.2}$$

where the sensor-timestamp pair $\left(s^j_{t,k}, \tau^j_{t,k}\right)$ denotes the question: "has sensor $s^j_{t,k}$ detected the target at time $\tau^j_{t,k}$?", and $J_{t,k}$ denotes the number of "questions" in the query $q_{t,k}$.

In response, the queried set of sensors transmit a single bit to the tracker. Specifically, the response $r_{t,k}$ to query $q_{t,k}$ can be written as

$$r_{t,k} = \begin{cases} 1, & \text{if } X_{\tau^j_{t,k}} = s^j_{t,k} \text{ for some } j; \\ 0, & \text{otherwise,} \end{cases} \tag{2.2.3}$$

that is, $r_{t,k} = 1$ if and only if, for some $1 \leq j \leq J_{t,k}$, sensor $s^j_{t,k}$ has detected the target at time $\tau^j_{t,k}$.

At every time step t, a querying scheme is modeled by a finite binary decision tree T_t. Here we use the same notion of decision tree as used by Rivest et al. in [74]. An internal node in T_t corresponds to queries $q_{t,k}$, $1 \leq k \leq K_t$. The right and left children of each internal node represent queries following a "yes" (1) and "no" (0) response, respectively (unless they are leaves). Note that the root in T_t is considered an internal node. Finally, associated with each leaf in T_t is a track estimate up to time step t, i.e., $\hat{X}^t_{[1:t]}$. With each leaf in T_τ we also associate a function $f(\tau)$, whose output is the time step t ($1 \leq t \leq \tau$) indicating certainty of correct estimation: $\hat{X}^\tau_{[1:t]} = X_{[1:t]}$. Since we assume prior knowledge of the initial target location, for any time step t, $f(t) \geq 0$ trivially. However, if the tracker is certain that $\hat{X}^t_{[1:t_1]} = X_{[1:t_1]}$ and $\hat{X}^t_{[1:t_2]} = X_{[1:t_2]}$, with $t_2 > t_1$, then $f(t) \geq t_2$; that is, $f(t)$ outputs the most recent time step at which the tracker is certain of being correct.

To summarize, at each time t, sensors are queried by the tracker according to a querying scheme modeled by a binary decision tree T_t, resulting in a sequence of queries $(q_{t,1}, q_{t,2}, \ldots, q_{t,K_t})$ and associated sequence of responses $(r_{t,1}, r_{t,2}, \ldots, r_{t,K_t})$, following a path in the decision tree, culminating with the output of an estimate of the track up to time t, $\hat{X}^t_{[1:t]}$, and an indication of what sub-track of this track is certain.

Let C denote the maximum number of queries allowed at each time step, called the query quota. Note that C is an integer. One could think of several scenarios where the number of queries at each time step is naturally limited. First, the capacity of the communication channel between the tracker and the sensors is usually scarce. Also, given the limited processing capabilities of small and simple sensors, a constraint on how fast queries can be processed is expected. Moreover, if sensors are deployed on a hostile enemy environment, it is reasonable to limit the maximum number of responses sent by sensors at each time step to avoid being detected. Therefore, for every time step t, T_t has at most $m = C + 1$ levels. Equivalently, we say the height of T_t, denoted by $h(T_t)$, satisfies $h(T_t) \leq C + 1$. Figure 2.1 illustrates a binary decision tree with four levels.

We further make the following remarks. All sensors are fault-free, have memory, and are able to communicate without error or delay with a tracker, which knows the initial location of the target and its mobility law. Sensor ranges do not overlap. Logarithms are taken to the base 2 and, by convention, $\log 0 = 0$ and $\log \frac{0}{0} = 0$.

In order to track a target, we use a strategy $\mathcal{S}$ defined as follows:

Definition 2.2.2. A strategy $\mathcal{S}$ is a rule that, at each time step, maps the current querying scheme and its results to the next querying scheme. In other words, at each time step t, $\mathcal{S}$ takes T_t and the sequence of responses at time t, and generates T_{t+1}.

Figure 2.2 depicts the mapping defined above. Note that Definition 2.2.2 defines a memoryless strategy in the sense that it only uses information from the previous time step, i.e., the querying scheme used to track a target at time $t + 1$ depends

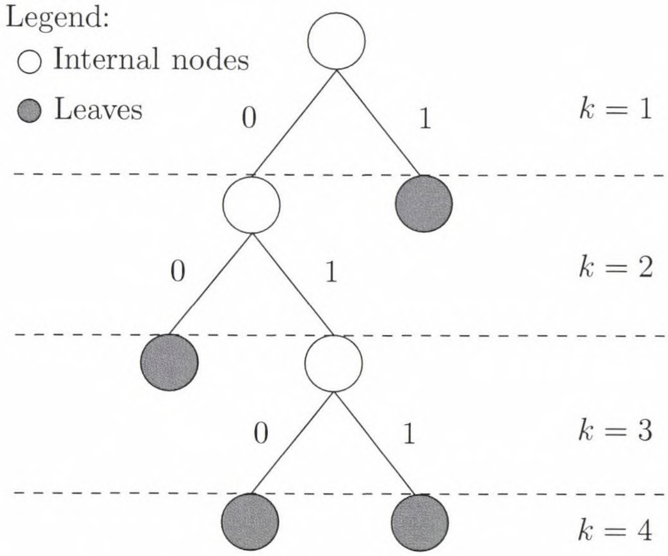

Figure 2.1: Binary decision tree T_t with $k = 4$ levels.

only on the querying scheme used at time t.

We investigate three distinct "degrees" of target tracking, which we call following, tracking, and d-tracking defined below. Whether or not these degrees hold depend on the instance of the target motion model, given by the tuple $(G, [p_{ij}], x_1)$.

Definition 2.2.3. A target is followable if there exists a strategy $\mathcal{S}$ such that, at each time step t, the target track estimate equals the true target track with probability one, that is,

$$\forall t \geq 1,\ \hat{X}^{t}_{[1:t]} = X_{[1:t]} \text{ and } f(t) = t \ a.s. \tag{2.2.4}$$

Definition 2.2.4. A target is trackable if there exists a strategy $\mathcal{S}$ such that the target track estimate equals the true target track infinitely often with probability one, that is,

$$\forall t \geq 1,\ \exists \tau > t \text{ s.t. } \hat{X}^{\tau}_{[1:\tau]} = X_{[1:\tau]} \text{ and } f(\tau) = \tau \ a.s. \tag{2.2.5}$$

In other words, even if the tracker loses track of the target, if it is able to "catch-up" and regain current information about the target track at a later time step, tracking is still accomplished. On the other hand, if the target track is estimated with certainty only after a fixed delay of d time steps, we define:

Definition 2.2.5. A target is d-trackable if there exists a strategy $\mathcal{S}$ such that the target track estimate equals the true target track, after a delay of d time steps, infinitely often with probability one, that is,

$$\forall t \geq 1,\ \exists \tau > t \text{ s.t. } \hat{X}^{\tau+d}_{[1:\tau]} = X_{[1:\tau]} \text{ and } f(\tau + d) = \tau \ a.s. \tag{2.2.6}$$

We are interested in answering the following questions:

1. What is the minimum query quota such that a target is followable?

2. What is the minimum query quota such that a target is trackable?

3. What is the minimum query quota such that a target is d-trackable?

As we shall see in the following sections, there is an intimate connection between querying and coding. A binary sequential source code naturally arises from the use of binary decision trees to represent the querying scheme. Recently, Borkar et al. [12] revisited the problem of sequential source coding, formulated as a constrained optimization problem on a convex set of probability measures. Although we also impose a causality constraint on codewords, one key difference between our work and [12] is that we introduce a constraint at each time step (the query quota) on the number of bits in a codeword. In the sections that follow, we state and prove three theorems on the conditions under which a target is followable, trackable, and d-trackable. We use the notation E_i to denote the set of neighbors of state $i \in \mathcal{X}$, that is, $E_i = \{j \in \mathcal{X} : p_{ij} > 0\}$.

2.3 Target Following

Theorem 2.3.1. *A target is followable if and only if*

$$C \geq \max_{i \in \mathcal{X}} \log |E_i|. \tag{2.3.1}$$

Proof. We first prove necessity. Let $X_1 = x_1 \in \mathcal{X}$. For all $t > 1$, assume

$$\mathrm{P}\left\{\hat{X}^t_{[0:t]} = X_{[1:t]}\right\} = 1, \tag{2.3.2}$$

and

$$C < \max_{i \in \mathcal{X}} \log |E_i|. \tag{2.3.3}$$

Hence, there exists a state $i \in \mathcal{X}$ such that $C < \log |E_i|$. Since the Markov chain $\{X_t : t \geq 1\}$ is ergodic, thus irreducible, there exists a time step $t > 1$ such that the t-step transition probability from state x_1 to state i is strictly positive, i.e., $p_{x_1 i}^{(t)} > 0$ (where $p_{x_1 i}^{(1)} = p_{x_1 i}$).

From the definition of query quota, we know that at most C bits per time step can be transmitted to the tracker. Therefore, the number of choices for estimating X_{t+1} must be at most 2^C at each time step. But $2^C < |E_i|$, and hence no strategy is able to identify all possible choices for X_{t+1}. Therefore,

$$\mathrm{P}\left\{\hat{X}_{[1:t+1]}^{t+1} \neq X_{[1:t+1]} \middle| X_t = i\right\} \geq \min_{j \in \mathcal{X}} p_{ij} > 0, \tag{2.3.4}$$

a contradiction.

To prove sufficiency, we show by induction that the simple and well-known binary search [43] yields a strategy $\mathcal{S}$ using which we can follow a target when

$$C \geq \max_{i \in \mathcal{X}} \log |E_i|. \tag{2.3.5}$$

Since the initial location of the target is known a priori, $\mathrm{P}\left\{\hat{X}_1^1 = X_1\right\} = 1$ trivially. For a fixed $t > 1$, assume $\mathrm{P}\left\{\hat{X}_{[1:t]}^t = X_{[1:t]}\right\} = 1$. It suffices to show that $\mathrm{P}\left\{\hat{X}_{[1:t+1]}^{t+1} = X_{[1:t+1]}\right\} = 1$.

The binary decision tree T_{t+1} is generated as follows. An internal node at level k in T_{t+1} denotes a query of the form:

$$q_{t+1,k} = \{(s_1, t+1), (s_2, t+1), \ldots, (s_n, t+1)\}, \tag{2.3.6}$$

where

$$|q_{t+1,1}| \leq \left\lceil \frac{|E_{x_t}|}{2} \right\rceil. \tag{2.3.7}$$

The set of sensors to be queried is repeatedly reduced by about half until the target track is estimated with certainty. Hence, either

$$|q_{t+1,k}| = \left\lceil \frac{|q_{t+1,1}|}{2^{k-1}} \right\rceil \quad \text{or} \quad |q_{t+1,k}| = \left\lfloor \frac{|q_{t+1,k}|}{2^{k-1}} \right\rfloor. \tag{2.3.8}$$

Since we are interested in showing sufficiency, we consider the case where the largest number of queries is required, that is, the case where $|q_{t+1,k}| = \left\lceil \frac{|q_{t+1,1}|}{2^{k-1}} \right\rceil$. By definition, T_{t+1} has at most $k = C + 1$ levels; thus, assuming the largest possible number of queries is required at time step $t + 1$, we can write

$$|q_{t+1,C}| \geq \frac{|q_{t+1,1}|}{2^{C-1}}. \tag{2.3.9}$$

When using binary search, the last query asked is a singleton, therefore $|q_{t+1,C}| = 1$, and we have

$$1 \geq \frac{\frac{|E_{x_t}|}{2}}{2^{C-1}} \Rightarrow C \geq \log |E_{x_t}|, \tag{2.3.10}$$

and hence we are able to estimate the value of X_{t+1} with certainty among all possible 2^C choices, that is, $\mathrm{P}\left\{\hat{X}^{t+1}_{[1:t+1]} = X_{[1:t+1]}\right\} = 1$. Thus, it suffices to have $C \geq \max_{i \in \mathcal{X}} \log |E_i|$ to follow a target. $\square$

2.4 Target Tracking

As described in Section 2.2, the target motion is modeled by an ergodic Markov chain with finite state space. Thus, the chain is also positive recurrent [15, 37], and its Shannon entropy rate can be calculated as [19, 93]:

$$H = -\sum_{i \in \mathcal{X}} \pi_i \sum_{j \in \mathcal{X}} p_{ij} \log p_{ij}, \tag{2.4.1}$$

where π_i is the stationary distribution of the Markov chain. In Equation (2.4.1), H represents the mean description complexity of the Markov chain, that is, the mean number of queries required per time step such that the monitoring station can correctly estimate the target's track. We shall now prove the following theorem:

Theorem 2.4.1.

a) If a target is trackable, then $C \geq H$.

b) A target is trackable if $C \geq H + 1$.

Proof. We prove part (a) using the concept of strong typicality [93] applied to the finite-state ergodic Markov chain $\{X_t : t \geq 1\}$. For every $t > 1$ and each state transition $(i, j) \in E$, we define the counting function N_{ij}^t on $X_{[1:t]}$ as

$$N_{ij}^t\left(x_{[1:t]}\right) = \sum_{k=0}^{t-1} \mathbb{1}_i\left(x_k\right) \mathbb{1}_j\left(x_{k+1}\right), \tag{2.4.2}$$

where $\mathbb{1}_i\left(x_k\right)$ denotes the indicator function, that is,

$$\mathbb{1}_i\left(x_k\right) = \begin{cases} 1, & \text{if } x_k = i; \\ 0, & \text{otherwise.} \end{cases} \tag{2.4.3}$$

For a fixed $\delta > 0$ and every $t > 0$, the set

$$\Delta_{t,\delta} = \left\{ x_{[1:t]} : \left| \frac{N_{ij}^t}{t} - \pi_i p_{ij} \right| < \delta, \quad \forall (i,j) \in E \right\} \tag{2.4.4}$$

is called strong typical set, and sequences in this set are called strong typical sequences. Lemma 2.4.2 below states two important properties of strong typical sets and strong typical sequences. Property (a) is shown in [93], and property (b) can be easily shown using a result from large deviation theory [44]. For completeness, we reproduce their proofs below.

Lemma 2.4.2. *For a fixed $\delta > 0$, we have:*

a) The probability of every strong typical sequence $x_{[0:t]}$ satisfies

$$2^{-t(H+c_1\delta)}p_{x_1} < \mathrm{P}\left\{X_{[1:t]} = x_{[1:t]}\right\} < 2^{-t(H-c_1\delta)}p_{x_1}, \tag{2.4.5}$$

where $c_1 = -\sum_{(i,j)\in E} \log p_{ij}$ and p_{x_1} is the initial distribution $P\{X_1 = x_1\}$.

b) For sufficiently large t,

$$\mathrm{P}\left\{x_{[1:t]} \notin \Delta_{t,\delta}\right\} \leq c_2 2^{-c_3 t}, \tag{2.4.6}$$

where c_2 and c_3 are positive constants.

Proof. To show property (a) in Lemma 2.4.2, we first bound $-\log \mathrm{P}\left\{X_{[1:t]} = x_{[1:t]}\right\}$ from above:

$$\begin{aligned}
-\log \mathrm{P}\left\{X_{[1:t]} = x_{[1:t]}\right\} &= \sum_{k=1}^{t-1} -\log p\left(x_{k+1}|x_k\right) - \log p_{x_1} \\
&= \sum_{i,j} N_{ij}\left(x_{[1:t]}\right)\left(-\log p_{ij}\right) - \log p_{x_1} \\
&< t\sum_{i,j}\left(\pi_i p_{ij} + \delta\right)\left(-\log p_{ij}\right) - \log p_{x_1} \\
&= t\Bigg[\sum_{i,j}(\pi_i p_{ij})(-\log p_{ij}) - \log p_{x_1} \\
&\quad + \sum_{i,j}\delta(-\log p_{ij})\Bigg] - \log p_{x_1} && (2.4.7) \\
&= t(H + c_1\delta) - \log_{p_{x_1}}, && (2.4.8)
\end{aligned}$$

where

$$c_1 = \sum_{(i,j)} - \log p_{ij}. \tag{2.4.9}$$

Similarly, we can bound

$$- \log \mathrm{P}\left\{X_{[1:t]} = x_{[1:t]}\right\} \tag{2.4.10}$$

from below:

$$t(H - c_1\delta) - \log p_{x_1} < - \log \mathrm{P}\left\{X_{[1:t]} = x_{[1:t]}\right\}. \tag{2.4.11}$$

Therefore, we have:

$$2^{-t(H+c_1\delta)}p_{x_1} < - \log \mathrm{P}\left\{X_{[1:t]} = x_{[1:t]}\right\} < 2^{-t(H-c_1\delta)}p_{x_1}. \tag{2.4.12}$$

We now show property (b) in Lemma 2.4.2 using the following result from large deviation theory:

Lemma 2.4.3. *[44] Suppose $\{X_t\}$ is an ergodic finite-state chain with state space $\mathcal{X}$ and let b_t denote its L^1 convergence parameter*

$$b_t = \sup_i \sup_j |p_{ij}^{(t)} - \pi_j|. \tag{2.4.13}$$

The series

$$b = \sum_{t>1} b_t \tag{2.4.14}$$

converges, and for any bounded function $F : \mathcal{X} \to \mathbb{R}$ and any $\delta > 0$, we have

$$\log \mathrm{P}\left\{\frac{\sum_{k=1}^{t} F(X_k)}{t} - \pi(F) \geq \delta\right\} \leq -\frac{t-1}{2}\left(\frac{\delta}{\alpha F} - \frac{3}{t-1}\right)^2, \tag{2.4.15}$$

as long as $t \geq 1 + 3b\bar{F}$, where

$$\bar{F} = \max_{x\in\mathcal{X}} |F(x)|, \tag{2.4.16}$$

and $\pi(F)$ *is the mean of the function* f *with respect to the stationary distribution* π *of the Markov chain, i.e.,*

$$\pi(F) = \sum_{i \in \mathcal{X}} F(i)\pi_i. \tag{2.4.17}$$

In order to apply the above lemma, we first construct an ergodic finite-state Markov chain $\{Y_t\}$ from the original Markov chain $\{X_t\}$ by taking the segment $Y_t = (X_{t-1}, X_t)$. Clearly, $\{Y_t\}$ is an ergodic Markov chain with finite state space $\mathcal{Y} = \mathcal{X}^2$. It is easy to verify that the stationary distribution λ_{ij} of $\{Y_t\}$ equals to $\pi_i p_{ij}$ for each edge $(i, j) \in \mathcal{X}^2$. Let b_t be the L^1 convergence parameter of $\{Y_t\}$ and $b = \sum_{t>1} b_t$. We take

$$F(Y_t) = \mathbb{1}_{ij}(Y_t), \tag{2.4.18}$$

so that F is bounded and

$$\bar{F} = \sup_j |F(j)| \leq 1. \tag{2.4.19}$$

Using Lemma 2.4.3,

$$\begin{aligned} \log \mathrm{P}\left\{\frac{\sum_{k=1}^{t} \mathbb{1}_{ij}(Y_t)}{t} - \lambda_{ij} \geq \delta\right\} &\leq -\frac{t-1}{2}\left(\frac{\delta}{b} - \frac{3}{t-1}\right)^2 \\ &\leq -\frac{t-1}{2}\left(\frac{\delta}{b} - 3\right)^2, \end{aligned} \tag{2.4.20}$$

for all $(i, j) \in \mathcal{X}^2$ and $t \geq 1 + \frac{3b}{\delta}$. Hence, for t sufficiently large,

$$\log \mathrm{P}\left\{\frac{N_{ij}}{t} - \pi_i p_{ij} \geq \delta\right\} \leq -\frac{t-1}{2}\left(\frac{\delta}{b} - 3\right)^2. \tag{2.4.21}$$

Similarly, applying Lemma 2.4.3 to function $F' = 1 - F$,

$$\log \mathrm{P}\left\{\frac{N_{ij}\left(X_{[1:t]}\right)}{t} - \pi_i p_{ij} \leq -\delta\right\} \leq -\frac{t-1}{2}\left(\frac{\delta}{b} - 3\right)^2, \tag{2.4.22}$$

for t sufficiently large. Combining inequalities (2.4.21) and (2.4.22), we have

$$\log \mathrm{P}\left\{\left|\frac{N_{ij}\left(X_{[1:t]}\right)}{t}-\pi_i p_{ij}\right| \geq \delta\right\} \leq -\frac{t-1}{2}\left(\frac{\delta}{b}-3\right)^2, \qquad (2.4.23)$$

for t sufficiently large.

