Urban Terrain Multiple Target Tracking Using The Probability Hypothesis Density Particle Filter

by

Meng Zhou

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved July 2011 by the
Graduate Supervisory Committee:

Antonia Papandreou-Suppappola, Chair
Cihan Tepedelenlioglu
Narayan Kovvali

ARIZONA STATE UNIVERSITY

August 2011

ABSTRACT

The tracking of multiple targets becomes more challenging in complex environments due to the additional degrees of nonlinearity in the measurement model. In urban terrain, for example, there are multiple reflection path measurements that need to be exploited since line-of-sight observations are not always available. Multiple target tracking in urban terrain environments is traditionally implemented using sequential Monte Carlo filtering algorithms and data association techniques. However, data association techniques can be computationally intensive and require very strict conditions for efficient performance.

This thesis investigates the probability hypothesis density (PHD) method for tracking multiple targets in urban environments. The PHD is based on the theory of random finite sets and it is implemented using the particle filter. Unlike data association methods, it can be used to estimate the number of targets as well as their corresponding tracks. A modified maximum-likelihood version of the PHD (MPHD) is proposed to automatically and adaptively estimate the measurement types available at each time step. Specifically, the MPHD allows measurement-to-nonlinearity associations such that the best matched measurement can be used at each time step, resulting in improved radar coverage and scene visibility. Numerical simulations demonstrate the effectiveness of the MPHD in improving tracking ability, both for tracking multiple targets and targets in clutter.

To my parents.

ACKNOWLEDGEMENTS

First and foremost, I want to express my utmost gratitude to my advisor, Professor Antonia Papandreou-Suppappola, for her invaluable guidance, motivation, and support. It has been an honor and a pleasure working with her for the past one years. She has a passion for signal processing that is only outweighed by her passion for teaching, and without her assistance, insight, and encouragement, this work would not have been possible. I would also like to express my thanks to Professor Cihan and Dr. Narayan for their willingness to sit on my supervisory committee. I am grateful to them for their interest in my work and their valuable suggestions and insights. Further, I would like to thank Jun Jason Zhang for his assistance in my research and his insight into the real world considerations of target tracking in a dense urban environment. Additionally, I would like to thank Lakshminarayan Ravichandran for all his help. Finally, I would like to thank all my friends and colleagues in the Signal Processing and Adaptive Sensing Lab for all the help and encouragement they gave me during my studies. Specifically, I want to thank Jun Jason Zhang and Bhavana Chakraborty for their help when I was just beginning my research.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1.1 Research Motivation

Target tracking in radar involves monitoring the position and movement of a target at every time step. As the tracking problem can be modeled using dynamic state and measurement equations, stochastic filtering techniques can be used to estimate the unknown target position and velocity states. When the model equations are linear and the model error and noise processes are Gaussian, then the Kalman filter estimation technique can be used [1–3]. When the model equations are nonlinear, the extended Kalman filter can be used, that linearizes the model equations using approximation techniques such as the Taylor series expansion [2]. Recently, sequential Monte Carlo techniques such as particle filtering were proposed to estimate the unknown state when the dynamic state model and measurement model are nonlinear and/or the random processes are non-Gaussian [2]. The particle filter (PF) uses Monte Carlo simulations to implement a recursive Bayesian filter. In particular, it estimates the posterior probability density function using a fixed number of particles and corresponding weights. When there are multiple targets, the tracking becomes more complicated. This is because, in every iteration, the tracker must first figure out the number of targets before applying the data to the filter. In addition, data association techniques need to be used as it is not known which measurement corresponds to which target [4].

In dense urban environments, most conventional radar tracking systems begin to fail due to the absence of line-of-sight returns, and the presence of multipath interference, obscuration from buildings, and high clutter [5–7]. As a result, radar tracking systems originally designed for operations in open environments need to be modified and improved to adapt to the characteristics of dense urban environments.

There has been a lot of research work on target tracking in urban environments during the past decade [5–11]. One of the main ideas proposed is to exploit multipath

returns instead of treating them as interference and mitigating them [12]. In particular, target information can be extracted from multipath returns when no line-of-sight (LOS) returns are available or multipath returns can be combined with LOS returns to provide diversity in extracting target information and thus enhance tracking performance [6–8]. This multipath exploitation radar (MER) approach utilizes prior knowledge about the environment, such as locations of buildings and road maps [10].

Recently, waveform agile sensing has been integrated with multipath exploitation to further improve tracking performance in urban terrain [11, 13–15]. Specifically, for a single target scenario, the agile sensing algorithm selects the parameters of the waveform to be transmitted at the next time step by minimizing the predicted tracking error covariance using a particle filter tracker with nonlinear measurements. The improved tracking performance was demonstrated for perfect detection [11, 13] and for high clutter scenarios [13, 15].