We can now bound the complement of the strong typical set $\Delta_{t,\delta}$ as follows:

$$\begin{aligned} \mathrm{P}\left\{x_{[1:t]} \notin \Delta_{t,\delta}\right\} &= \mathrm{P}\left\{\bigcup_{(i,j)\in\mathcal{X}^2}\left\{\left|\frac{N_{ij}\left(X_{[1:t]}\right)}{t}-\pi_i p_{ij}\right| \geq \delta\right\}\right\} \\ &\leq \sum_{(i,j)\in\mathcal{X}^2} \mathrm{P}\left\{\frac{N_{ij}\left(X_{[1:t]}\right)}{t}-\pi_i p_{ij} \geq \delta\right\} \\ &\leq |\mathcal{X}|^2\, 2^{-(t-1)\left(\frac{\delta}{b}-3\right)^2} \\ &\leq c_2 2^{c_3(t-1)}, \end{aligned} \qquad (2.4.24)$$

for t sufficiently large, where

$$c_2 = |\mathcal{X}|^2 \qquad (2.4.25)$$

and

$$c_3 = \left(\frac{\delta}{b}-3\right)^2. \qquad (2.4.26)$$

□

Now, assume there exists a strategy $\mathcal{S}$ such that a target is trackable and $C < H$. Using Lemma 2.4.2, we get the following bound for the probability of the event $\left\{\hat{X}^t_{[1:t]} = X_{[1:t]}\right\}$:

$$
\begin{aligned}
\mathrm{P}\left\{\hat{X}^t_{[1:t]} = X_{[1:t]}\right\} &= \sum_{x_{[1:t]} \in \mathcal{X}_{[1:t]}} \mathrm{P}\left\{\hat{X}^t_{[1:t]} = x_{[1:t]} | X_{[1:t]} = x_{[1:t]}\right\} \mathrm{P}\left\{X_{[1:t]} = x_{[1:t]}\right\} \\
&= \sum_{x_{[1:t]} \in \Delta_{t,\delta}} \mathrm{P}\left\{\hat{X}^t_{[1:t]} = x_{[1:t]} | X_{[1:t]} = x_{[1:t]}\right\} \mathrm{P}\left\{X_{[1:t]} = x_{[1:t]}\right\} \\
&\quad + \sum_{x_{[1:t]} \notin \Delta_{t,\delta}} \mathrm{P}\left\{\hat{X}^t_{[1:t]} = x_{[1:t]} | X_{[1:t]} = x_{[1:t]}\right\} \mathrm{P}\left\{X_{[1:t]} = x_{[1:t]}\right\} \\
&\leq \sum_{x_{[1:t]} \in \Delta_{t,\delta}} \mathrm{P}\left\{\hat{X}^t_{[1:t]} = x_{[1:t]} | X_{[1:t]} = x_{[1:t]}\right\} \mathrm{P}\left\{X_{[1:t]} = x_{[1:t]}\right\} \\
&\quad + c_2 2^{-c_3 t} \\
&< \sum_{x_{[1:t]} \in \Delta_{t,\delta}} \mathrm{P}\left\{\hat{X}^t_{[1:t]} = x_{[1:t]} | X_{[1:t]} = x_{[1:t]}\right\} 2^{-t(H - c_1 \delta)} \\
&\quad + c_2 2^{-c_3 t}, \qquad (2.4.27)
\end{aligned}
$$

Given the constraint on the height of the tree T_t for each time step $t \geq 1$, there are at most 2^{tC} choices for $\hat{X}^t_{[1:t]}$. Hence,

$$
\sum_{x_{[1:t]} \in \Delta_{t,\delta}} \mathrm{P}\left\{\hat{X}^t_{[1:t]} = x_{[1:t]} | X_{[1:t]} = x_{[1:t]}\right\} \leq 2^{tC}. \qquad (2.4.28)
$$

Therefore,

$$
\mathrm{P}\left\{\hat{X}^t_{[1:t]} = X_{[1:t]}\right\} < 2^{-t[H - C - c_1 \delta]} + c_2 2^{-c_3 t}. \qquad (2.4.29)
$$

Since we can always choose $\delta > 0$ such that $H - C - c_1\delta > 0$, and $c_3 > 0$, it is clear that

$$
\sum_{t \geq 1} \mathrm{P}\left\{\hat{X}^t_{[1:t]} = X_{[1:t]}\right\} < \infty, \qquad (2.4.30)
$$

By the first Borel-Cantelli lemma [10], we have

$$
\mathrm{P}\left\{\hat{X}^t_{[1:t]} = X_{[1:t]} \quad \text{i.o.}\right\} = 0, \qquad (2.4.31)
$$

thus contradicting the assumption that a tracking strategy $\mathcal{S}$ catches-up infinitely often with probability one. Hence, it is necessary that $C \geq H$ for a target to be trackable.

To prove part (b), assume $C \geq H + 1$. We describe an interactive tracking strategy which we refer to as the catch-up strategy, and we show (again, using induction) that a target is trackable when $C \geq H + 1$. Let $t = 1$. Given the initial target location $X_1 = x_1 \in \mathcal{X}$, we first show that there exists $\tau > 1$ such that $\hat{X}^{\tau}_{[1:\tau]} = X_{[1:\tau]}$ and $f(\tau) = \tau$ almost surely.

In the catch-up strategy, each binary decision tree T_t is generated according to a scheme based on Huffman coding [19], where the tracker uses the Markov chain transition probabilities to get Huffman codewords. Traversing T_t (that is, querying) takes place as follows.

The first nodes to be queried are those associated with codewords whose leftmost bit is 1 (equivalently, one could first query nodes whose leftmost bit is 0). If the response to this query is 1, the tracker would then query nodes whose corresponding codewords first two bits (from left to right) are 1, and so forth. On the other hand, if the response to the first query is 0, the following nodes queried would then be those associated with codewords whose first bit (from left to right) is 0 and second bit is 1.

From Huffman coding, we can directly derive a binary decision tree T_{x_1} from E_{x_1}, the set of neighbors of state x_1. Each internal node in T_{x_1} represents a possible query, and each leaf corresponds to a possible target location at $t = 2$. We maintain an auxiliary tree T^*, initialized with T_{x_1}. In order to generate T_2, we use the fact that at most C queries can be asked at any given time step, and prune T^* at level

$m = \min(C+1, h(T^*))$. The resultant tree is T_2, the binary decision tree at $t = 2$. Each leaf in T_2 is associated with an estimate $\hat{X}^2_{[1:2]}$ and the certainty function $f(2)$. We now have two possibilities:

Case 1: If $f(2) = 1$, the track $X_{[1:2]}$ is known with certainty, and we have caught-up. The procedure described above is repeated for $t \geq 3$, that is, we first derive tree T_{x_2} from E_{x_2}, where $X_2 = x_2 \in \mathcal{X}$; we then set $T^* = T_{x_2}$, and generate T_3 with height $h(T_3) = \min(C+1, h(T^*))$. If $f(3) = 0$, we continue as in Case 2 below; otherwise, proceed as in Case 1.

Case 2: If $f(2) = 0$, the tracker does not have certainty about the estimate $\hat{X}^2_{[1:2]}$, which is assumed to be the maximum a posteriori (MAP) estimate of $X_{[1:2]}$, i.e.,

$$\hat{X}^2_2 = \arg\max_{i \in \mathcal{X}} \mathrm{P}\{X_2 = i | X_0\}. \tag{2.4.32}$$

In this case, at $t = 3$, we use Huffman coding to derive trees T_i, where $i \in \mathcal{X}$ is associated with leaves of the subtree in T^*, whose root corresponds to the chosen estimate in T_2 (that is, the last node traversed in T_2).

Each tree T_i is appended to T^*, replacing the leaf corresponding to location i in T^* by the root in T_i, thus yielding an updated T^*. We then prune this updated T^* so that $h(T_3) = \min(C+1, h(T^* - h(T_2)))$. In general, we prune T^* so that $h(T_t) = \min(C+1, h(T^* - h(T_{t-1})))$, for any $t \geq 1$. The root in T_3 corresponds to the internal node in T^* whose left and right subtrees are the appended T_i's. If $f(3) = 0$, there is no certainty about the estimate $\hat{X}^3_{[1:3]}$, and we derive T_4 the same way as T_3.

This process continues until $f(\tau) = \tau$, for some $\tau \geq 3$, when we would have caught-up. Querying is then resumed as in Case 1.

Note that there is no need to keep the complete auxiliary tree T^* as described in Case 2 above. At each time step $t \geq 1$, the root in T^* can be selected as the internal node whose subtrees are the appended T_i's, and all other nodes on upper levels can be discarded. However, this is an implementation issue which does not affect the proof. The example below illustrates the catch-up strategy.

Example 1. *Let* $X_1 = x_1 \in \mathcal{X}$, $C = 2$, *and assume* $X_2 = a_2 \in \mathcal{X}$. *Also, let* $E_{x_1} = \{a_1, a_2, a_3, a_4, a_5\}$ *with transition probabilities* $\{0.2, 0.1, 0.3, 0.2, 0.2\}$. *Thus, one possible set of Huffman codewords is* $\{10, 011, 00, 11, 010\}$, *and the binary decision tree* T_{x_1} *is illustrated in Figure 2.3.*

Since $t = 2$, T^* *is the same as* T_{x_1}. *Moreover, since* $C = 2$, T_2 *has three levels, as shown in Figure 2.4, where* $a_5 = \arg\max_{i \in \mathcal{X}} \mathrm{P}\{X_2 = i | X_1 = x_1\}$.

Given that $f(2) = 0$, *the estimate* $\hat{X}^2_{[1:2]} = x_1 a_5$ *is not known with certainty, thus* T_{a_2} *and* T_{a_5} *are generated using the set of neighbors* E_{a_2} *and* E_{a_5}, *respectively. Let* $E_{a_2} = \{a_1, a_3, a_4, a_5\}$ *with transition probabilities* $\{0.4, 0.1, 0.1, 0.4\}$, *and* $E_{a_5} = \{a_4, a_3\}$ *with transition probabilities* $\{0.5, 0.5\}$. T_{a_2} *and* T_{a_5} *are directly derived from Huffman coding and illustrated in Figure 2.5.*

Trees T_{a_2} *and* T_{a_5} *are appended to* T^* *as follows: the root in* T_{a_2} *takes place of* *the leaf corresponding to location a_2 in T^*; likewise, the root in T_{a_5} takes place of the leaf corresponding to location a_5 in T^*.*

The updated T^* is shown in Figure 2.6. At $t = 3$, T_3 is generated as follows: its root corresponds to the internal node in T^* whose left and right subtrees are T_{a_5} and T_{a_2}, respectively. The height of T_3 is $h(T_3) = \min(C+1, h(T^*) - h(T_2)) = 3$. Assuming $X_3 = a_5$, $\hat{X}^3_{[1:3]} = X_{[1:3]}$, and we have caught-up. This is shown in Figure 2.7.

We now show that the catch-up strategy can indeed be used to track a target. In other words, we show that the tracker regains current information about the target location infinitely often with probability one when the catch-up strategy is applied. Let $l_{[1:\tau]}$ be the average number of queries asked up to time τ, that is,

$$l_{[1:\tau]} = \frac{\sum_{k=1}^{\tau} l_k}{\tau}, \quad \tau > 1, \tag{2.4.33}$$

where l_k is the number of queries asked to estimate X_k with certainty. Clearly, l_k is a bounded function of the Markov chain $\{X_t : t \geq 1\}$, for every $k \geq 1$. By the Generalized Convergence Theorem for bounded functions of discrete-time and ergodic Markov chains with finite state space [62], we have

$$l_{[1:\tau]} \xrightarrow{a.s.} L, \tag{2.4.34}$$

where the limit L (according to the Source Coding Theorem [19, 93]) satisfies $H \leq L < H + 1$, and H is given by Equation (2.4.1). It suffices to show that

$$\mathrm{P}\{l_{[1:\tau]} < C, \quad \text{for some } \tau > 1\} = 1, \tag{2.4.35}$$

and we show it by contradiction. Assume

$$\mathrm{P}\{l_{[1:\tau]} \geq C, \quad \forall \tau > 1\} = p > 0. \tag{2.4.36}$$

Then, since $C \geq H + 1$ and $H \leq L < H + 1$,

$$\mathrm{P}\{l_{[1:\tau]} \geq L, \ \forall \tau > 1\} \geq \mathrm{P}\{l_{[1:\tau]} \geq C, \ \forall \tau > 1\} = p > 0. \tag{2.4.37}$$

Hence,

$$\mathrm{P}\left\{\lim_{\tau \to \infty} l_{[1:\tau]} = L\right\} \leq 1 - p, \tag{2.4.38}$$

a contradiction.

Therefore, there exists $\tau > 1$ such that $\hat{X}^{\tau}_{[1:\tau]} = X_{[1:\tau]}$ and $f(\tau) = \tau$ almost surely. By induction, now assume the above is true for $t = t'$, that is, there exists $\tau' > t'$ such that $\hat{X}^{\tau'}_{[1:\tau']} = X_{[1:\tau']}$ and $f(\tau') = \tau'$ almost surely.

Using the same reasoning as above (for the case $t = 1$), we can show that there exists $\tau'' > \tau'$ such that $\hat{X}^{\tau''}_{[1:\tau'']} = X_{[1:\tau'']}$ and $f(\tau'') = \tau''$ almost surely. Hence, it suffices to have $C \geq H + 1$ to track a target. □

2.5 Target d-tracking

Theorem 2.5.1.

a) If a target is d-trackable for some positive integer d, then $C \geq H$.

b) For any positive integer d, a target is d-trackable if $C \geq H + \frac{1}{d}$.

Proof. Part (a) is proven once again using contradiction and strong typicality. Similarly to the tracking case, we assume that $C < H$ and that there exists a strategy $\mathcal{S}$ such that a target is d-trackable. The track estimate $\hat{X}_{[1:t]}^{t+d}$ has at most $2^{(t+d)C}$ choices, since $\hat{X}_{[1:t]}^{t+d}$ can be true at most $2^{(t+d)C}$ times. Hence, the probability of the event $\left\{\hat{X}_{[1:t]}^{t+d} = X_{[1:t]}\right\}$ is bounded by

$$\mathrm{P}\left\{\hat{X}_{[1:t]}^{t+d} = X_{[1:t]}\right\} < 2^{-(t+d)[(H-C)-c_1\delta]} + c_2 2^{-c_3(t+d)}. \tag{2.5.1}$$

Again, by the first Borel-Cantelli lemma, we have

$$\mathrm{P}\left\{\hat{X}_{[1:t]}^{t+d} = X_{[1:t]} \quad \text{i.o.}\right\} = 0, \tag{2.5.2}$$

a contradiction. Thus, $C \geq H$.

We show part (b) using a block version of the catch-up strategy described in the proof of Theorem 2.4.1. Consider the sequence of random variables $\{W_n : n > 0\}$, where $W_n = (X_{d(n-1)+1}, \ldots, X_{dn})$, $d > 0$, that is, each random variable W_n is a segment of of length d of the sequence $\{X_t : t \geq 1\}$. We call the sequence $\{W_n : n > 0\}$ a block Markov chain taking values in the state space $\mathcal{X}^d$. Assuming $C \geq H + \frac{1}{d}$, and given the initial target location x_1, we skip querying during the first d time steps. For each time step t, from $t = d+1$ to $t = 2d$, we apply the catch-up strategy to get $\hat{X}^{2d}_{[1:d]}$ and $f(2d)$. This is done using the transition probabilities of the Markov chain $\{W_n : n > 0\}$ to generate Huffman codewords. Thus, at $t = 2d$, we have the estimate $\hat{W}_1 = \left(\hat{X}^{2d}_1, \hat{X}^{2d}_2, \ldots, \hat{X}^{2d}_d\right)$. This procedure is repeated for every "block" of d time steps, hence if C_W is the maximum number of queries allowed during each block, $C_W = dC$. Moreover, the entropy H_W of the Markov chain $\{W_n : n > 0\}$ can be calculated in terms of the entropy rate H of the original Markov chain as

$$\begin{aligned}
H_W &= \lim_{n\to\infty} \frac{-\log \mathrm{P}\left\{X_{[1:nd]} = x_{[1:nd]}\right\}}{n} \\
&= \lim_{n\to\infty} d \left\{ \frac{-\log \mathrm{P}\left\{X_{[1:nd]} = x_{[1:nd]}\right\}}{nd} \right\} \\
&= -d \sum_{i,j\in\mathcal{X}} \pi_i p_{ij} \log p_{ij}, \qquad (2.5.3)
\end{aligned}$$

that is, $H_W = dH$.

By Theorem 2.4.1, if $C_W \geq H_W + 1$, that is, if $C \geq H + \frac{1}{d}$ and $d > 0$, a target is d-trackable. □

2.6 Concluding Remarks

In this chapter, we have studied the number of queries required to follow, track, and d-track a target that moves according to a Markov chain. Necessary and sufficient conditions have been presented for all cases, as well as corresponding following, tracking, and d-tracking strategies. One possible area for future work is to extend these results to the multitarget scenario, as well as to consider the case where sensors are faulty (i.e., their query responses may be wrong), and where noise is present in the communication between sensors and tracker. In this direction, it would be natural to introduce the notion of distance between sensors (states) and analyze tracking performance under criteria such as the mean squared error. We conjecture that results related to rate-distortion theory are possible. Another interesting variation is to take sensors responses to be the number of sensors that reply "yes" to a query. Future work could also include investigating the mean number of time steps (in terms of number of queries) involved in the catch-up strategy before the target track can be estimated, i.e., the mean lag time. Although the simplicity of this strategy is particularly attractive, it is of interest to find the strategy that incurs the minimum lag time. Recently, Hua Li extended this work by considering the case when we have no a priori knowledge about the target motion model [52].

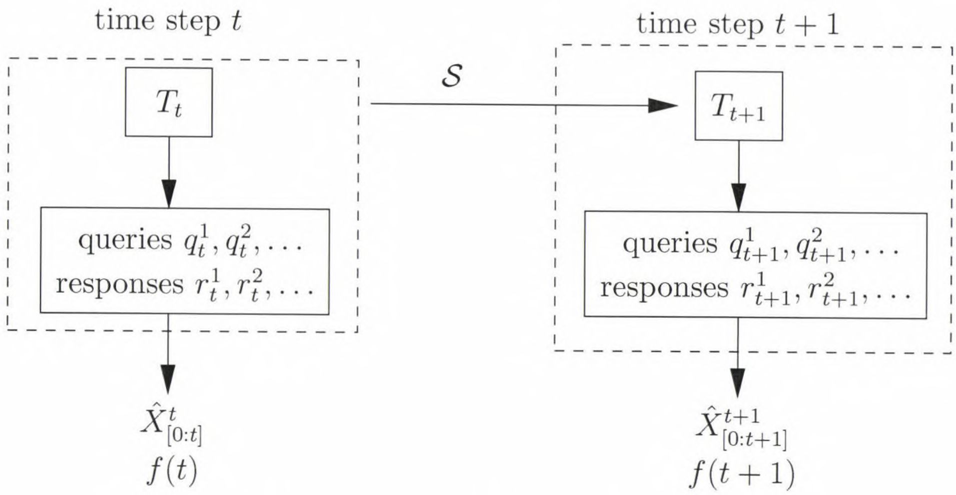

Figure 2.2: Strategy mapping.

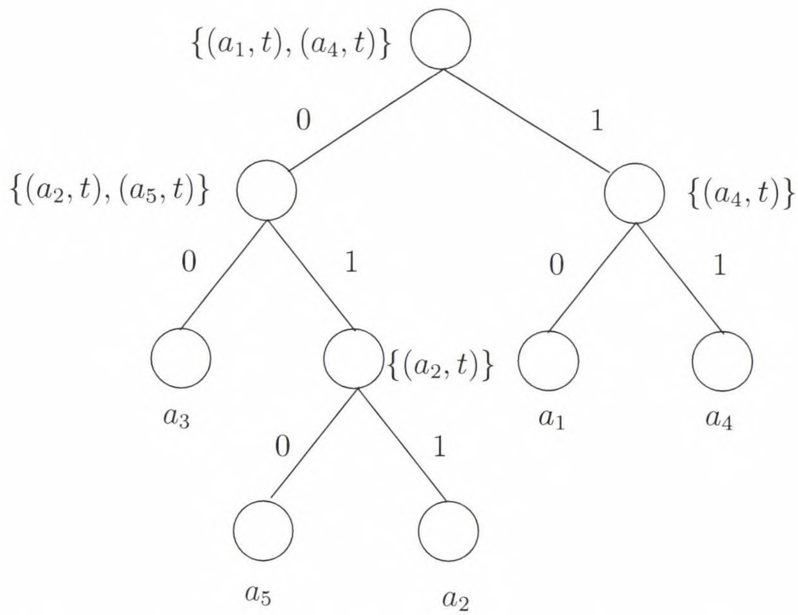

Figure 2.3: Binary decision tree T_{x_1} (equivalently, T^*) derived from Huffman coding. Each leaf correspond to a possible target location, and each internal node represents a possible query at any $t > 0$.

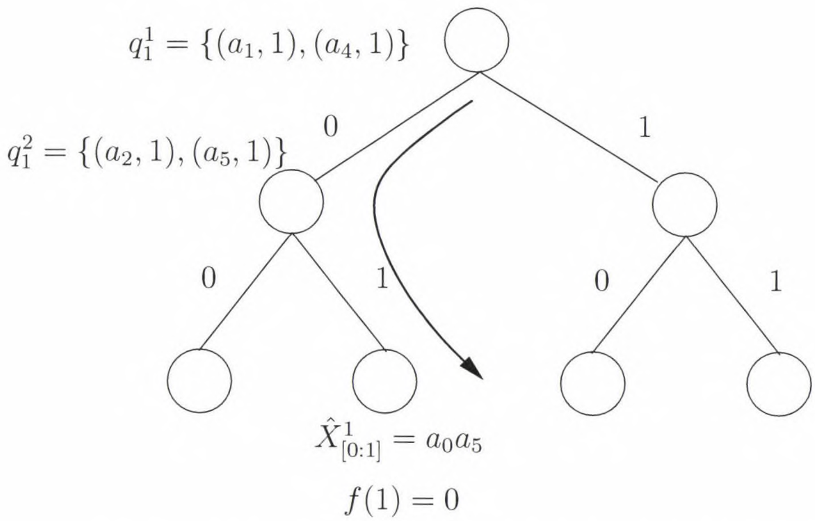

Figure 2.4: Binary decision tree T_2. The arrow shows the tree traversal at $t = 1$.

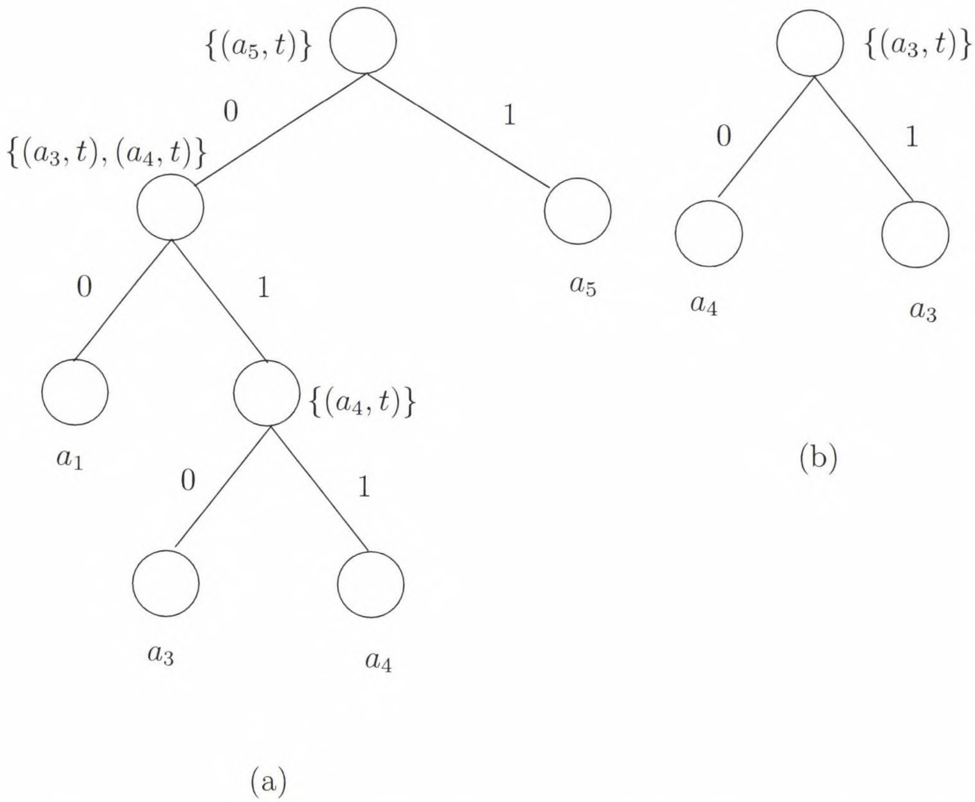

Figure 2.5: (a) Binary decision tree T_{a_2}. (b) Binary decision tree T_{a_5}.

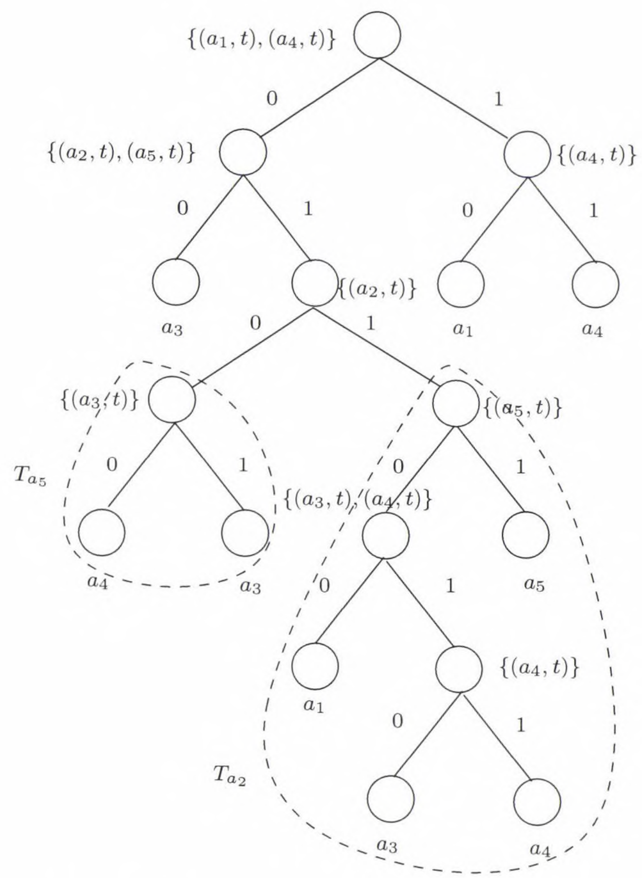

Figure 2.6: T^* updated with subtrees T_{a_5} and T_{a_2}.

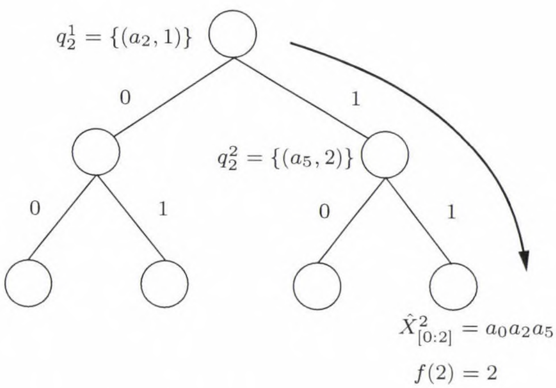

Figure 2.7: Binary decision tree T_3. The arrow shows the tree traversal at $t = 3$.

Chapter 3

Models for Active Sensing in Urban Terrain

We consider models for the various elements present in the urban environment required in the design of an active sensing platform for such scenarios. Specifically, we provide clutter, sensor, multipath, signal, and motion models. To the best of our knowledge, this work is the first to investigate how the interplay among these elements affect the overall performance of a closed-loop multitarget-multisensor tracking system.

3.1 Introduction

The past few years have shown that conventional warfare belongs in the past. The battles have now moved to dense urban environments, where surveillance and tracking systems are denied line-of-sight, thus creating the need for innovative systems that will give allied troops an advantage over enemy fighters. While the primary motivation behind this work is the urban battlefield presented to military forces, radiolocation applications in the transportation and communications industries (e.g., urban vehicular sensing platforms) would also benefit from effective solutions to the problem of target tracking in urban terrain. The interdisciplinary nature of this work highlights the challenges involved in designing a closed-loop active sensing platform for next-generation tracking and surveillance systems, as well as the importance of considering different diversity modes under unfavorable environmental conditions.

The first step in designing an active sensing platform for target tracking in an urban environment is to model the various elements that are part of this environment. Although intrinsically imperfect due to the many simplifying assumptions explained in this section, the models considered facilitate the analysis of the interplay among these different elements, and how they ultimately affect the overall tracking system performance.

The remainder of this chapter is organized as follows. In Section 3.2, clutter and multipath in urban terrain are discussed and the model of the specific scenario considered is presented. In Section 3.3, the transmitted and received signal models are explained. Target motion models are discussed in Section 3.4. Finally, Section 3.5 concludes this chapter.

3.2 Clutter and Multipath in Urban Terrain

The overwhelming complexity of the urban environment makes active sensing for ground target tracking a very interesting and challenging problem. Although we may have access to layouts of streets, buildings and vegetation via satellite radar imagery and city maps, the urban scenario is particularly difficult for a tracker due to the unpredictability and nonlinearity of target maneuvers, as well as the presence of arbitrary obscuration, that is, areas in the scenario that may not be visible to a sensor at any given instant of time, thus making targets "disappear" occasionally. Figure 3.1 is a pictorial representation of the urban terrain.

A number of terrain factors have major impact on the overall system performance. Different road classes (e.g., highways, arterial roads, residential streets, alleys) impose different constraints on ground vehicles (e.g., change in speed, stopping points, turns, exits). In addition, different construction materials (e.g., glass, concrete, brick, wood) have different reflectivity coefficients, and therefore different multipath conditions. Clearly, it is virtually impossible to consider every detail in every possible scenario. In this work, we consider a representative scenario that allows the tracker to experience the main technical challenges observed in practice: multipath ambiguities, lack of continuous target visibility, and measurement-to-track uncertainty due to clutter. The simulated scenario is depicted in Figure 3.2, which shows four building structures at an intersection. The uneven nature of urban clutter is represented by the '+', indicating vegetation on the center median and sidewalks. A radar transmitter, represented by '$\bigtriangledown$', is located at (2085,1470.5); whereas two radar receivers, each with three sensor array elements, are located at (2088,1475) and (2078,1467), both represented by '$\bigcirc$'.

Throughout this work, the term clutter is used to describe the signal received as a result of scattering from background objects other than targets of interest. Hence, a wide variety of background elements, from raindrops and birds to buildings and trees, are collectively described as clutter. Usually dense and unevenly distributed over the surveillance area, urban clutter increases the false alarm rate and missed detections when modeled inappropriately. At first glance, clutter may behave as

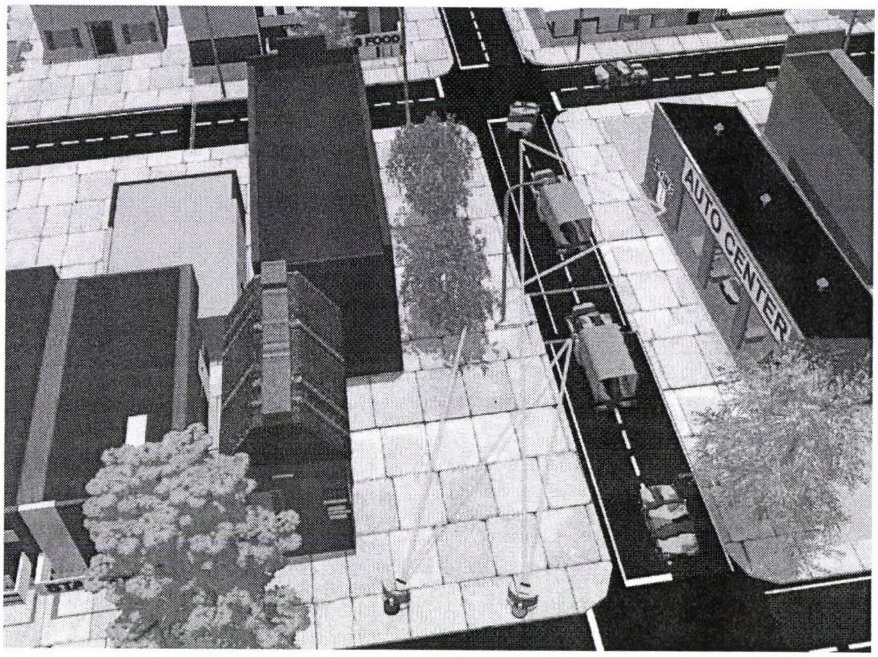

Figure 3.1: Pictorial representation of the urban terrain.

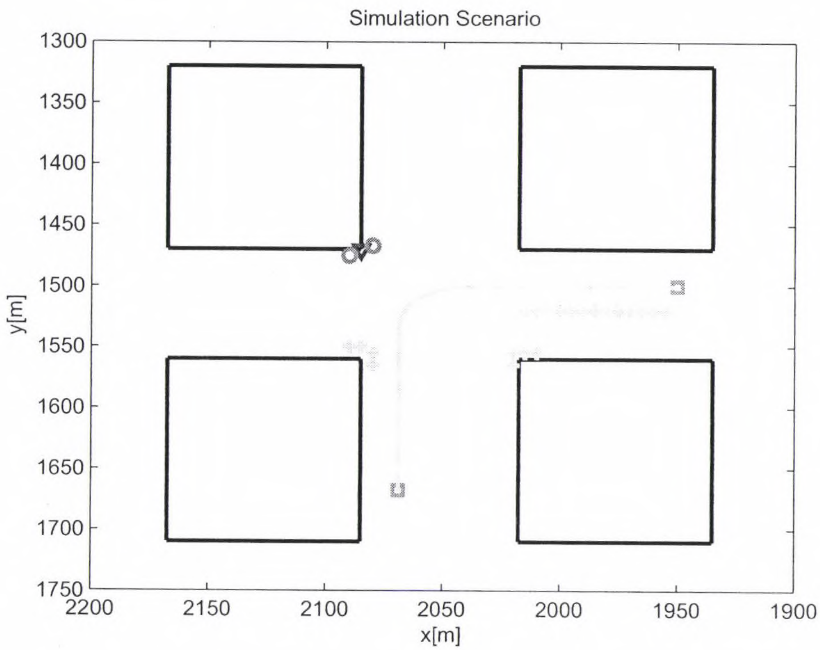

Figure 3.2: The simulated urban terrain. The start and end trajectory points are shown as □; receivers are shown as ◯; the transmitter is shown as ▽; and clutter discretes are shown as +.

noise, in the sense that it creates a background intensity above which the target return must rise to be detected. However, clutter, as opposed to noise, is caused by the transmitted signal, and therefore it is directly related to the signal reflected by targets. Hence, ideally, clutter should not be modeled as noise. Another widely used model is one that assumes clutter to be homogeneous over the sensor's surveillance area. However, variations in the underlying terrain, vegetation, and different urban structures contribute to homogeneity violations, making this clutter model far from ideal [30, 31, 61]. Figure 3.3 shows the radar return of an urban scene in which no target is present. Thus, this image represents the relative intensity of clutter only, where we can clearly distinguish bright, discrete, clumpy, non-Gaussian, and non-homogeneous scattering.

We consider clutter as a superposition of N_c independent scatterers,

$$n_c(t) = \sum_{i=1}^{N_c} a_i s\left(t - \tau_i\right) e^{2\pi j \nu_i}, \tag{3.2.1}$$

where the ith scatterer has reflectivity a_i, τ_i is the time-delay from the transmitter to the ith scatterer and back to the receiver, ν_i is the Doppler shift incurred during propagation, and $s(t)$ is the transmitted signal.

Target detection in urban terrain is affected by multipath propagation due to the inability of sensors to distinguish between the received signal scattered directly from a target and the received signal that traversed an indirect path in the urban scenario. Therefore, targets can also be detected due to reflections from buildings, vegetation, and other clutter scatterers having different reflectivity coefficients, thus presenting different multipath conditions.

Using prior knowledge of the terrain, a physical scattering model can be derived.

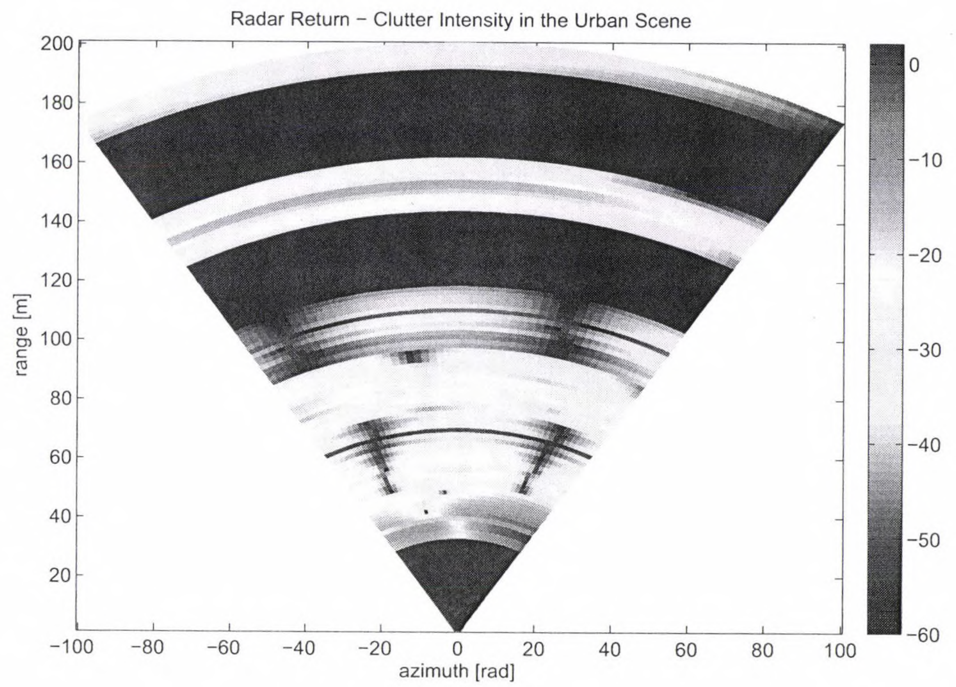

Figure 3.3: Urban clutter example.

We assume reflective surfaces are smooth, reflectivity coefficients are constant, and the angle of incidence equals the angle of reflection. We further assume that the strength of the radar return is negligible after three reflections. For an unobstructed target, the direct path can be described as follows. Let $\mathbf{p}$ and $\mathbf{q}$ be vectors corresponding to the paths from transmitter to target and from target to receiver, respectively. The length of the direct path is the sum of the lengths of $\mathbf{p}$ and $\mathbf{q}$; azimuth is the angle between the receiver and $\mathbf{q}$; and the Doppler shift is the sum of the projected target velocity onto $\mathbf{p}$ and $\mathbf{q}$. In all other cases, path length, azimuth and Doppler shift can be calculated once the reflection point on the clutter scatterer has been determined. For instance, let (x_c, y_c) be the incidence point of the transmitted signal on the clutter scatterer, (x_k, y_k) the target position at time step k, and (x_r, y_r) the receiver location. Using simple geometry and the line equation, the reflection point (x_c, y_c) can be found solving the equations below:

$$y_c = mx_c + c \tag{3.2.2}$$

$$\frac{[-m(x_r - x_c) + y_r - y_c]^2}{(x_r - x_c)^2 + (y_r - y_c)^2} = \frac{[-m(x_k - x_c) + y_k - y_c]^2}{(x_k - x_c)^2 + (y_k - y_c)^2}, \tag{3.2.3}$$

where m is the slope and c is the y-intercept in the line equation representing the clutter scatterer. Both m and c are assumed to be known. Note that such a point may not exist due to possible obscuration and the finite dimensions of scatterers. Equations above refer to the transmitter-target-clutter-receiver path, and the approach is analogous for other paths.