Another effective approach for tracking moving targets in urban terrain is by using multiple-input and multiple-output (MIMO) radar systems. One of the main advantages of using MIMO radar systems is that each radar sensor can transmit a different waveform [16]. Two different types of MIMO radar systems have been investigated: MIMO radar systems with collocated antennas and with widely-separated antennas. MIMO radar systems with collocated antennas use diverse beam patterns and have been shown to increase radar performance in detection and parameter identification applications [17–20], whereas MIMO radar systems with widely-separated antennas have been shown to provide high diversity gain [16, 21, 22]. More details on MIMO radar systems can be found in [23]. In urban environment applications, a method was proposed in [15] that maximizes the target information using an optimal configuration of MIMO widely-separated radar sensor while exploiting multipath returns from all the sensors. The MIMO waveform parameters were also adaptively configured at each time step in order to minimize the overall mean-squared tracking error.

The presence of clutter in urban environment makes tracking an even more challenging problem due to possible observed multipath returns. Clutter can be separated from true measurements using classical data association techniques that associate the correct measurements to the urban tracks before applying Bayesian filtering approaches to perform tracking. Multipath data association was used with over-the-horizon radar (OTHR) in [24] to initiate and track nonmaneuvering targets with constant probability of detection. It was also used in [25] to find the LOS measurement mitigating non-LOS measurements.

Tracking multiple objects arises in various applications, such as robotics, signal processing and medicine [4, 26–30]. The problem of multiple target tracking is to instantaneously estimate both the number of targets present as well as each target's trajectory. The estimation problem needs to take into account the interference from clutter and from measurements generated from other targets. Conventional multiple target tracking filtering techniques first couple the correct measurement to existing tracks through *measurement-to-track* associations, and then they estimate the target states using techniques originally designed for tracking single targets [4, 26, 31–34]. The methods are similar to those discussed for clutter, since clutter can be considered a special case of false alarm targets. In most cases for targets, only one measurement is assumed from each target. If several returns are available from each target, then different data association methods would need to be considered [35, 36]. One of the simplest and computational inexpensive data association methods is the nearest neighbor (NN), which selects the measurement closest to the track if it falls within a specific gate, and it fails as the clutter density increases and more than one measurement falls within one gate. The most general data association method that overcomes this problem is the multiple hypothesis tracking (MHT) approach [26, 27, 34]. The MHT first performs data association on a sequence of measurements and then filters each data association hypothesis. However, as the MHT is an exhaustive approach, it is very computation-

ally expensive [37]. As an alternative, the joint probabilistic data association (JPDA) method estimates the states by summing all the association hypothesis weighted by the probabilities from the likelihood [38, 39]. Specifically, it considers associations that survive after gating and combines those associations according to their likelihood. The JPDA is computationally less intensive [4, 40], however it requires that the number of targets is fixed and its performance is poor when the targets are close to each other [26, 27].

The problems of estimating the number of targets as they change with time and distinguishing targets in close proximity are very critical in a multiple target tracking application. One approach toward solving these problem is the use of random finite set (RFS) theory [29, 41–43]: treating multiple target states as set-valued entities, and propagating them using the Bayesian framework. Following the RFS theory, the probability hypothesis density filter (PHDF) was proposed in [44] as a suboptimal but computationally tractable algorithm for multiple target tracking [42, 44–48]. Two closed-form solutions, the sequential Monte Carlo probability hypothesis density filter (SMC-PHDF) and the Gaussian-mixture probability hypothesis density filter (GM-PHDF) were proposed in [49, 50] and shown to provide good performance for multiple target tracking under different assumptions. The particle probability hypothesis density filter (PPHDF) can reason and determine the number of targets at each time step. Specifically, in [48, 50–54], the PPHDF was shown to perform well in multiple target tracking problems. Using the Wasserstein distance as a multiple target tracking performance metric, the PPHDF outperformed the MHT filter in [55]. It was also demonstrated to achieve a substantial reduction in computation and memory requirements when compared with data association based approaches [56].

1.2 Multiple Target Tracking in Urban Environment

One of the most difficult problems in multiple target tracking is the time-varying number of targets. Due to spontaneous target birth or target spawning, new targets may

appear. Meanwhile a target can disappear in the surveillance region if the existing targets do not survive. For a multiple target tracking problem, the duration for which a target lasts is unknown. Moreover, the received signal from a radar sensor may not have measurements from all the targets, i.e., some measurements from targets could be missing. In addition, the sensor signal can also contain measurements from clutter and thus not generated by targets. It is also possible that measurements are indistinguishable from one another and hence there is no way of knowing which measurement is generated by a target or clutter. In this case, the multiple target tracking problem involves jointly estimating the number of targets as well as the states of the targets. This problem is especially more difficult in urban environments, when the measurements obtained are mostly multipath returns and often targets are in shadow regions.