3.3 Signals for Active Sensing in Urban Terrain

Radar has become an essential sensor in tracking and surveillance systems due to its ability to survey wide areas rapidly under any weather conditions. In this work, we consider a sensing system where small low-power multistatic radars are distributed over the surveillance area. In particular, we consider bistatic radar pairs augmented by additional sensors (transmitters or receivers). The physical separation between transmitter and receiver in such system provide the spatial diversity needed to improve coverage, and thus detection. Although it is out of the scope of this work to delve into details of radar sensors, for the sake of completeness, we give a brief overview below.

A radar is called an active sensor because it initiates the energy reflected by objects in its surveillance area. This energy is then collected by the radar's aperture [11]. The active nature of a radar is the primary factor that makes it an effective sensor, since it gives the system designer some control over both transmitted and received signals. In other words, it is possible to adaptively schedule the waveform being sent according to changing environmental conditions. On the other hand, a disadvantage of active sensors is that if the sensor location is revealed to a target, this target can take evasive or hostile actions. This scenario is considered in Chapter 5.

Most current radar systems are monostatic, that is, transmitter and receiver are co-located. The performance of such systems has been greatly improved by the advent of high-resolution imaging, low-sidelobe antennas, and high-speed digital signal processing. However, since scattering occurs in all directions, a single receiver can only intercept a very small portion of this energy, and much of the signal is lost. Moreover, as targets become faster, more agile, and stealthier, the shortcomings of monostatic systems are accentuated.

The separation of transmitter and receiver in bistatic and multistatic radar systems can overcome these limitations and offer the potential to extend the capabilities and performance of current systems, improving coverage and detection. Although the inherent advantages of multistatic systems make them attractive for a variety of applications, they come with the cost of increased complexity and new challenges. While in monostatic radars synchronization between transmission and reception is straightforward, the separation of transmitter and receiver makes this task much more challenging, and synchronization has to be achieved via atomic clocks, GPS signals, or a signal sent directly from the designated transmitter. Another crucial issue is the data fusion scheme that must be used to combine measurements from various receivers. To date, there have been few good multistatic models and further research is needed [32] .

A survey of the literature reveals that definitions of bistatic and multistatic radars are quite widely varying with no universal acceptance of a single description. The IEEE defines bistatic radars as a radar system that uses sensors at different locations for transmission and reception. However, there is no stipulation as to how far apart the two sensors should be. Clearly, if they are near co-located, then the system approximates a monostatic radar. If a further sensor (either transmitting or receiving) is added to the bistatic pair, then this might be called a multistatic radar. However, other terminology often includes netted radars [4], multisite radars [16], distributed radars [90], and MIMO radars [27]. The distinctions between these are, at best, blurred. In this work we use the term multistatic to mean any system comprising a bistatic pair augmented by an additional sensor.

The mth receiver ($m = 1, \ldots, M$) is a uniform linear array of L_m sensor elements, separated by distance d_m, and where the direction of arrival of the signal sent by the nth transmitter ($n = 1, \ldots, N$) is $\theta_{n,m}$. We assume coherent processing, i.e., radar returns that arrive at different receiver sampling intervals can be processed jointly. In other words, we are assuming that radar returns can be stored, aligned, and subsequently fed to the receiver for fusion.

Before we describe transmitted and received signals, it is important to understand the different time frames involved. While transmitter and receiver perform signal processing on a intrapulse fashion, the tracker works on a interpulse time frame. Therefore, in the proposed signal model, three time scales are used: the state sampling period $T = t_k - t_{k-1}$, the pulse repetition interval T_1, and the receiver sampling period T_2. In general, $T_2 \ll T_1 \ll T$. In addition, we use the far-field assumption and consider the signal wave to be planar.

At time t_k, a series of pulses is transmitted at periods of T_1 seconds. Assuming Gaussian-windowed up-sweep and down-sweep chirp signals of unit energy, the signal transmitted by the nth transmitter at time t_k is given by:

$$s_{k,n}(t) = \sum_{b=-(B-1)/2}^{(B-1)/2} \frac{\exp\left\{\left[\pm j\gamma - 1/(2\kappa^2)\right](t - bT_1)^2\right\}}{(\pi\kappa^2 B^2)^{1/4}}, \tag{3.3.1}$$

where $t \in \mathbb{R}$, B is the number of pulses transmitted, κ represents the pulse duration, and γ is the chirp rate. The complex exponential is positive or negative according to the waveform scheduled for transmission: up-sweep chirp or down-sweep chirp, respectively.

The received signal is a summation of reflections from targets of interest and clutter scatterers. Let $P_{k,n,m}$ be the total number of reflections received by the mth receiver at time t_k, that originated from the nth transmitter. Signals from the pth path ($p = 1, \ldots, P_{k,n,m}$) are subject to a random phase shift $\phi^p_{n,m}$, uniformly distributed in $(-\pi, \pi]$. Hence, the signal received by the lth element ($l = 1, \ldots, L_m$) of the mth sensor array at time $t_k + uT_2$ can be written as:

$$\mathbf{y}_{k,m,l}(u) = \sum_{n=1}^{N} \sum_{p=1}^{P_{k,n,m}} e^{j\phi^p_{n,m}} \mathbf{g}^p_{n,m}\left(\mathbf{x}_k; uT_2\right) + \mathbf{e}(u), \tag{3.3.2}$$

where $u = (0, \ldots, U-1)$ is the sample index, and $\mathbf{e}(u)$ is a complex white Gaussian process. We can write $\mathbf{g}^p_{n,m}\left(\mathbf{x}_k; uT_2\right)$ as:

$$\begin{aligned} \mathbf{g}^p_{n,m}\left(\mathbf{x}_k; uT_2\right) = & \\ & \alpha^p_{n,m}\left(\mathbf{x}_k\right) \cdot s_{k,n}\left(uT_2 - \tau^p_{n,m}\left(\mathbf{x}_k\right)\right) \cdot \\ & e^{2\pi j \nu^p_{n,m}(\mathbf{x}_k) uT_2 - j(l_m - 1)\bar{d}_m\left[\cos(\theta^p_{n,m}(\mathbf{x}_k))\right]} \cdot \\ & e^{-2\pi j \nu^p_{n,m}(\mathbf{x}_k) uT_2 + j(l_m - 1)\bar{d}_m\left[\sin(\theta^p_{n,m}(\mathbf{x}_k))\dot{\theta}^p_{n,m}(\mathbf{x}_k) uT_2\right]}, \end{aligned} \tag{3.3.3}$$

where for the carrier signal wavelength λ, $\bar{d}_m = d_m/\lambda$, and $s_{k,n}\left(uT_2 - \tau^p_{n,m}\left(\mathbf{x}_k\right)\right)$ is the delayed replica of the transmitted signal given in Equation (3.3.1). In addition, the following received signal parameters are defined for the pth path between the nth transmitter and mth receiver: $\alpha^p_{n,m}\left(\mathbf{x}_k\right)$ is the magnitude of the radar return, including transmitted signal strength and path attenuation; $\tau^p_{n,m}\left(\mathbf{x}_k\right)$ is the time-delay incurred during propagation; $\nu^p_{n,m}\left(\mathbf{x}_k\right)$ represents the Doppler shift; and $\theta^p_{n,m}\left(\mathbf{x}_k\right)$ is the direction of arrival, where $\dot{\theta}^p_{n,m}\left(\mathbf{x}_k\right)$ is its rate of change. The parameters above can be computed for each target state $\mathbf{x}_k$, given prior knowledge of the urban scenario.

3.4 Models for Target Motion in Urban Terrain

One of the two major challenges in target tracking is the uncertainty about the target motion; the other being the uncertainty about measurement origin. The target motion uncertainty refers to the fact that an accurate dynamic model of the target(s) being tracked is not available to the tracker. To this end, various mathematical motion models have a been developed over the years. A comprehensive and updated survey, emphasizing the underlying ideas and assumptions of these models is given in [53, 54, 55]. While the tracking community traditionally has been concerned with modeling the motion of civilian and military aircrafts, there is little literature available about modeling the motion behavior of urban ground vehicles and dismounts (i.e., human motion) [26, 40]. Target motion in urban terrain can be described by a large number of models, mixed in various ways. It is not the objective of this work to design novel motion models for targets in urban terrain. Instead, we adopt existing models in the literature.

Before we proceed, a few observations should be made. First, we consider motion models of a "point target." Although a target is probably never really a point in the urban environment, for the purposes of this work, it suffices to treat each target as a point object without shape or any other spatial characteristic or feature. Second, although measurements are usually available only at discrete instants, the target motion is more accurately modeled in continuous time, since the target dynamics does not depend on how or when samples are taken. However, since this work relies on computer simulations, we adopt discrete-time models.

Let the target state vector at time t_k be

$$\mathbf{x}_k = \left[x_k, \dot{x}_k, y_k, \dot{y}_k, \ddot{x}_k, \ddot{y}_k\right]^\top, \tag{3.4.1}$$

where $\top$ denotes matrix transpose, $\left[\dot{x}_k, \dot{y}_k\right]^\top$ is the velocity vector, and $\left[\ddot{x}_k, \ddot{y}_k\right]^\top$ is the acceleration vector.

Motion models can be divided into two categories: uniform motion (or non-maneuvering) models and maneuvering models. The most commonly used non-maneuvering motion model is the nearly constant velocity (NCV) model, which can be written as:

$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{G}\mathbf{w}_k, \tag{3.4.2}$$

where the process noise $\mathbf{w}_k$ is a zero-mean white-noise sequence,

$$\mathbf{F} = \begin{pmatrix} 1 & T & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & T & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{G} = \begin{pmatrix} \frac{1}{2}T^2 & 0 \\ T & 0 \\ 0 & \frac{1}{2}T^2 \\ 0 & T \\ 0 & 0 \\ 0 & 0 \end{pmatrix}, \tag{3.4.3}$$

and T is the state sampling period. The process noise covariance multiplied by the gain is the design parameter

$$\mathbf{Q} = \text{diag}\left[\sigma^2_{w_x}\mathbf{Q}', \sigma^2_{w_y}\mathbf{Q}'\right], \tag{3.4.4}$$

where $\mathbf{Q}' = \mathbf{G}\mathbf{G}^\top$, and $\sigma^2_{w_x}$ and $\sigma^2_{w_y}$ are uncorrelated variances in x and y directions, respectively, corresponding to noisy "accelerations" that account for modeling errors. To achieve nearly constant velocity or uniform motion, changes in velocity over the sampling interval need to be small compared to the actual velocity, i.e., $\sigma^2_{w_x} T \ll \dot{x}_k$ and $\sigma^2_{w_y} T \ll \dot{y}_k$.

We consider two different models to describe accelerations and turns. Left and right turns are modeled by the coordinated turn (CT) model with known turn rate ω. This model assumes targets move with nearly constant velocity and nearly constant angular turn rate. Although ground target turns are not exactly coordinated turns, the CT model, originally designed for airborne targets, is a reasonable and sufficient approximation for our purposes. Knowledge of each turn rate is based on prior information about the urban scenario. For the six-dimensional state vector, the CT model follows Equation (3.4.2), where $\mathbf{w}_k$ is a zero-mean additive white Gaussian noise (AWGN) that models small trajectory perturbations, and

$$\mathbf{F} = \begin{pmatrix} 1 & \frac{\sin \omega T}{\omega} & 0 & -\frac{1-\cos \omega T}{\omega} & 0 & 0 \\ 0 & \cos \omega T & 0 & -\sin \omega T & 0 & 0 \\ 0 & \frac{1-\cos \omega T}{\omega} & 1 & \frac{\sin \omega T}{\omega} & 0 & 0 \\ 0 & \sin \omega T & 0 & \cos \omega T & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{3.4.5}$$

Similarly to the NCV model, $\mathbf{Q} = \sigma^2_w \mathrm{diag}\,[\mathbf{Q}', \mathbf{Q}']$, $\mathbf{Q}' = \mathbf{G}\mathbf{G}^\top$, and σ^2_w is the process noise variance. However, contrary to the NCV model, x and y directions are now coupled.

Accelerations and decelerations are described by the Wiener-sequence acceleration model, where

$$\mathbf{F} = \begin{pmatrix} 1 & T & 0 & 0 & \frac{1}{2}T^2 & 0 \\ 0 & 1 & 0 & 0 & T & 0 \\ 0 & 0 & 1 & T & 0 & \frac{1}{2}T^2 \\ 0 & 0 & 0 & 1 & 0 & T \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{3.4.6}$$

and

$$\mathbf{G} = \begin{pmatrix} \frac{1}{2}T^2 & 0 \\ T & 0 \\ 0 & \frac{1}{2}T^2 \\ 0 & T \\ 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{3.4.7}$$

For this model, the process noise $\mathbf{w}_k$ in Equation (3.4.2) is a zero-mean white-noise sequence with uncorrelated variances in the x and y directions. We consider $\sigma_{w_x}^2 T \ll \ddot{x}_k$ and $\sigma_{w_y}^2 T \ll \ddot{y}_k$, in which case the Wiener-sequence acceleration model is also known as the nearly constant acceleration (NCA) model.

3.5 Concluding Remarks

In this chapter, we described models for urban clutter, multipath, sensor signals, and target motion. These models are used in the design of a closed-loop active sensing platform for tracking multiple targets, discussed in Chapter 4. Throughout the modeling process, a compromise between representation accuracy and model complexity was sought, and it became clear that a more unified approach to modeling all the different elements involved is needed.

Chapter 4

Multitarget-Multisensor Tracking in Urban Terrain

We investigate the integration of detection, signal processing, tracking, and scheduling by exploiting three distinct levels of diversity: (1) spatial diversity through the use of coordinated multistatic radars; (2) waveform diversity by adaptively scheduling the transmitted waveform; and (3) motion model diversity by using a bank of parallel filters matched to different motion models. Specifically, we proposed a closed-loop active sensing system in which, at every radar scan, the waveform that yields the minimum trace of the one-step-ahead error covariance matrix is transmitted; the received signal goes through a matched-filter, and curve fitting is used to extract range and range-rate measurements that feed the LMIPDA-VSIMM algorithm for data association and filtering. Monte Carlo simulations demonstrate the effectiveness of the proposed system in an urban scenario contaminated by dense and uneven clutter, strong multipath, and limited line-of-sight.

4.1 Introduction

When compared to tracking airborne targets, tracking ground targets in urban terrain poses a new set of challenges. Target mobility is constrained by road networks, and the quality of measurements is affected by dense and uneven clutter, strong multipath, and limited line-of-sight.

We propose a closed-loop active sensing system for the urban terrain that integrates multitarget detection and tracking, multistatic radar signal processing, and waveform scheduling. The proposed system simultaneously exploits three distinct levels of diversity: (1) spatial diversity through the use of coordinated multistatic radars; (2) waveform diversity by adaptively scheduling the transmitted radar waveform according to the urban scene conditions; and (3) motion model diversity by using a bank of parallel filters, each one matched to a different motion model. Specifically, at each radar scan, the waveform that yields the minimum trace of the one-step-ahead error covariance matrix is transmitted; the received signal goes through a matched-filter, and curve fitting is used to extract measurements that feed the LMIPDA-VSIMM algorithm for data-association and filtering. The overall system is depicted in Figure 4.1. This feedback structure is fundamentally different from more conventional designs where processing is done sequentially without any feedback.

In this chapter, we explore the models presented in Chapter 3 and outline the main aspects of each building block of the proposed closed-loop active sensing platform for target tracking in urban terrain depicted in Figure 4.1. The remainder of this chapter is organized as follows. Section 4.2 describes the detection process performed by each radar receiver. Specifically, the matched filter detector is described. Radar imaging and how range and azimuth measurements are extracted are also explained. Section 4.3 discusses the various elements of the multitarget-multisensor tracker, including: automatic track initiation and termination criteria, measurement validation, filtering, and data association. In Section 4.4, waveform scheduling is considered. Simulations results are presented in Section 5.4. Finally, Section 5.5 concludes this chapter.

4.2 From Signal Detection to Discrete Measurements

Under the modeling assumption of AWGN, the optimal signal detection is the correlator receiver, or equivalently, the matched-filter [71]. The signal received by the lth element of the mth sensor array, given in Equation (3.3.2), is compared to a template signal by computing a correlation sum of sampled signals. The template signal is a time-shifted, time-reversed, conjugate and scaled replica of the signal transmitted by the nth transmitter at time t_k:

$$h_{k,n}(t) = as^{*}_{k,n}\left(t_d - t\right), \tag{4.2.1}$$

where t_d is the time-delay incurred during propagation, * represents complex conjugate, and a is the scaling factor assumed to be unity.

In the general multistatic setting with N transmitters and M receivers, we consider the case where the mth receiver is a uniform linear array of L_m sensor elements. Therefore, we need to combine the signals received by each of the sensor array elements to obtain the total signal received by the mth receiver at time t_k, which can be written as:

$$\mathbf{y}_{k,m} = \sum_{u=0}^{U-1} \sum_{t=1}^{T_1} \sum_{l=1}^{L_m} \mathbf{y}_{k,m,l}(u) h_{k,n}\left(uT_2 - t\right), \tag{4.2.2}$$

where $\mathbf{y}_{k,m,l}$ is given by Equation (3.3.2), and $h_{k,n}$ is given by Equation (4.2.1).

Before range and range-rate measurements that feed the tracker can be extracted, pre-processing the radar intensity image is necessary. Assuming each target is a point object (as opposed to an extended object with spatial shape), we use peak detection to locate a point source corresponding to a received power peak on a time-delay versus Doppler image. Due to strong local (but lack of global) similarities exhibited by urban clutter, image processing techniques aimed to suppress this type of clutter should be based on segmentation analysis and process each image segment individually in order to distinguish between targets of interest from background scatterers [2]. However, this could be extremely computationally intensive, therefore we use a more standard form of clutter suppression. Specifically, we calculate a background model prior to tracking using the average of radar intensity images over time to approximate the true urban scenario. The average background image is then subtracted from each image formed using radar returns during the tracking process. For each time-delay τ and Doppler shift ν, the average magnitude of the radar return is given by:

$$\bar{A}(\tau, \nu) = \frac{1}{J} \sum_{j=1}^{J} A_j(\tau, \nu), \tag{4.2.3}$$

where j indexes times during which the urban scenario was under surveillance prior to tracking, and

$$A_k(\tau, \nu) = A_k^-(\tau, \nu) - \bar{A}(\tau, \nu) \tag{4.2.4}$$

is the magnitude of the radar return for time-delay τ and Doppler shift ν at time step t_k during tracking. If further improvements on the contrast between background and targets of interest are needed, it is possible to reduce the image noise applying smoothing techniques.

Peak detection is implemented iteratively. For each peak $\left(\tau_k^{peak}, \nu_k^{peak}\right)$ found in $A_k(\tau, \nu)$, a nonlinear optimization algorithm is used to find a curve that fits the underlying image within a window centered at the peak. When performing curve fitting, we are interested in estimating the measurement error covariance matrix. Since noise sources are assumed to be AWGN, Gaussian curve fitting has been widely used in target detection. However, locally within a window centered at the peak, a Gaussian can be approximated by a quadratic. In this work, we fit a two-dimensional quadratic function to each peak in the underlying image. Specifically, at every time step t_k,we solve the following optimization problem:

$$\min_{\sigma_\tau,\sigma_\nu} \sum_\tau \sum_\nu \mid f_{\sigma_\tau,\sigma_\nu}(\tau,\nu) - A_k(\tau,\nu) \mid^2, \tag{4.2.5}$$

where for $\epsilon > 0$, the window containing time-delay and Doppler values is defined by $\tau \in \left(\tau_k^{peak} - \epsilon, \tau_k^{peak} + \epsilon\right)$ and $\nu \in \left(\nu_k^{peak} - \epsilon, \nu_k^{peak} + \epsilon\right)$, and

$$f_{\sigma_\tau,\sigma_\nu}(\tau,\nu) = \sigma_\tau^2\tau^2 + 2\sigma_\tau\sigma_\nu\tau\nu + \sigma_\nu^2\nu^2. \tag{4.2.6}$$

We define a scan as the set of measurements generated by a radar receiver from an individual look over the entire surveillance area. The kth scan by the mth receiver corresponding to its kth look is denoted by:

$$Z_{k,m} = \left\{\mathbf{z}_{k,m}^1, \mathbf{z}_{k,m}^2, \ldots, \mathbf{z}_{k,m}^{N_{k,m}}\right\}, \tag{4.2.7}$$

where $N_{k,m}$ is the total number of measurements in scan $Z_{k,m}$. In this work, the jth ($j = 1, \ldots, N_{k,m}$) measurement in the kth scan of the mth receiver is the following two-dimensional vector of range and range-rate:

$$\mathbf{z}^j_{k,m} = \begin{bmatrix} r^j_{k,m} \\ \dot{r}^j_{k,m} \end{bmatrix}, \tag{4.2.8}$$

where $\left(r^j_{k,m}, \dot{r}^j_{k,m}\right)$ corresponds to the location of the jth peak in the time-delay and Doppler image from receiver m by trivial transformation. Associated with each measurement vector $\mathbf{z}^j_{k,m}$ is an error covariance matrix

$$\mathbf{R}^j_{k,m}(\psi) = \begin{bmatrix} \sigma^2_{r^j_{k,m}} & \rho_{r\dot{r}}\left(\sigma_{r^j_{k,m}}\sigma_{\dot{r}^j_{k,m}}\right) \\ \rho_{r\dot{r}}\left(\sigma_{r^j_{k,m}}\sigma_{\dot{r}^j_{k,m}}\right) & \sigma^2_{\dot{r}^j_{k,m}} \end{bmatrix}, \tag{4.2.9}$$

where ψ is the vector of parameters that characterize the waveform transmitted at t_k, and $\rho_{r\dot{r}}$ is the correlation coefficient between range and range-rate measurement errors. The vector ψ is included in the measurement noise covariance matrix description to show the explicit dependence of this matrix on the transmitted waveform. In particular, transmitted waveforms defined by Equation (3.3.1) are characterized by pulse duration κ and chirp rate γ; hence, in this case,

$$\psi = \begin{bmatrix} \kappa \\ \gamma \end{bmatrix}. \tag{4.2.10}$$

The correlation coefficient between measurement errors $\rho_{r\dot{r}}$ also depends on the transmitted waveform, and can be calculated using the waveform's ambiguity function as shown in [69]. In particular, the correlation coefficient for the up-sweep and down-sweep chirp waveforms considered in this work are strongly negative and positive, respectively.

In state estimation, the measurement model describing the relationship between the target state at time t_k and the kth radar scan can be written as:

$$\mathbf{z}^j_{k,m} = \mathbf{H}\left(\mathbf{x}_k\right) + \mathbf{v}_k, \tag{4.2.11}$$

where $\mathbf{H}$ is a vector-valued function that maps the target state $\mathbf{x}_k$ to its range and range-rate, and $\mathbf{v}_k$ is the zero-mean Gaussian measurement noise vector with covariance matrix $\mathbf{R}^j_{k,m}(\psi)$.

4.3 Multitarget-Multisensor Tracker

We consider a tracker implemented as a sequential filter that weighs measurements in each scan. In addition, tracks are initiated, maintained and terminated in an integrated fashion. Note that we use the word *track* instead of *target* since we have no a priori knowledge of the number of targets in the urban scenario. Also, algorithms discussed in this section have been previously presented in the literature. Hence, rather than deriving each algorithm below, we highlight their main features related to the design of a closed-loop active sensing system for urban terrain.

4.3.1 Automatic Track Initiation and Termination

The goal in track initiation is to estimate tentative tracks from raw measurements without any prior information about how many targets are present in the surveillance area.

We follow the two-point differencing algorithm, according to which it takes two time steps (or two radar scans) for a track to be initiated [5]. For each receiver m ($m = 1, \ldots, M$), a tentative track is initiated for every declared detection, i.e., for every peak in the time-delay versus Doppler image exceeding a given detection threshold, and that cannot be associated with an existing track. In particular, at t_1 a tentative track $\mathbf{x}_1^{(j)}$ is initiated for each measurement $j = 1, \ldots, N_{1,m}$ in scan $Z_{1,m}$. Assuming the velocity of a target along the x and y coordinates lies within the intervals $\left[-\dot{x}_{k-1}^{max}, \dot{x}_{k-1}^{max}\right]$ and $\left[-\dot{y}_{k-1}^{max}, \dot{y}_{k-1}^{max}\right]$, respectively, a track is initiated at (x_k, y_k) when

$$x_k \in \left[\left(-\dot{x}_{k-1}^{max} - 2\sigma_{\dot{x}_{k-1}}\right) T, \left(\dot{x}_{k-1}^{max} + 2\sigma_{\dot{x}_{k-1}}\right) T\right] \tag{4.3.1}$$

and

$$y_k \in \left[\left(-\dot{y}_{k-1}^{max} - 2\sigma_{\dot{y}_{k-1}}\right) T, \left(\dot{y}_{k-1}^{max} + 2\sigma_{\dot{y}_{k-1}}\right) T\right], \tag{4.3.2}$$

where $T = t_k - t_{k-1}$ is the state sampling period, and $\sigma_{\dot{x}_{k-1}}$ and $\sigma_{\dot{y}_{k-1}}$ are the standard deviations of target velocities in x and y directions, respectively. Each measurement that falls into the track initiation area yields an initial position and velocity from which a track is initiated. Measurements can then be associated with this new track starting at $t_k > 2$, and the target's acceleration can then be estimated at the filtering stage of the tracker, described in Section 4.3.3.

The usual approach to track termination is to declare a track terminated if such track has not been associated with any new measurements for two consecutive time steps. We adopt a more integrated approach and use a probability of track existence, defined in Section 4.3.4, that is initialized for every initiated track. Specifically, a track is terminated if the probability of track existence falls below a given track termination threshold.

4.3.2 Measurement Validation

For each initiated track $\mathbf{x}_k^{(t)}$, $t = 1, \ldots, \mathrm{T}_k$, we define a gate in the measurement space within which measurements to be associated with track $\mathbf{x}_k^{(t)}$ are expected to lie. Only those measurements that lie within the gate are said to be validated, and are therefore associated to track $\mathbf{x}_k^{(t)}$. The size and shape of the gate can be defined in several different ways. We use the so-called ellipsoidal validation gating [56] and apply the following statistical test:

$$\left[\mathbf{z}_k^i - \hat{\mathbf{z}}_k^{(t)}\right]^\top \left(\mathbf{S}_k^{(t)}\right)^{-1} \left[\mathbf{z}_k^i - \hat{\mathbf{z}}_k^{(t)}\right] < g^2, \tag{4.3.3}$$

where $\mathbf{z}_k^i$ represents the ith measurement in the kth scan; $\hat{\mathbf{z}}_k^{(t)}$ is the predicted measurement for track $\mathbf{x}_k^{(t)}$; $\mathbf{S}_k^{(t)}$ represents the innovation covariance at scan k; and g is a threshold computed from Chi-square distribution tables, such that, if a target is detected, its measurement is validated with gating probability P_G. The number of degrees of freedom of g is equal to the dimension of the measurement vector. In the two-dimensional case, the area of the validation ellipse is $g^2\pi\,\det(\mathbf{S}_k^{(t)})^{1/2}$, where det is the matrix determinant.

4.3.3 Filtering

The tracking algorithm needs to be adaptive in order to handle a time-varying number of targets and dynamic urban conditions. We show in Section 4.5 that the variable structure interacting multiple model (VS-IMM) estimator is effective under such conditions [42, 55]. The VS-IMM estimator implements a separate filter for each model in its model set, which is determined adaptively according to the underlying terrain conditions. Specifically, at each time step t_k, the model set is updated to:

$$\mathcal{M}_k = \left\{ r_k \in \mathcal{M}^{total} \mid \mathcal{I}, \mathbf{x}_{k-1}^{(t,r)}, \mathbf{P}_{k-1}^{(t,r)}, r_{k-1} \in \mathcal{M}_{k-1} \right\}, \tag{4.3.4}$$

where $\mathbf{x}_{k-1}^{(t,r)}$ and $\mathbf{P}_{k-1}^{(t,r)}$ are the mean and covariance of track t in the filter matched to model r at t_{k-1}; $\mathcal{I}$ represents prior information about the urban scenario; and $\mathcal{M}^{total}$ is the set of all possible motion models. Changes in track trajectory are modeled as a Markov chain with transition probabilities given by:

$$\pi_{ij} = P\left\{r_k = i | r_{k-1} = j\right\}, \quad i, j \in \mathcal{M}^{total}. \tag{4.3.5}$$

In this work, we consider the unscented Kalman filter (UKF) algorithm. Initially proposed by Julier and Uhlmann [36], the UKF represents the state distribution by a set of deterministically chosen sample points. Each UKF filter matched to a different motion model runs in parallel in the VS-IMM framework. The estimated mean and covariance from each model-matched filter are mixed (Gaussian mixture) before the next filtering time step. The overall output of the VS-IMM estimator is then calculated by probabilistically combining the individual estimates of each filter [39].

4.3.4 Data Association

We consider the recently introduced linear multitarget integrated probabilistic data association (LMIPDA) algorithm [68, 66]. An extension of the single-target integrated probabilistic data association [67], LMIPDA models the notion of track existence as a Markov chain. Let χ_k denote the event that a track exists at t_k. The a priori probability that a track exists at t_k is given by:

$$\psi_{k|k-1} \triangleq \mathrm{P}\left\{\chi_k \,\middle|\, \bigcup_{i=1}^{k-1} Z_i\right\}, \tag{4.3.6}$$

where Z_k is the set of measurements from all receivers at time t_k, i.e.,

$$Z_k = \cup_{m=1}^{M} Z_{k,m}. \tag{4.3.7}$$

The evolution of track existence over time satisfies the following equations:

$$\psi_{k|k-1} = p_{11}\psi_{k-1|k-1} + p_{21}\left(1 - \psi_{k-1|k-1}\right) \tag{4.3.8}$$

$$1 - \psi_{k|k-1} = p_{12}\psi_{k-1|k-1} + p_{22}\left(1 - \psi_{k-1|k-1}\right), \tag{4.3.9}$$

where p_{ij}, $i, j = 1, 2$, are the corresponding transition probabilities.

The central ideal behind the LMIPDA algorithm is the conversion of a single-target tracker in clutter into a multitarget tracker in clutter by simply modifying the clutter measurement density according to the predicted measurement density of other tracks. The modified clutter density of track $\mathbf{x}_k^{(t)}$ given the ith measurement can be written as:

$$\Omega_i^{(t)} = \rho_i^{(t)} + \sum_{s=1, s\neq t}^{T_k} p_i^{(s)} \frac{P_i^{(s)}}{1 - P_i^{(s)}}, \tag{4.3.10}$$

where $\rho_i^{(t)}$ is the clutter density in the validation gate of track $\mathbf{x}_k^{(t)}$, given the ith measurement in the kth scan $\mathbf{z}_k^i$; $P_i^{(t)}$ is the a priori probability that $\mathbf{z}_k^i$ is the true measurement for track $\mathbf{x}_k^{(t)}$, i.e.,

$$P_i^{(t)} = P_D P_G \psi_{k|k-1}^{(t)} \frac{p_i^{(t)}/\rho_i^{(t)}}{\sum_{i=1}^{N_k^{(t)}} p_i^{(t)}/\rho_i^{(t)}}, \tag{4.3.11}$$

where $\psi_{k|k-1}^{(t)}$ is the probability of existence of track $\mathbf{x}_k^{(t)}$, $p_i^{(t)}$ is the a priori measurement likelihood (Gaussian density), and $N_k^{(t)}$ is the total number of measurements associated with track $\mathbf{x}_k^{(t)}$ at time t_k. The probability of track existence is calculated as follows:

$$\psi_{k|k}^{(t)} = \frac{\left(1 - \delta_k^{(t)}\right) \psi_{k|k-1}^{(t)}}{1 - \delta_k^{(t)} \psi_{k|k-1}^{(t)}}, \tag{4.3.12}$$

where

$$\delta_k^{(t)} = P_D P_G \left(1 - \sum_{i=1}^{N_k^{(t)}} \frac{p_i^{(t)}}{\Omega_i^{(t)}}\right). \tag{4.3.13}$$

For each model $r \in \mathcal{M}_k$, we define the following probabilities of data association:

$$\beta_{k,0}^{(t,r)} = \frac{1 - P_D P_G}{1 - \delta_k^{(t,r)}} \tag{4.3.14}$$

for clutter measurements and, for each target measurement $i > 0$,

$$\beta_{k,i}^{(t,r)} = \frac{1 - P_D P_G p_i^{(t,r)}}{\left(1 - \delta_k^{(t,r)}\right) \Omega_i^{(t)}}, \tag{4.3.15}$$

where $p_i^{(t,r)}$ is the a priori likelihood of measurement i assuming association with track $\mathbf{x}_k^{(t)}$ that follows motion model r, i.e.,

$$p_i^{(t)} = \sum_{r \in \mathcal{M}_k} p_i^{(t,r)}, \tag{4.3.16}$$

and

$$\delta_k^{(t,r)} = P_D P_G \left(1 - \sum_{i=1}^{N_k^{(t)}} \frac{p_i^{(t,r)}}{\Omega_i^{(t)}}\right). \tag{4.3.17}$$

Finally, the motion model for each track $\mathbf{x}_k^{(t)}$ is updated according to the following model probabilities:

$$\mu_k^{(t,r)} = \mu_{k|k-1}^{(t,r)} \frac{1 - \delta_k^{(t,r)}}{1 - \delta_k^{(t)}}, \tag{4.3.18}$$

for $r \in \mathcal{M}_k$.

4.4 Waveform Scheduling

Many modern airborne radars have a waveform scheduler implemented. Ideally, the scheduler would use a library of waveforms especially designed to improve detection and the overall tracking performance.

We consider the general waveform selection problem, which in the multitarget tracking case can be written as:

$$\min_{\psi \in \Psi} \frac{1}{\mathrm{T}_k} \sum_{t=1}^{\mathrm{T}_k} \mathrm{E}\left\{ \|\mathbf{x}_k^{(t)} - \hat{\mathbf{x}}_k^{(t)}\|^2 \mid Z_k \right\}, \tag{4.4.1}$$

where Ψ represents the waveform library, $\hat{\mathbf{x}}_k$ is the tracking state estimate, and T_k is the total number of tracks at t_k.

In particular, we consider the one-step ahead (or myopic) waveform scheduling problem, where the waveform selected for transmission at t_{k+1} is given by:

$$\psi_{k+1} = \underset{\psi_{k+1} \in \Psi}{\operatorname{argmin}} \frac{1}{\mathrm{T}_k} \sum_{t=1}^{\mathrm{T}_k} \mathrm{Tr}\left\{ \mathbf{P}_{k+1}^{(t)}\left(\psi_{k+1}\right) \right\}, \tag{4.4.2}$$

where Tr is the matrix trace and $\mathbf{P}_{k+1}^{(t)}$ is the posterior state error covariance matrix corresponding to track $\mathbf{x}_k^{(t)}$. The performance measure in Equation (4.4.2) is equivalent to minimizing the mean square tracking error over all existing tracks. The posterior state covariance error matrix $\mathbf{P}_{k+1}^{(t)}$ defines a six-dimensional ellipsoid centered at $\mathbf{x}_k^{(t)}$ that is a contour of constant probability of error [88], and its trace is proportional to the perimeter of the rectangular region enclosing this ellipsoid.

In order to evaluate Equation (4.4.2), we first approximate the measurement error covariance matrix by the Fisher information matrix $\mathbf{J}(\psi)$ corresponding to the measurement using waveform ψ [38, 88]. Specifically,

$$\mathbf{R}(\psi) = \mathbf{U}\mathbf{J}(\psi)^{-1}\mathbf{U}^{\top}, \tag{4.4.3}$$

where $\mathbf{U}$ is the transformation matrix between the time-delay and Doppler measured by the receiver and the target's range and range-rate. In particular, for the up-sweep Gaussian chirp with pulse duration κ, chirp rate γ, and wavelength λ defined by Equation (3.3.1), we have:

$$\mathbf{R}(\psi) = \frac{1}{\eta}\begin{bmatrix} \frac{c^2\kappa^2}{2} & -\frac{2\pi c^2\gamma\kappa^2}{\lambda} \\ -\frac{2\pi c^2\gamma\kappa^2}{\lambda} & \left(\frac{2\pi c}{\lambda}\right)^2\left(\frac{1}{2\kappa^2}+2\gamma^2\kappa^2\right) \end{bmatrix}, \tag{4.4.4}$$

where η is the signal-to-noise ratio (SNR). A similar expression can be obtained for the down-sweep Gaussian chirp.

The posterior state error covariance matrix can then be calculated for each waveform $\psi \in \Psi$ using the UKF's covariance update equations.

4.5 Simulation Setup and Results

Monte Carlo simulations are used to evaluate the effectiveness of the proposed closed-loop system in urban terrain. The scenario used is shown in Figure 3.2, where the overall clutter density is assumed to be $2.5\mathrm{e}^{-4}\mathrm{m}^2$. The maximum sensor range is 300 meters, and the SNR experienced is 0.2.

Although in reality the transmitted signal can be reflected by multiple scatterers, we assume that the strength of the radar return is negligible after three reflections, therefore we restrict our simulation to the following paths: transmitter→target→receiver (direct path), transmitter→clutter→receiver, transmitter→target→clutter→receiver, transmitter→clutter→target→receiver, transmitter→clutter→clutter→receiver, transmitter→clutter→target→clutter→receiver, transmitter→target→clutter→clutter→receiver, and transmitter→clutter→clutter→target→receiver.

The simulation experiment consisted of 100 runs, each with a total of 140 radar scans, where a radar scan takes 0.25 seconds. Two targets 10 seconds apart from each other are simulated using the same trajectory as follows. Starting at (1950,1500), each target moves at constant velocity of 10 m/s in the x direction for 10 seconds; as they approach the intersection, they start decelerating at constant rate of 1 m/s^2 for 5 seconds; they enter a left turn with constant turn rate of $\pi/20$ rad/s for 10 seconds; after completing the turn, each target accelerates for 5 seconds at 1 m/s^2 rate; finally, they end their trajectories with constant velocity at (2068.8,1667.8).

Two model sets are used during motion model adaptation. In the vicinity of intersections, a set consisting of NCA, left-turn CT with turn rate of $\pi/20$ rad/s, and right-turn CT with the same turn rate is used. Specifically for the simulated trajectory, this model set is used between scans 20 and 100. During the remaining radar scans, a set containing the NCA and NCV motion models is used instead. We consider the following motion model transition probability matrix is:

$$\begin{pmatrix} 0.99 & 0.01 & 0 & 0 \\ 0.1 & 0.7 & 0.1 & 0.1 \\ 0 & 0.1 & 0.99 & 0 \\ 0 & 0.1 & 0 & 0.99 \end{pmatrix}.$$

A waveform library consisting of four different Gaussian-windowed chirp signals is considered. Waveforms vary in pulse duration κ. In particular, radar sensors considered support the following pulse durations: $\kappa = 0.5$ μs, and $\kappa = 1.375$ μs. In general, longer pulses return more power; however, finer details may be lost.

In addition, waveforms of each pulse duration can be either an up-sweep or down-sweep chirp. Pulses are repeated at every 10 milliseconds, and waveforms operate at 4 GHz with 40 MHz of bandwidth.

A more traditional open-loop system, which does not support any diversity modes, is used as a baseline for comparison. In the baseline implementation, a single UKF using the NCV motion model is considered.

Note that the simulation parameters used in this work do not represent any particular system, and were chosen exclusively for illustration purposes.

Simulation results in Figure 4.2 and Figure 4.3 show that the closed-loop system clearly outperforms its open-loop counterpart. The average number of confirmed tracks is increased by approximately 15% over 140 radar scans, and the position RMSE is reduced by approximately 60%. Figure 4.4 shows the evolution of each motion model probability over time. Although there is some "model competition" between NCV and NCA, the closed-loop system satisfactorily identifies the correct motion model throughout the simulation.

4.6 Concluding Remarks

The closed-loop active sensing system discussed in this chapter highlights the major challenges in the design of multisensor-multitarget tracking systems, while significantly outperforming its open-loop counterpart. New capabilities for tracking and surveillance, and the seamless integration of different sensing platforms will only be possible with advances in the areas of multisensor data fusion, intelligent algorithms for signal processing and resource allocation, and creative ways to unravel multipath propagation. This work is a first step towards understanding how these research areas interact from a systems engineering perspective to ultimately be integrated into a active sensing tracking platform that operates effectively in urban terrain.

Future work in this area could include tracking dismounts, and the investigation of coordinated non-myopic waveform scheduling schemes for distributed transmitters, which do not have to simultaneously transmit the same waveform. One could also consider the expansion of the simulated waveform library. Mobile sensing platforms, such as UAVs and vehicular sensor networks, are especially important in hostile urban environments, since their completely distributed and opportunistic nature makes it difficult for hostiles to disable surveillance, while potentially increasing the coverage area.