The motivation of the research work in this thesis is to investigate the multiple target tracking problem in urban environments by efficiently estimating target states using particle probability hypothesis density filters (PPHDF). The proposed approach aims to track multiple targets moving between dense buildings, where we characterize the dynamic state model of the moving targets using interactive multiple models. This is because the targets are assumed to either move in nearly constant velocity or using coordinated turns. The measurement model is highly nonlinear as it is designed to take into consideration any possible finite set of available measurements at any given time. A possible measurement could include any combination of one, two or three of the following signal returns: LOS return, one multipath return, two multipath returns. It can also include the no measurement due to shadowing regions. The PHDF in [44] can dynamically estimate the total number of targets as the integration of PHD in any region of state space is the expected number of targets contained in that region. Also the urban multiple state model can be solved by using the multi-model estimator method [1] to achieve robustness and improved tracking performance when targets exhibit different kinetic patterns at different times.

In most urban tracking literature, the assumption was made that the type of return paths at each time step were known a priori. However, this assumption cannot be made in real applications as it requires knowledge of the true target position and urban scene geometry. Thus, in this thesis, we propose a modified PPHDF (MPPHDF) that allows measurement-to-path data associations such that the best matched measurement return path can be used at each time step. In particular, the new MPPHDF algorithm models the relationship between the measurements and the various path returns as data associations by utilizing prior knowledge about the target position relative to the transmitting antenna in the urban environment and about the propagation and scattering of the electromagnetic waves. Using the MPPHDF algorithm, we can automatically and adaptively estimate the measurement multipath scenario at any given time. Specifically, we compute the most likely measurement path return using the current measurement, the predicted target state based on the measurements, and the prior geometry information. Then, the target state is sequentially estimated at each time step based on the measurements and predicted type of returns.

1.3 Thesis Organization

This thesis is organized as follows. Chapter 2 provides the state-space model and the formulation of target tracking in urban environments. Chapter 3 presents conventional multiple target tracking techniques and PPHDF filtering algorithm. Chapter 4 investigates the use of the PPHDF for tracking multiple targets in urban environments and proposes the MPPHDF as well as the simulation results of the aforementioned techniques.

Chapter 2

Target Tracking in Urban Environments

2.1 Target State Model for Urban Environments

Conventional target tracking algorithms have been originally designed for use in open, unobscured areas. However, with warfare moving to rural obscured areas, tracking formulation and algorithms need to be modified in order to adapt to the characteristics of urban environments.

The dynamic state space formulation approach has been effectively used in target tracking [2,3] to describe the state and measurement equations [1]. In our study, the target tracking is performed in three-dimensional (3-D) space. The state of a target is characterized by the vector $\mathbf{x}_k = [x_k, \dot{x}_k, y_k, \dot{y}_k, z_k, \dot{z}_k]^T$, that provides the target's position (x_k, y_k, z_k) and velocity $(\dot{x}_k, \dot{y}_k, \dot{z}_k)$ in Cartesian coordinates, where k is the time index and T denotes vector transpose.

For target tracking in urban environments, the dynamic state model is assumed to be linear, and the target state $\mathbf{x}_k$ at time step k is given by

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{n}_k, \tag{2.1}$$

where $\mathbf{n}_k$ is the modeling random process that is assumed to additive Gaussian and $\mathbf{F}$ is a matrix that describes the target state evolution. In urban environments, one motion model is not sufficient to describe the moving target state as, for example, the target may be moving in a straight line at a constant velocity before taking a sharp turn. Taking this into consideration, robust state equations are comprised of a combination of different maneuvering models, including moving in a straight line, turning, accelerating, and stopping. In this thesis, two target state models are considered [57]. The first model is the nearly constant velocity (NCV) model, with state transfer matrix $\mathbf{F}$ in (2.1) given

by

$$\mathbf{F} = \begin{bmatrix} 1 & \delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \delta t \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

where δt is the time interval between successive measurements.