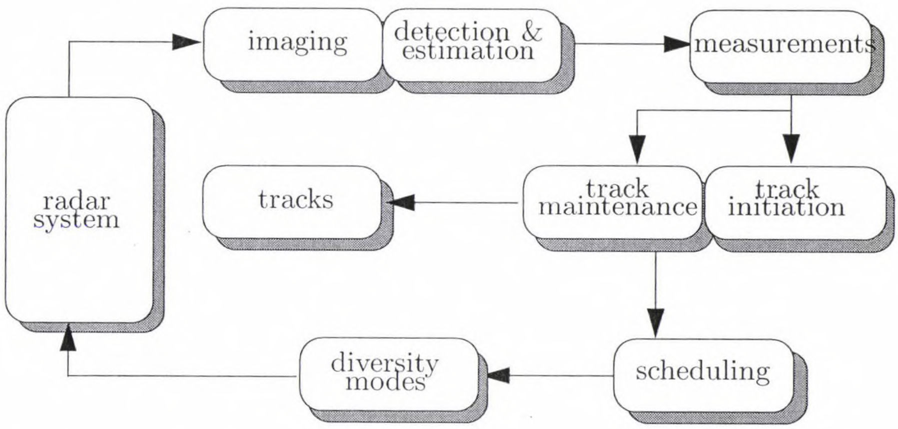

Figure 4.1: Systems-level architecture of the proposed closed-loop active sensing platform.

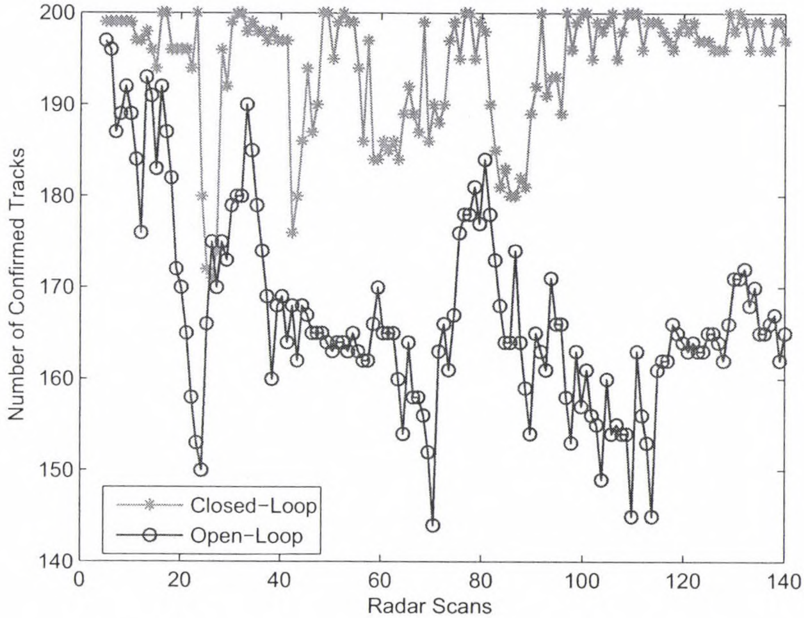

Figure 4.2: Number of confirmed tracks for close-loop and open-loop systems.

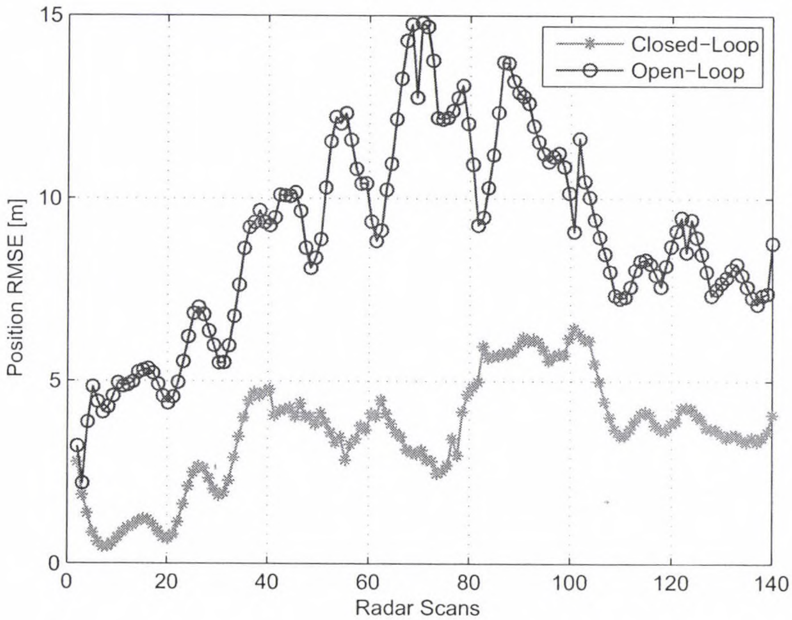

Figure 4.3: Position RMSE for closed-loop and open-loop systems.

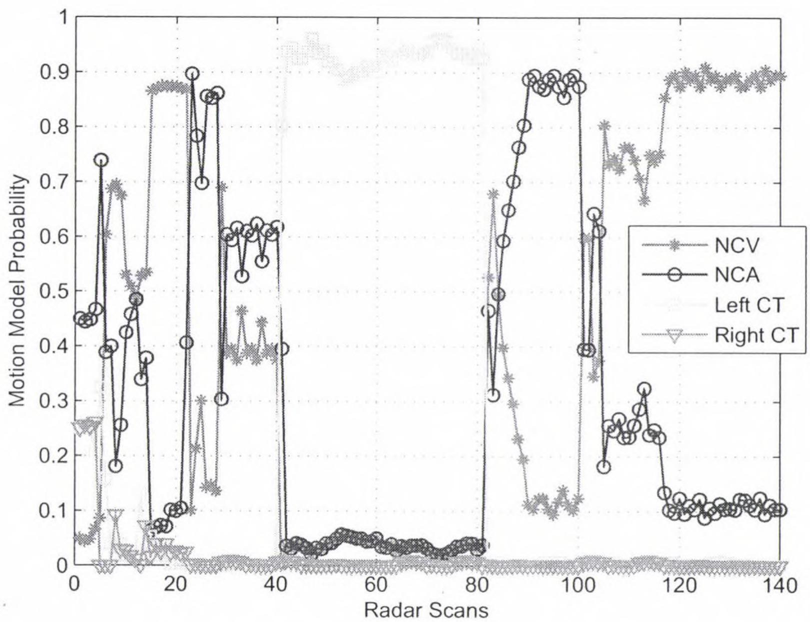

Figure 4.4: Motion model probabilities in the closed-loop system.

Chapter 5

Two-Level Scheduling for Target Tracking in Covert Operations

We consider an active sensing system where multiple waveform-agile radars scan a hostile surveillance area for targets. A central controller adaptively schedules the sequence of transmitters and waveforms that maximizes the overall tracking accuracy, while simultaneously maintaining the sensing systems's covertness. We formulate this problem as a partially observable Markov decision process (POMDP), and propose a novel "two-level" scheduling scheme that uses two distinct schedulers: (1) at the lower level, a myopic waveform scheduler; and (2) at the upper level, a non-myopic transmitter scheduler. Scheduling decisions at these two levels are carried out separately. Although waveforms are updated at every radar scan, a new set of transmitters only becomes active if the overall tracking accuracy falls below a given threshold, or if a "detection risk" is exceeded. By simultaneously exploiting myopic and non-myopic scheduling schemes, we benefit from trading off short-term for long-term performance, while maintaining low computational costs. Monte Carlo simulations are used to evaluate the performance of the proposed adaptive sensing scheme in a multitarget tracking setting.

5.1 Introduction

Recent advances in sensing technology and embedded systems have made it possible to deploy multiple sensors for a variety of applications, including: reconnaissance, surveillance and target tracking; autonomous vehicle navigation; and remote habitat monitoring. Regardless of their nature, multisensor applications rely on effectively managing sensor resources. In particular, next-generation multifunctional agile radars demand innovative resource management techniques to achieve a common sensing goal while satisfying resource constraints. In this work, this dynamic management of sensor resources is called adaptive sensing.

We expand upon the closed-loop active sensing system for urban terrain proposed in Chapter 4, and consider a more general system that also exploits transmit diversity. In particular, we consider an active sensing system where multiple waveform-agile radars scan a hostile surveillance area for targets, and a central controller adaptively schedules the sequence of transmitters and waveforms that maximizes the overall tracking accuracy, while simultaneously maintaining the sensing system's covertness. We formulate this adaptive sensing problem as a POMDP, and use this formulation to develop a novel scheduling scheme. Specifically, we use two distinct schedulers: (1) at the lower level, a myopic waveform scheduler; and (2) at the upper level, a non-myopic transmitter scheduler. Scheduling decisions at these two levels are carried out separately. While waveforms are updated at every radar scan, a new set of transmitters is activated only if the overall tracking accuracy falls below a given threshold, or if a detection risk is exceeded. We define the detection risk as a limit on the number of consecutive scans during which a transmitter remains active. By simultaneously exploiting myopic and non-myopic scheduling schemes, we benefit from trading off short-term for long-term performance, while maintaining low computational costs.

The remainder of this paper is organized as follows. Section 5.2 describes our adaptive sensing problem, the motivation behind the two-level controller, and its outline. In Section 5.3, we provide the POMDP formulation of the problem, its corresponding solution, and the relationship between the two scheduling timeframes involved. In Section 5.4, Monte Carlo simulations are used to evaluate the performance of the proposed adaptive sensing scheme in an urban multitarget tracking setting. In addition to mean square error results, we analyze the frequency of transmitter activation when varying the detection risk. Finally, we conclude in Section 5.5 with summary remarks and future directions.

5.2 Adaptive Sensing Platform

The central controller is faced with the following tradeoff: either activate a transmitter that provides higher quality measurements in the short term, but that also exceeds the detection risk, thus jeopardizing future sensing ability; or activate a transmitter that yields poorer quality measurements, but that poses lower detection risk.

Figure 5.1 illustrates how tracking and scheduling components interact to form the proposed adaptive sensing platform. This basic systems-level control architecture is particularly suitable for the lookahead POMDP framework discussed in Section 5.3. The top block depicts the sensing system, which in this work consists of multiple waveform-agile radars distributed over a surveillance area. The sensing system's inputs are actions (external control commands) that represent which transmitters to activate and which waveforms to transmit. The corresponding outputs can be divided into two different classes: (1) fully observable sensing quantities, such as sensor locations, which transmitters are activated, and which waveforms are transmitted; and (2) unobservables, i.e., measurements of those aspects that are not directly observable, such as sensor outputs representing observations of the target state. The bottom block depicts the proposed two-level controller and its two main components: the tracker and the scheduler (action selector). At each time step, the tracker provides the current state estimate, i.e., the posterior distribution of unobservables. The two-level controller takes both observables and unobservables estimates into consideration to make sensing allocation decisions, and output control actions. These actions are then used to generate new measurements that are subsequently used by the tracker, thus closing the loop.

5.3 Problem Formulation

In this section, we formulate the adaptive sensing problem in the POMDP framework to include long-term performance considerations. The POMDP model formalizes the interaction between the sensing system and the two-level controller depicted in Figure 5.1. In the literature, POMDPs have been used in dynamic probabilistic systems to make sequential decisions that optimize an appropriate objective. We start with a brief introduction to POMDPs. Since a review of POMDP optimization techniques is beyond the scope of this paper, we refer the reader to Bertsekas

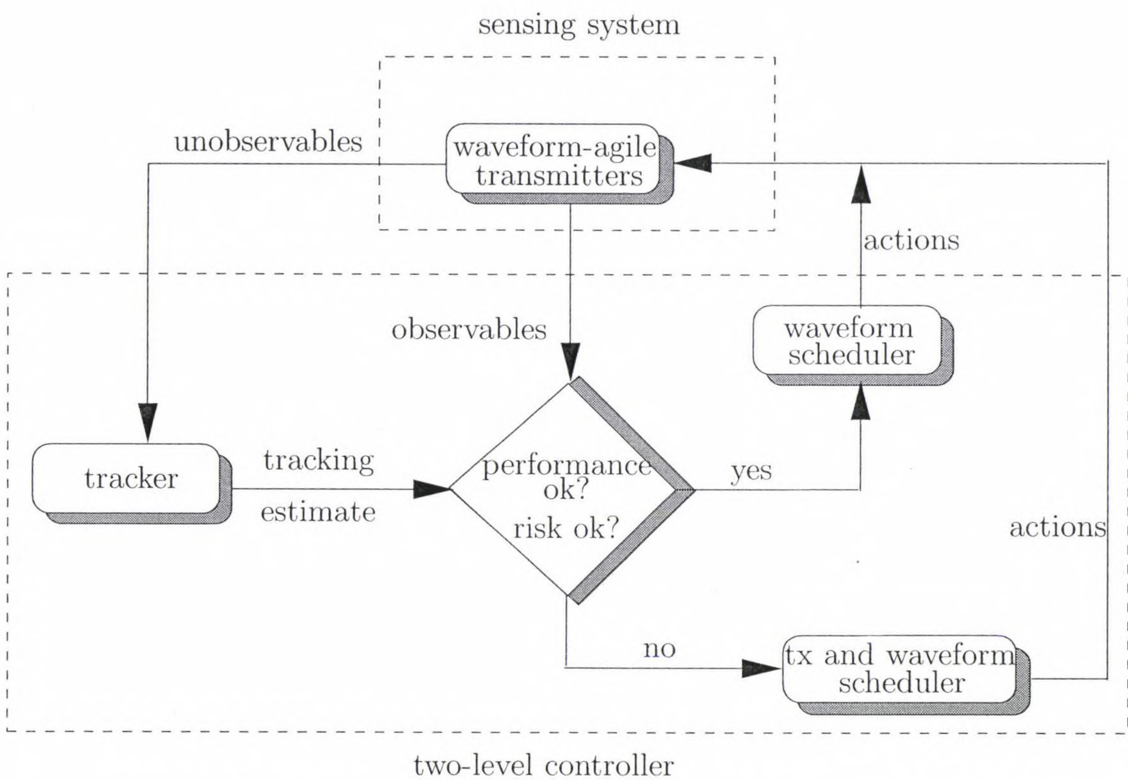

Figure 5.1: Systems-level control architecture of the adaptive sensing platform.

work on dynamic programming and optimal control [9] for a comprehensive and rigorous treatment of POMDPs.

5.3.1 POMDP Definition

A POMDP generalizes the Markov decision process model [64, 76, 72] by incorporating the notion that some aspects of the environment are not directly observable; therefore decision makers may not be able to perfectly monitor such an environment [8]. A POMDP consists of the following:

- A finite set of possible states $\mathcal{X}$, and a distribution p_0 specifying the random initial state. A state describes features that evolve over time.
- A finite set of possible actions $\mathcal{A}$ over controllable aspects of the sensing system that may be selected by the scheduler.
- A state-transition law that represents how states change over time by specifying the next-state distribution given an action chosen at the current state, i.e., it defines the transition probability $p\left(x_{k+1} \mid x_k, a_k\right)$, $x_k \in \mathcal{X}$, $a_k \in \mathcal{A}$, for any time step $k \geq 0$.
- A set of possible observations $\mathcal{Z}$, describing those features that depend on the states or that are directly observable, including prior actions.
- An observation law that relates states and actions to observations through the distribution of observations given the current action and the resulting state, i.e., it defines the observation probability $p\left(z_{k+1} \mid x_{k+1}, a_k\right)$, $z_k \in \mathcal{Z}$, $x_k \in \mathcal{X}$, $a_k \in \mathcal{A}$, for any time step $k \geq 0$.

- A one-step cost function $C(x_k, a_k)$ specifying the immediate cost incurred (a real number) by choosing action $a_k \in \mathcal{A}$ at state $x_k \in \mathcal{X}$.

Since the states in a POMDP are not directly observable, there is an uncertainty about the state space, represented by the so-called belief state. Specifically, the belief state b_k is the posterior probability distribution of the underlying state given the history of observations up to time step k, i.e.,

$$b_k(x) = p_{x_k}\left(x \mid z_0, \ldots, z_k; a_0, \ldots, a_{k-1}\right), \tag{5.3.1}$$

and b_0 is its initial distribution.

Given b_k, the scheduler selects an action $a_k \in \mathcal{A}$, incurring a cost $C(x_k, a_k)$; the state transitions from x_k to (unobserved) state x_{k+1} with probability $p(x_{k+1} \mid x_k, a_k)$; and the observation $z_{k+1} \in \mathcal{Z}$ is made according to $p(z_{k+1} \mid x_k, a_k)$. The process is repeated at the state x_{k+1}. Hence, the belief-state can be recursively updated using Bayes rule as follows:

$$b_{k+1} = \frac{p\left(z_{k+1} \mid x_{k+1}, a_k\right) \sum_{x \in \mathcal{X}} p\left(x_{k+1} \mid x_k, a_k\right) b_k}{p\left(z_{k+1} \mid b_k, a_k\right)}, \tag{5.3.2}$$

where $x_k \in \mathcal{X}$, $a_k \in \mathcal{A}$, $z_k \in \mathcal{Z}$, for $k \geq 0$.

Therefore, a POMDP is a Markov decision process where the states are only partially observable through the observation law, but where the belief state is fully observable and represents a sufficient statistic. Hence, scheduling decisions in a POMDP are based on recursively calculating the belief state. Figure 5.2 shows the causal relationship between POMDP states, actions, observations, and costs.

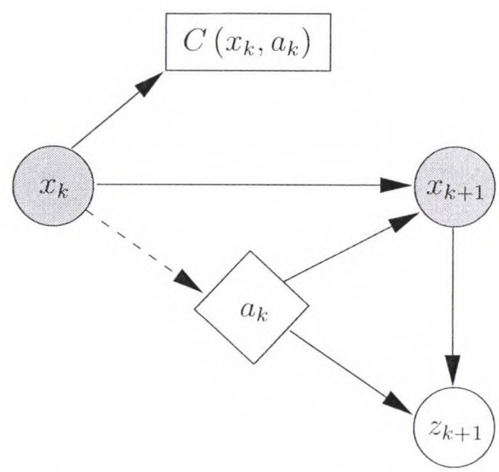

Figure 5.2: Causal representation of a POMDP. Circles represent random variables; shaded circles indicate unobserved random variables, and un-shaded circles indicate observed variables. A diamond represents a decision node, and a rectangle represents a cost node. Solid directed arcs indicate causal effect, whereas the dashed arc indicates that a distribution is used instead of the actual unobserved valued.

5.3.2 POMDP Formulation

To formulate our adaptive sensing problem as a POMDP, we must first specify each of the components defined in Section 5.3.1 in the context of the proposed problem. Following the notation in [63], we have:

States

We consider a factored state vector with three components: the sensor(s) state s_k, the target(s) state ζ_k, and the tracker state (ξ_k, P_k). Hence, we can write the state vector at time step k as

$$x_k = (s_k, \zeta_k, \xi_k, P_k) . \tag{5.3.3}$$

In the proposed adaptive sensing problem, s_k specifies the number of consecutive time steps each sensor has been active up to (and including) time k; and ζ_k specifies the position, velocity, and acceleration of each target at time k. Assuming a Kalman filter-based tracker, (ξ_k, P_k) represents the tracker internal state at time k, where ξ_k is the posterior mean vector and P_k is the posterior covariance matrix.

Actions

Waveform-agile radars are controlled through a sensor activation control and a waveform mode control. Therefore, the action at each time step is to specify which transmitters are active (or idle), and which waveform is sent by active transmitters. We denote the action space by $\mathcal{A} = \{0, 1\}^{N\times\Omega}$, where N is the total number of transmitters in the surveillance area and Ω is the total number of waveforms available in the library. For instance, at time step k, the action is a matrix

$$a_k = \begin{bmatrix} a_k^{(1,1)} & \cdots & a_k^{(1,\Omega)} \\ \vdots & \ddots & \vdots \\ a_k^{(N,1)} & \cdots & a_k^{(N,\Omega)} \end{bmatrix}, \tag{5.3.4}$$

where $a_k^{(n,\omega)} = 1$ specifies that the nth transmitter is active and transmits waveform ω at time step $k+1$ to generate observations based on the system state x_k. Conversely, $a_k^{(n,\omega)} = 0$ specifies that waveform ω is not sent by transmitter n at time step $k+1$. For a given transmitter n and some $k \geq 0$,

$$\sum_{\omega=1}^{\Omega} a_k^{(n,\omega)} = 0 \tag{5.3.5}$$

if and only if this transmitter is idle at $k+1$. Note that radars whose transmitters are idle still operate in passive mode, i.e., they could still receive echoes from the scatterers in the surveillance area.

State-transition law

We decompose the state transition law in three parts, corresponding to the evolution in time of each of the three state components. The sensor state evolves according to

$$s_{k+1} = \psi\left(s_k, a_k\right), \tag{5.3.6}$$

where ψ is the map defining how the sensor state changes from one time step to the next, according to the action a_k. Specifically, for the nth transmitter we have:

$$s_{k+1}^{(n)} = s_k^{(n)} + 1, \quad \text{if } a_k^{(n,\omega)} = 1 \text{ for some } \omega \in [1, \Omega], \tag{5.3.7}$$

and

$$s_{k+1}^{(n)} = 0, \quad \text{if } a_k^{(n,\omega)} = 0 \text{ for all } \omega \in [1, \Omega]. \tag{5.3.8}$$

The evolution in time of the target state is defined by the target kinematics

$$\zeta_{k+1} = f(\zeta_k) + v_k, \tag{5.3.9}$$

where v_k represents the randomness in the target state transition and f defines the target motion model. Finally, the track state evolves according to a Kalman filter, which updates the posterior mean ξ_k and covariance P_k.

Observations and observation law

The observation z_k is generated according to the observation law, which depends on the sensor model. We factor z_k into three components corresponding to the state factorization as follows. Since the sensor state and the track state are fully observable, their corresponding observations are equal to their respective underlying state components, i.e.,

$$z_k^s = s_k, \quad z_k^\xi = \xi_k, \quad z_k^P = P_k. \tag{5.3.10}$$

On the other hand, the target state is only partially observable through noisy sensor measurements. In this work, each sensor measurement consists of a number of observations that can be measurements from established tracks, false alarms from clutter, or measurements from newly initiated tracks. We consider the detection scheme described in our prior work [6], and use peak detection radars to locate a point source corresponding to a received power peak in a time-delay versus Doppler image. For the collection of target state observations from all sensors at time k, z_k^ζ, we can write

$$z_k^\zeta = h(\zeta_k, s_k, w_k), \tag{5.3.11}$$

where h is a function that maps the target state ζ_k to its time-delay and Doppler (or to its range and range-rate through trivial transformation), and w_k represents the zero-mean Gaussian measurement noise with covariance matrix $R_k(\zeta_k, s_k)$. For each peak found in the time-delay versus Doppler image at time step k, a nonlinear optimization algorithm is used to find a curve that fits the underlying image within a window centered at the peak to estimate $R_k(\zeta_k, s_k)$.

Belief State

Given the factorization of state x_k, the belief state at time step k can be written as:

$$b_k = \left(b_k^s, b_k^\zeta, b_k^\xi, b_k^P\right), \tag{5.3.12}$$

Each component of the factored belief state b_k is updated according to:

$$b_k^s(s) = \delta(s - s_k), \tag{5.3.13}$$

$$b_k^\zeta(\zeta) = \mathcal{N}(\zeta - \xi_k, P_k), \tag{5.3.14}$$

$$b_k^\xi(\xi) = \delta(\xi - \xi_k), \tag{5.3.15}$$

$$b_k^P(P) = \delta(P - P_k), \tag{5.3.16}$$

where $\mathcal{N}(\zeta - \xi_k, P_k)$ denotes the normal distribution of the target state ζ with mean ξ_k and covariance P_k; and $\delta(s - s_k)$ is the Dirac delta function centered at $s = s_k$. Note that when defining the belief state corresponding to the target state we assumed a perfect tracking model and Gaussian statistics; hence this belief state component can be updated using a Kalman filter.

Cost function

In this work, the goal is to minimize the total accumulated tracking error. The most commonly used measure of tracking accuracy is the mean square error (MSE). Hence, the one-step cost function $C(x_k, a_k)$ simply maps the state and action at each time step to the mean square tracking error incurred, i.e.,

$$C(x_k, a_k) = \underset{v_k, w_{k+1}}{\mathrm{E}} \left[\|\zeta_{k+1} - \xi_{k+1}\|^2 \mid x_k, a_k \right], \tag{5.3.17}$$

where $\mathrm{E}[\cdot \mid x_k, a_k]$ denotes the conditional expectation given state x_k and action a_k.

Before we describe the POMDP approximate solution, it is important to understand the two scheduling timeframes involved. While our problem formulation and solution approach are general enough for both myopic and non-myopic schedulers, and do not depend on how the scheduling decision epoch is specified, each scheduler in the proposed two-level controller operates on a different timeframe, as explained below. Let $\Delta\tau$ be the decision interval, that is, the difference between two decision epochs, corresponding to the non-myopic scheduler. Similarly, let $\Delta\delta$ be the decision interval corresponding to the myopic scheduler, that is, the difference between two radar scans. Note that, while $\Delta\delta$ is fixed, $\Delta\tau$ varies according to tracking performance. However, assuming the detection risk is given by $T_D \geq 1$, $\Delta\tau$ is bounded between $\Delta\delta$ and T_D, i.e.: $\Delta\delta \leq \Delta\tau \leq T_D$. Figure 5.3 illustrates this relationship.

5.3.3 POMDP Approximate Solution

Given the definition of one-step cost function, at each time step $k \geq 0$, a scheduler's objective is to choose an action $a_k \in \mathcal{A}$ based on the belief state b_k that minimizes the total expected cost. We call such decision rule a scheduling policy. Specifically, a policy is the sequence of mappings from belief states to actions $\pi = \{\pi_k\}_{k \geq 0}$, where $\pi_k(b_k) \in \mathcal{A}$, and an optimal policy $\pi_k^*(b_k) \in \mathcal{A}$ is the one which minimizes the total expected cost.

Assuming a finite horizon of H time steps, the total expected cost over the horizon is given by

$$J_H = \mathrm{E}\left[\sum_{k=0}^{H-1} C(x_k, a_k)\right], \tag{5.3.18}$$

which can be written in terms of belief states as

$$J_H = \mathrm{E}\left[\sum_{k=0}^{H-1} c(b_k, a_k) \;\middle|\; b_0\right], \tag{5.3.19}$$

where

$$c(b_k, a_k) = \sum_{x \in \mathcal{X}} C(x, a_k) b_k(x) \tag{5.3.20}$$

is the cost incurred by selecting action a_k at belief state b_k. In this work, we assume perfect data association and approximate $c(b_k, a_k)$ by

$$c(b_k, a_k) = \int \mathop{\mathrm{E}}_{v_k, w_{k+1}} \left[\|\zeta_{k+1} - \xi_{k+1}\|^2 \mid s_k, \zeta, \xi_k, a_k\right] b_k^{\zeta}(\zeta)\, d\zeta = \mathrm{Tr}(P_{k+1}), \tag{5.3.21}$$

where Tr is the trace operator, which in this case is proportional to the perimeter of the rectangular region enclosing the covariance ellipsoid. In scenarios of dense clutter and multiple highly maneuvering crossing targets, it might be wise to reconsider this approximation. However, since our main goal is to show the effectiveness of the proposed scheme, we adopt this approximation throughout our work.

According to Bellman's Optimality Principle for POMDPs [8], the optimal policy $\pi_k^*(b_k)$ is given by

$$\pi_k^*(b_k) = \underset{a}{\operatorname{argmin}} \left\{ c(b_k, a) + \mathrm{E}\left[J_{H-k-1}^*(b_{k+1}) \mid b_k, a \right] \right\} \tag{5.3.22}$$

where $J_{H-k-1}^*(b_{k+1})$ is the so-called expected cost-to-go, that is, the optimal expected total cost value over $H-k-1$ time steps, starting at the next belief state b_{k+1}.

By defining the Q-value as

$$Q_{H-k}(b_k, a) = c(b_k, a) + \mathrm{E}\left[J_{H-k-1}^*(b_{k+1}) \mid b_k, a \right], \tag{5.3.23}$$

the optimal policy according to Bellman's principle can be written as

$$\pi_k^* = \underset{a}{\operatorname{argmin}} \left\{ Q_{H-k}(b_k, a) \right\}. \tag{5.3.24}$$

Hence, the optimal action at belief state b_k is the one with smallest Q-value.

Assuming the time horizon is sufficiently long, the optimal policy can be assumed to be stationary, i.e.,

$$\pi^* = \underset{a}{\operatorname{argmin}} \left\{ Q_H(b, a) \right\}. \tag{5.3.25}$$

Hence, the choice of action at a given belief state does not depend on any time index. In this case, the difference between Q_H and Q_{H-1} is negligible, and the horizon is constant at H regardless of the current time k. Thus the Q-value can be written as

$$Q(b, a) = c(b, a) + \mathrm{E}\left[J^*(b') \mid b, a \right], \tag{5.3.26}$$

where b' is the next belief state, that is, the belief state after choosing action a at belief state b; $c(b, a)$ is the cost incurred by choosing this action; and J^* is called the cost-to-go, that is, the optimal expected total cost over the horizon. This approach is called receding horizon control.

In general, analytical calculation of the Q-value is impossible since the complexity of searching for an optimal solution grows exponentially with the state and action spaces. Therefore, it is necessary to use approximate solutions. Several Q-value approximation methods have been proposed in the literature [17]. Many of these methods, including policy rollout/CO-rollout and hindsight/foresight optimization, usually involve approximating the Q-value through Monte Carlo simulations. However, Monte Carlo approximation methods can be very computationally expensive; hence we use the recently proposed nominal belief-state optimization (NBO) method [63]. In this approach, the cost-to-go is approximated by

$$J^*(b) = \min_{(a_0, a_1, \ldots, a_{H-1})} \sum_{k=0}^{H-1} c(\hat{b}_k, a_k), \tag{5.3.27}$$

where the nominal belief state $\hat{b}_k$ is the maximum a posteriori (MAP) estimate of b_k, and the minimization is over a sequence of actions. The NBO method, which approximates the belief-state evolution, is particularly suitable for our adaptive sensing problem, in which the primary source of randomness is in the cost function (mean square tracking error), and where it would be prohibitive to simulate multiple random noise samples to estimate expectations and calculate the belief state analytically.

5.4 Simulation Experiments

In this section, we start by describing our simulation setup. Our goal is to demonstrate the effectiveness of the proposed two-level scheduling scheme in a multitarget tracking setting when compared to a fully-myopic baseline scheduler. Therefore, simulation parameters used do not represent any particular system, and were chosen exclusively for illustration purposes. We then analyze simulation results with respect to tracking accuracy and frequency of transmitter activation with varying detection risk.

5.4.1 Simulation Setup

We exploit the dynamic adaptation of waveforms to the varying urban terrain, and combat obscurations created by urban canyons by taking advantage of spatial diversity. The simulation scenario is depicted in Figure 5.4, which shows an intersection of four city blocks. Two target trajectories are simulated using the nearly constant velocity model (NCV) on two straight line segments. Both targets move at 10 m/s; one target starts at (2150,1525) and makes a right turn at (2080,1525); the other starts at (1986,1502) and makes a left turn at (2055,1500). Three radar sensors with different viewpoints are distributed at (2050,1725), (2180,1525) and (1950,1525), represented by the symbol '□'. Receiver and transmitter locations are assumed to be the same.

Although all sensors can receive returns during the same radar scan (receivers whose corresponding transmitter is idle can still operate in passive mode), for simulation purposes we assume that only one transmitter can be active and transmit pulses at each radar scan. A waveform library consisting of four different Gaussian-windowed chirp signals is considered. Waveforms vary in pulse duration. In particular, radar sensors considered support pulse durations of 0.5 μs and 1.375 μs. In general, longer pulses return more power; however, finer details may be lost. In addition, waveforms of each pulse duration can be either an up-sweep or down-sweep chirp. Pulses are repeated at every 10 ms, and waveforms operate at 4 GHz with 40 MHz of bandwith. The maximum sensor range is 300 meters.

Each radar receiver performs matched-filtering. The received signal is a summation of reflected signals from targets and clutter, whose density is $2.5e^{-6}$ m^2 throughout the surveillance area. The multipath model assumes that the radar return is negligible after three reflections, and it can be derived using prior knowledge of the urban scene. We further assume each target is a point object, and use peak detection to locate a point source corresponding to a received power peak on a time-delay versus Doppler image, as detailed in [6].

The tracker uses an extended Kalman filter (EKF) with the NCV model for state estimation and a linear multitarget integrated probabilistic data association algorithm (LMIPDA) for data association. A total of 100 Monte Carlo runs were used to evaluate the system. Each run consists of 100 radar scans of 0.25 seconds.

5.4.2 Simulation Results

Figure 5.5 shows that the two-level scheduler outperforms the fully-myopic scheduler. In this case, both the detection risk and the receding horizon were assumed to be 10 radar scans. The performance gains obtained by the proposed method is evident when multipath and obscurations are present, since the non-myopic method is able to anticipate the future loss of observations and use this information to schedule transmitters and waveforms accordingly. Note that the position RMSE peaks when targets are farthest apart from the sensors. We can also observe that the two-level scheduler has a faster "response" to the environment, whereas the fully-myopic scheduler needs more time to recover from bad sensing decisions. In this particular simulation, the non-myopic scheduler activates a new transmitter at regular intervals of 10 time steps (according to the detection risk), except during the interval $[68, 75]$ when the non-myopic scheduler is used during each radar scan in this interval due to decreased tracking performance, which in this case means an overall position RMSE above 1.9 meters. The RMSE of both schedulers drops again when both targets get closer to one of the three sensors at the end of their respective trajectories. The waveform selected by the myopic scheduler differs from the one previously chosen by the non-myopic scheduler at radar scans 16, 36, 43, 48, 55, 59, 86 and 93. Hence, in this case the myopic scheduler improves on non-myopic decisions taken in the past approximately 8% of the time. The improvement in performance (RMSE), however, is several tens of %.

The percentage of time the non-myopic scheduler is used as a function of the detection risk is shown in Figure 5.6. Note that the transmitter activation frequency does not necessarily grow with a higher risk tolerance, since decisions made by the non-myopic scheduler could result in lower tracking accuracy such that the selection of a new set of transmitters is required.

5.5 Concluding Remarks

In this chapter, we investigated the problem of tracking targets in a hostile environment, where a central controller adaptively schedules the sequence of radar transmitters and waveforms that maximize the overall tracking accuracy, while simultaneously maintaining the sensing system's covertness. We formulated this adaptive sensing problem as a POMDP, and proposed a novel scheduling scheme were two separate schedulers are involved: a myopic waveform scheduler that operates at every radar scan, and a non-myopic transmitter scheduler that activates a new set of transmitters only when certain conditions are met, that is, if tracking performance falls below a tolerance level, or if a detection risk (i.e., risk of losing covertness) is exceeded.

Simulation results show that the proposed two-level scheduler significantly improves tracking performance when compared to a fully-myopic scheme, while still maintaining relatively low computational costs. We have shown through Monte Carlo simulations that the intelligent selection of transmitters and waveforms over time, where the long-term effects of this action are taken into account, is highly effective in tracking mobile ground targets with fixed ground radars. In addition, the proposed scheduling scheme is able to dynamically address sensor covertness. Although several simplifying assumptions were made to facilitate implementation, the intention was not necessarily to test our scheme using the most realistic scenario possible, but to demonstrate the effectiveness of the approach. Hence, simulation scenarios and parameters were chosen for illustration purposes only.

Future work could involve the study of coordinated waveform scheduling schemes, where waveforms with different interpulse and intrapulse characteristics would likely be simultaneously sent by transmitters with different viewpoints. Moreover, when multistatic radars are involved, receiver scheduling according to the scene conditions can also be considered. In these cases, innovative data fusion algorithms are needed. Given the generality of our problem formulation, it is possible to extend it to the situation of adversarial targets that can evade tracking and surveillance, and to apply sensor motion control when a mobile sensing platform is involved, in a straightforward manner. On the other hand, opportunistic sensing platforms, such as vehicular sensor networks, offer a different solution approach to the problem of target tracking in covert operations. The completely distributed, highly scalable, and opportunistic nature of such platforms makes it harder for hostiles to disable surveillance, while potentially increasing the sensing coverage area. Additional areas for further improvement of the proposed two-level controller are the analysis of different POMDP solution approximation methods (e.g., heuristic expected cost-to-go), and the study of the relationship between time horizon and tracking performance.

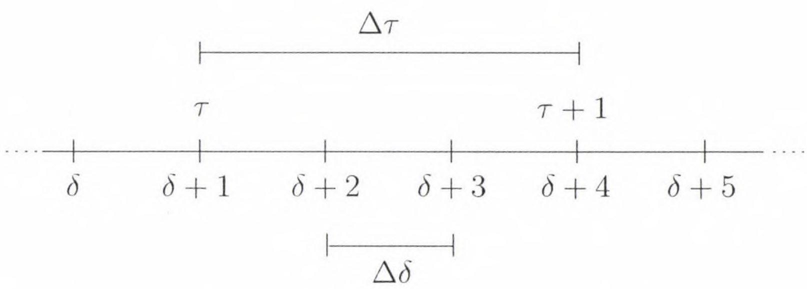

Figure 5.3: Relationship between myopic and non-myopic decision epochs for $T_D = 3$. Myopic scheduling decisions occur at δ, $\delta + 1$, $\delta + 2$, $\delta + 3$, $\delta + 4$, and $\delta + 5$; non-myopic scheduling decisions occur at τ and $\tau + 1$.

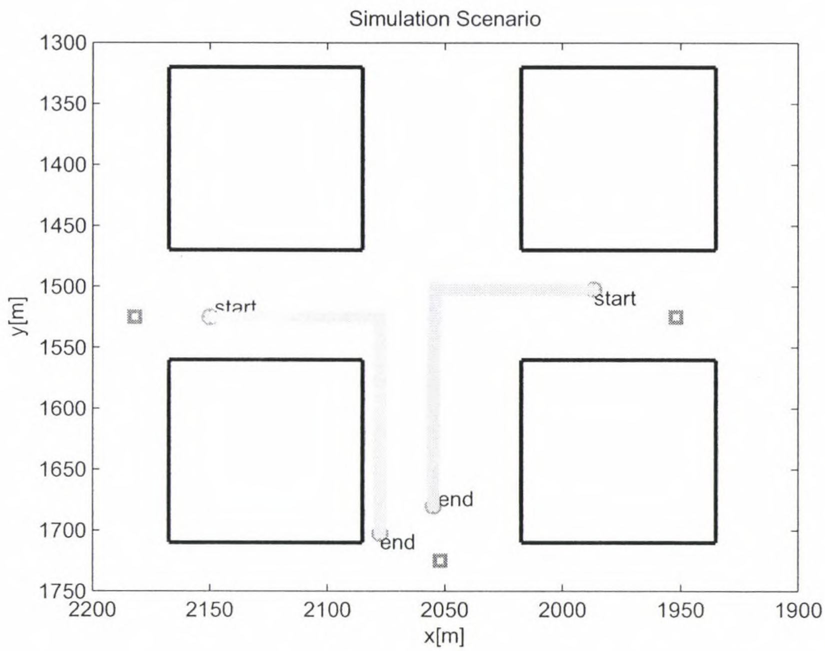

Figure 5.4: Simulation scenario with two target trajectories; sensor locations are represented by '□'.

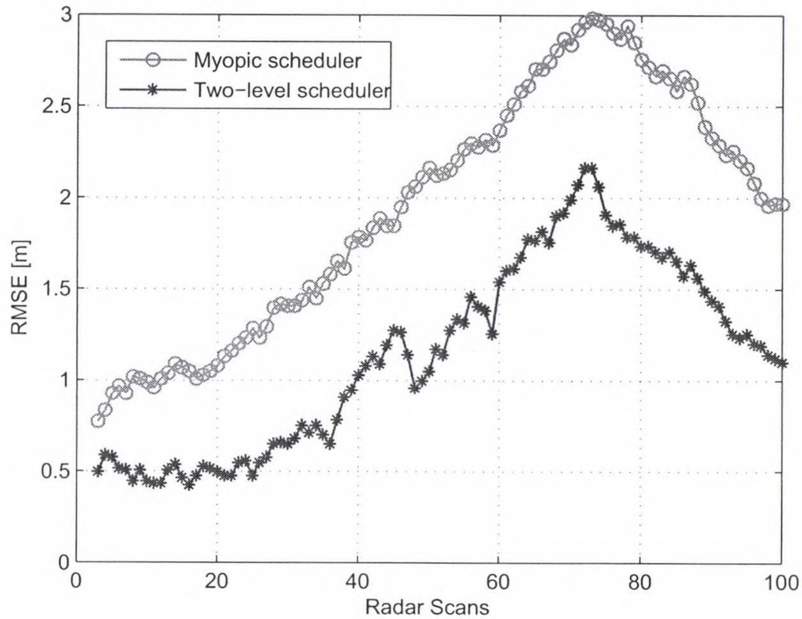

Figure 5.5: Position RMSE when using the proposed two-level scheduler or a fully-myopic scheduler.

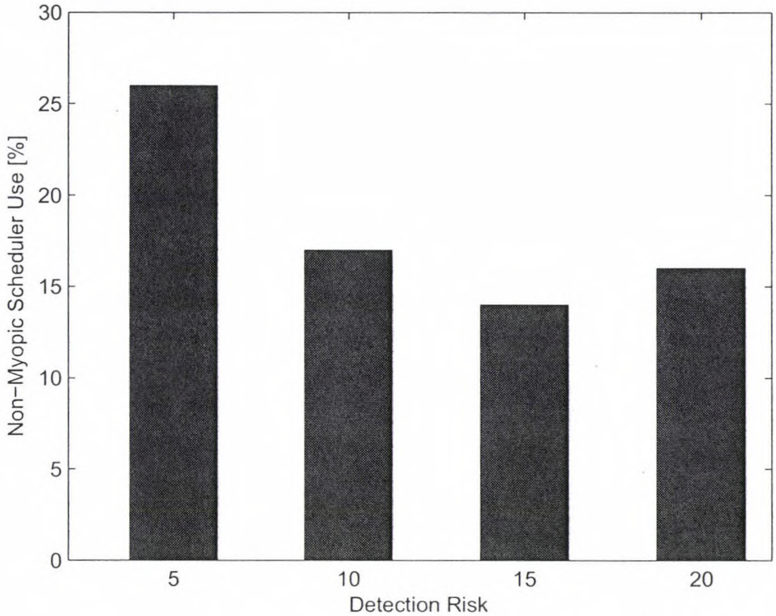

Figure 5.6: Percentage of time the non-myopic scheduler is used as a function of the detection risk.

Bibliography

[1] A. Ambainis, S. A. Bloch, and D. L. Schweizer, *Delayed binary search, or playing twenty questions with a procrastinator*, Algorithmica **32** (2002), no. 4, 641–651.

[2] T. Asano, D. Z. Chen, and N. Tokuyama, *Polynomial-time solutions to image segmentation*, Proceedings of the 7th Annual SIAM-ACM Conference on Discrete Algorithms, January 1996, pp. 104–113.

[3] J. Aslam, Z. Butler, F. Constantin, V. Crespi, G. Cybenko, and D. Rus, *Tracking a moving object with a binary sensor network*, Proceedings of ACM Conference on Embedded Networked Sensor Systems (SENSYS) (Los Angeles, California), November 2003, pp. 150–161.

[4] C. J. Baker and A. L. Hume, *Netted radar sensing*, IEEE Aerospace and Electronic Systems Magazine **18** (2003), no. 2, 3–6.

[5] Y. Bar-Shalom (ed.), *Multitarget-multisensor tracking: Advanced applications*, ch. 2, Artech House, 1990.

[6] P. R. Barbosa, E. K. P. Chong, S. Suvorova, and W. Moran, *Multitarget multi-sensor tracking in an urban environment: A closed-loop approach*, Proceedings of the Conference on Signal and Data Processing of Small Targets 2008, part of the SPIE Symposium on Defense and Security (Orlando, FL), vol. 6969, March 2008, pp. 69690W1–69690W12.

[7] L. J. Beeton and S. L. Hall, *Closed loop tracking systems for naval applications*, The IEE Seminar on Target Tracking: Algorithms and Applications (Birmingham, United Kingdom), March 2006, pp. 115–122.

[8] R. E. Bellman, *Dynamic programming*, Courier Dover Publications, 2003.

[9] D. P. Bertsekas, *Dynamic programming and optimal control*, vol. 1 & 2, Athena Scientific, 2001.

[10] P. Billingsley, *Probability and measure*, Wiley-Interscience, 1995.

[11] S. S. Blackman and R. Popoli, *Design and analysis of modern tracking systems*, Artech House, Norwood, MA, 1999.

[12] V. S. Borkar, S. K. Mitter, A. Sahai, and S. Tatikonda, *Sequential source coding: an optimization viewpoint*, Proceedings of the 44^{th} IEEE Conference on Decision and Control, and the European Control Conference (Seville, Spain), December 2005, pp. 1035–1042.

[13] N. Bulusu, J. Heidemann, and D. Estrin, *GPS-less low-cost outdoor localization for very small devices*, IEEE Personal Communications (2000), no. 5, 28–34.

[14] J. Carle and D. Simplot-Ryl, *Energy-efficient area monitoring for sensor networks*, IEEE Computer **37** (2004), no. 2, 40–46.

[15] E. Çinlar, *Introduction to stochastic processes*, Prentice-Hall, 1975.

[16] V. S. Chernyak, *Fundamentals of multisite radar systems: Multistatic radars and multiradar systems*, CRC, 1998.

[17] E. K. P. Chong, C. Kreucher, and A. O. Hero III, *Foundations and Applications of Sensor Management*, ch. 8, Springer, 2007.

[18] D. Cochran, S. Suvorova, S. Howard, and B. Moran, *Waveform libraries*, IEEE Signal Processing Magazine **26** (2009), no. 1, 12–21.

[19] T. M. Cover and J. A. Thomas, *Elements of information theory*, John Wiley & Sons, Inc., 1991.

[20] W. de Jon van Norden, F. J. Bolderheij, and L. Rothkrantz, *Intelligent task scheduling in sensor networks*, Proceedings of the 8th International Conference on Information Fusion (Philadelphia, PA), July 2005, 8 pp.

[21] A. Dhagat, P. Gacs, and P. Winkler, *On playing twenty questions with a liar*, Proceedings of the Third Annual ACM-SIAM Symposium on Discrete Algorithms (SODA) (Orlando, Florida), January 1992, pp. 16–22.

[22] J. Elson and D. Estrin, *Time synchronization for wireless sensor networks*, Proceedings of the 15^{th} International Parallel and Distributed Processing Symposium (San Francisco, California), April 2001, p. 186.

[23] P. Erdös and A. Rényi, *On two problems of information theory*, MTA Mat. Kut. Int. Kozl. **8A** (1963), 229–243.

[24] E. Ertin, J. Fisher, and L. Potter, *Maximum mutual information principle for dynamic sensor query problems*, Proceedings of IPSN'03 (Palo Alto, CA), April 2003, pp. 405–416.

[25] R. Evans, V. Krishnamurthy, G. Nair, and L. Sciacca, *Networked sensor management and data rate control for tracking maneuvering targets*, IEEE Transactions on Signal Processing **53** (2005), no. 6, 1979–1991.

[26] M. E. Farmer, R-L. Hsu, and A. K. Jain, *Interacting multiple model (IMM) Kalman filters for robust high speed human motion tracking*, Proceedings of the 2002 International Conference on Pattern Recognition, August 2002, pp. 20–23.

[27] E. Fishler, A. Haimovich, R. Blum, D. Chizhik, L. Cimini, and R. Valenzuela, *MIMO radar: an idea whose time has come*, Proceedings of the IEEE Radar Conference, April 2004, pp. 71–78.

[28] S. Ganeriwal, R. Kumar, and M. Srivastava, *Timing-sync protocol for sensor networks*, Proceedings of ACM Conference on Embedded Networked Sensor Systems (SENSYS) (Los Angeles, California), November 2003, pp. 138–149.

[29] D. Ganesan, R. Govindan, S. Shenker, and D. Estrin, *Highly-resilient, energy-efficient multipath routing in wireless sensor networks*, ACM SIGMOBILE Mobile Computing and Communications Review **5** (2001), no. 4, 11–25.

[30] J. R. Guerci, *Space-time adaptive processing for radar*, Artech House, 2003.

[31] J. R. Guerci and E. J. Baranoski, *Knowledge-aided adaptive radar at DARPA*, IEEE Signal Processing Magazine (2006), 41–50.

[32] E. Hanle, *Survey of bistatic and multistatic radar*, IEE Proceedings, Part F (1986), 587–595.

[33] Y. He and E. K. P. Chong, *Sensor scheduling for target tracking: A Monte Carlo sampling approach*, Digital Signal Processing **16** (2006), no. 5, 533–545.

[34] R. Hill, J. Karim, and E. Berlekamp, *The solution of a problem of ulam on searching with lies*, Proceedings of the IEEE International Symposium on Information Theory (ISIT) (Cambridge, Massachusetts), August 1998, pp. 16–21.

[35] R. Iyengar and B. Sikdar, *Scalable and distributed GPS free positioning for sensor networks*, Proceedings of IEEE International Conference on Communications(ICC) (Anchorage, Alaska), May 2003, pp. 338–342.

[36] S. J. Julier and J. K. Uhlmann, *Unscented filtering and nonlinear estimation*, Proceedings of the IEEE **92** (2004), no. 3, 401–422.

[37] V. Kalashnikov, *Topics on regenerative processes*, CRC Press, 1994.

[38] D. J. Kershaw and R. J. Evans, *Optimal waveform selection for tracking systems*, IEEE Transactions on Information Theory **40** (1994), no. 5, 1536–1550.

[39] Y. S. Kim and K. S. Hong, *An IMM algorithm for tracking maneuvering vehicle in an adaptive cruise control environment*, International Journal on Control and Automatic Systems **2** (2004), no. 3, 310–318.

[40] T. Kirubarajan and Y. Bar-Shalom, *Tracking evasive move-stop-move targets with an MTI radar using a VS-IMM estimator*, Proceedings of SPIE Signal and Data Processing of Small Targets, vol. 4048, 2000, pp. 236–246.

[41] T. Kirubarajan, Y. Bar-Shalom, W. D. Blair, and G. A. Watson, *IMMPDA solution to benchmark for radar resource alllocation and tracking in the presence of ECM*, IEEE Transactions on Aerospace and Electronic Systems **34** (1998), 1023–1036.

[42] T. Kirubarajan, Y. Bar-Shalom, K. R. Pattipatti, and I. Kadar, *Ground target tracking with variable structure IMM estimator*, IEEE Transactions on Aerospace and Electronic Systems **36** (2000), 26–45.

[43] D. Knuth, *The art of computer programming, vol. 3: Sorting and searching*, Addison-Wesley, 1997.

[44] L. L. M. I. Kontoyiannis and S. Meyn, *Relative entropy and exponential deviation bounds for general markov chains*, Proceedings of the IEEE International Symposium on Information Theory (ISIT) (Adelaide, Australia), December 2005, pp. 1563–1567.

[45] C. Kreucher, D. Blatt, A. Hero, and K. Kastella, *Adaptive multi-modality sensor scheduling for detection and tracking of smart targets*, Digital Signal Processing **16** (2006), 546–567.

[46] C. Kreucher and A. Hero, *Non-myopic approaches to scheduling agile sensors for multitarget detection, tracking, and identification*, Proceedings of International Conference on Acoustics, Speech, and Signal Processing, vol. 5, March 2005, pp. 885–888.

[47] C. M. Kreucher, A. O. Hero, K. D. Kastella, and M. R. Morelande, *An information-based approach to sensor management in large dynamic networks*, Proceedings of the IEEE **95** (2007), no. 5, 978–999.

[48] V. Krishnamurthy, *Algorithms for optimal scheduling of hidden Markov model sensors*, IEEE Transactions on Signal Processing **50** (2002), 1382–1397.

[49] V. Krishnamurthy and D. Evans, *Hidden markov model multiarm bandits: A methodology for beam scheduling in multitarget tracking*, IEEE Transactions on Signal Processing **49** (2001), 2893–2908.

[50] U. Lee and M. Gerla, *A survey of urban vehicular sensing platforms*, Computer Networks **54** (2010), no. 4, 527–544.

[51] U. Lee, E. Magistretti, B. Zhou, M. Gerla, P. Bellavista, and A. Corradi, *MobEyes: smart mobs for urban monitoring with vehicular sensor networks*, IEEE Wireless Communications **13** (2006), no. 5, 51–57.

[52] H. Li, *Information theoretic problems in decision making and networks*, Ph.D. thesis, 2007.

[53] X. R. Li and V. P. Jilkov, *A survey of maneuvering target tracking: dynamic models*, Proceedings of the SPIE Conference on Signal and Data Processing of Small Targets (Orlando, FL), vol. 4048, April 2000, pp. 212–235.

[54] ______, *Survey of maneuvering target tracking. Part I. Dynamic models.*, IEEE Transactions on Aerospace and Electronic Systems **39** (2003), no. 4, 1333–1364.

[55] ______, *A survey of maneuvering target tracking. Part V. Multiple-model methods*, IEEE Transactions on Aerospace and Electronic Systems **41** (2005), no. 4, 1255–1321.

[56] X. Rong Li and Y. Bar-Shalom, *Tracking in clutter with nearest neighbor filters: analysis and performance*, IEEE Transactions on Aerospace and Electronic Systems **32** (1996), no. 3, 995–1010.

[57] Y. Li, W. Krakow, E. K. P. Chong, and K. N. Groom, *Approximate stochastic dynamic programming for sensor scheduling to track multiple targets*, Digital Signal Processing **19** (2009), no. 6, 978–989.

[58] J. Liu, P. Cheung, L. Guibas, and F. Zhao, *A dual-space approach to tracking and sensor management in wireless sensor networks*, Proceedings of the 1^{st} ACM International Workshop on Wireless Sensor Networks and Applications (Atlanta, Georgia), April 2002, pp. 131–139.

[59] J. Liu, J. Reich, and F. Zhao, *Collaborative in-network processing for target tracking*, EURASIP JASP: Special Issues on Sensor Networks **2003** (2003), no. 4, 378–391.

[60] J. Manyika and H. Durrant-Whyte, *Data fusion and sensor management: a descentralized information-theoretic approach*, Ellis Horwood, New York, 1994.

[61] W. L. Melvin, *Space-time adaptive radar performance in heterogeneous clutter*, IEEE Transactions on Aerospace Electronic Systems **36** (2000), no. 2, 621–633.

[62] S. Meyn and R. Tweedie, *Markov chain and stochastic stability*, Communications and Control Engineering Series, Springer-Verlag, 1993.

[63] S. A. Miller, Z. A. Harris, and E. K. P. Chong, *A POMDP framework for coordinated guidance of autonomous uavs for multitarget tracking*, EURASIP Journal on Applied Signal Processing, special issue on Signal Processing Advances in Robots and Autonomy **2009** (2009), Article ID 724597, 17 pp.

[64] H. Mine and S. Osaki, *Markovian decision processes*, Elsevier, 1970.

[65] S. Miranda, C. Baker, K. Woodbridge, and H. Griffiths, *Knowledge-based resource management for multifunction radar*, IEEE Signal Processing Magazine **23** (2006), no. 1, 66–76.

[66] D. Mušicki, S. Challa, and R. Evans, *Multitarget tracking in clutter without measurement assignment*, Proceedings of the 44^{th} IEEE Conference on Decision and Control (Paradise Island, Bahamas), December 2004, pp. 716–721.

[67] D. Mušicki, R. Evans, and S. Stanković, *Integrated probabilistic data association (IPDA)*, IEEE Transactions on Automatic Control **39** (1994), no. 6, 1237–1241.

[68] D. Mušicki, S. Suvorova, and S. Challa, *Multi target tracking of ground targets in clutter with LMIPDA-IMM*, Proc. of The 7^{th} International Conference on Information Fusion (Stockholm, Sweden), June 2004, pp. 1104–1110.

[69] R. Niu, P. Willet, and Y. Bar-Shalom, *System level performance for radar waveforms*, Proceedings of the 7^{th} Mediterranean Conference on Control and Automation (MED 99) (1999), 868–897.

[70] A. Pelc, *Searching with known probability of error*, Theoretical Computer Science **63** (1989), 185–202.

[71] J. Proakis, *Digital communications*, 4^{th} ed., McGraw Hill, 2001.

[72] M. L. Putterman, *Markov decision processes: Discrete stochastic dynamic programming*, Wiley, 1994.

[73] A. Rényi, *On a problem of information theory*, MTA Mat. Kut. Int. Kozl. **6B** (1961), 505–516.

[74] R. L. Rivest, A. R. Meyer, D. J. Kleitman, K. Winkimann, and J. Spencer, *Coping with errors in binary search procedures*, Journal of Computer and System Sciences **20** (1980), no. 3, 396–404.

[75] B. Robinson and D. Sasaki, *Software systems testing of a closed loop tracking system using a simulink based simulation*, vol. 4025, April 2000, pp. 160–169.

[76] S. M. Ross, *Applied probability models with optimization applications*, Holden-Day, 1970.

[77] A. Sahai and S. Mitter, *The necessity and sufficiency of anytime capacity for stabilization of a linear system over a noisy communication link - Part I: Scalar systems*, IEEE Transactions on Information Theory **52** (2006), 3369–3395.

[78] J. N. Sanders-Reed, *Multi-target, multi-sensor, closed loop tracking*, Proceedings of SPIE XVIII Acquisition, Tracking, and Pointing (Orlando, FL), vol. 5430, April 2004, pp. 160–169.

[79] T. Sathyan, K. Bharadwaj, A. Sinha, and T. Kirubarajan, *Intelligence-aided multitarget tracking for urban operations. a case study: Counter-terrorism*, Proceedings of SPIE **6201** (2006), 62010M.

[80] D. S. Sawicki, *The traffic radar handbook*, Grove Enterprises, 1993.

[81] S. Suvorova, D. Mušicki, B. Moran, S. Howard, and B. La Scala, *Multi step ahead beam and waveform scheduling for tracking of manoeuvering targets in clutter*, Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 5, March 2005, pp. 889–892.

[82] A. S. Tanenbaum and A. S. Woodhull, *Operating systems: Design and implementation*, 2^{nd} ed., Prentice Hall, 1997.

[83] S. Tatikonda and S. Mitter, *Control over noisy channels*, IEEE Transactions on Automatic Control **49** (2004), no. 7, 1196–1201.

[84] ______, *Control under communication constraints*, IEEE Transactions on Automatic Control **49** (2004), no. 7, 1056–1068.

[85] S. Tatikonda, A. Sahai, and S. Mitter, *Control of LQG systems under communication constraints*, Proceedings of the 37^{th} IEEE Conference on Decision and Control (Tampa, FL), vol. 1, December 1998, pp. 1165–1170.

[86] ———, *Control of LQG systems under communication constraints*, Proceedings of the 1999 American Control Conference (San Diego, CA), vol. 4, June 1999, pp. 2778–2782.

[87] D. Tian and N. Georganas, *Energy efficient routing with guaranteed delivery in wireless sensor networks*, Proceedings of IEEE Wireless Communications and Networking Conference(WCNC) (New Orleans, Louisiana), March 2003, pp. 1923–1929.

[88] H. L. Van Trees, *Detection, estimation, and modulation theory*, vol. 3, Wiley Interscience, 2001.

[89] S. M. Ulam, *Adventures of a mathematician*, Scribner, New York, 1976.

[90] J. van Kleef, J. Bergmans, L. Kester, and F. Groen, *Multiple-hypothesis trilateration and tracking with distributed radars*, Proceedings of the 9^{th} International Conference on Information Fusion, 2006. ICIF '06., July 2006, pp. 1–7.

[91] H. Wang, K. Yao, G. Pottie, and D. Estrin, *Entropy-based sensor selection heuristic for target localization*, Proceedings of the 3rd international symposium on Information processing in sensor networks (Berkeley, CA), April 2004, pp. 36–45.

[92] H. L. Wu and A. Cameron, *A Bayesian decision theoretic approach for adaptive goal-directed sensing*, Proceedings of the Third International Conference on Computer Vision (Osaka, Japan), December 1990, pp. 563–567.

[93] R. W. Yeung, *A first course in information theory*, Kluwer Academic, 2002.

[94] F. Zhao, J. Shin, and J. Reich, *Information-driven dynamic sensor collaboration*, IEEE Signal Processing Magazine **19** (2002), no. 2, 61–72.