The second state model is the coordinated turn (CT) model that assumes that a target turns left or right with nearly constant velocity and nearly constant angular turning rate. The state transfer matrix $\mathbf{F}$ in (2.1) for the CT model is given by

$$\mathbf{F} = \begin{bmatrix} 1 & \sin(\omega\delta t)/\omega & 0 & -(1-\cos(\omega\delta t))/\omega \\ 0 & \cos(\omega\delta t) & 0 & -\sin(\omega\delta t) \\ 0 & (1-\cos(\omega\delta t))/\omega & 1 & \sin(\omega\delta t)/\omega \\ 0 & \sin(\omega\delta t) & 0 & \cos(\omega\delta t) \end{bmatrix},$$

where $\omega \neq 0$ is the angular turning rate, which is assumed known. The modeling random process for both the NCV model and the CT model are assumed to have a zero-mean Gaussian distribution with covariance matrix

$$\mathbf{Q} = q \begin{bmatrix} \frac{\delta t^3}{3} & \frac{\delta t^2}{2} & 0 & 0 \\ \frac{\delta t^2}{2} & \delta t & 0 & 0 \\ 0 & 0 & \frac{\delta t^3}{3} & \frac{\delta t^2}{2} \\ 0 & 0 & \frac{\delta t^2}{2} & \delta t \end{bmatrix},$$

where q is a coefficient that determines the process intensity.

2.2 Measurement Model for Urban Environments

After detection using the received signals at the receiver, the matched filter output provides an observation vector $\mathbf{z}_k$ which can obtain range and rang-rate measurements that have originated from the true targets or from false alarms due to clutter [8]. In urban environments, the received radar signal consists of line-of-sight (LOS) returns as well as reflections off surrounding buildings. Traditionally, these multipath return

signals are considered to be interference, and methods are applied to mitigate them. However, as LOS returns are not always available, techniques have been developed to model multipath propagation and thus exploit multipath returns by extracting target information from them. Specifically, by making use of prior knowledge of road maps and building geometry information, multipath signals can be utilized to enhance tracking performance.

LOS returns are obtained when the transmitted signal reaches the target and is reflected back directly to the radar receiver. An example urban scene of a target moving

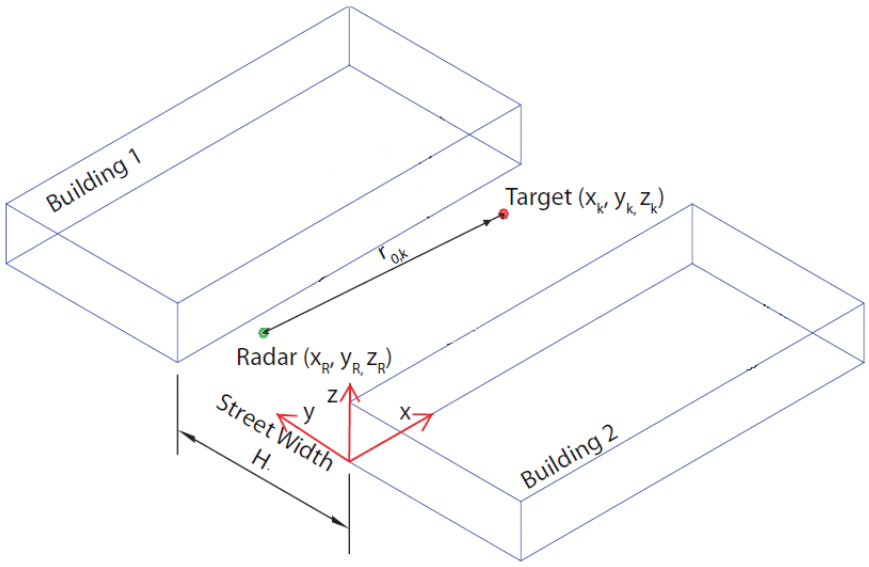

Figure 2.1: 3-D geometry of an LOS path in an urban scene for a target moving between two buildings

between two buildings with an LOS path is depicted in Fig. 2.1, the LOS path is denoted by $r_{0,K}$ and the street width is H. The radar receiver is located at (x_R, y_R, z_R), and the transmitter and receiver are assumed to be stationary and collocated. The location and velocity of the target at time k are given by (x_k, y_k, z_k) and $(\dot{x}_k, \dot{y}_k, \dot{z}_k)$, respectively.

The measurement equation corresponding to the LOS return path depends on the range and range-rate between the target and the radar. The range is thus given by

$$r_{o,k} = ((x_k - x_R)^2 + (y_k - y_R)^2 + (z_k - z_R)^2)^{1/2}, \tag{2.2}$$

and, the range-rate can be computed by taking the derivative of the range with respect to time to obtain

$$\dot{r}_{o,k} = (\dot{x_k}(x_k - x_R) + \dot{y_k}(y_k - y_R) + \dot{z_k}(z_k - z_R))/r_{o,k}. \tag{2.3}$$

Note that we often assume that the height of the target and radar, in urban terrain applications, are assumed to remain constant over all time; thus the target velocity in the z-direction is often assumed to be zero.

The measurement model for multipath returns is more complicated to formulate than for LOS returns. We first assume that the walls of the buildings are perfectly smooth so that all reflections can be assumed to be specular reflections, i.e., for all the reflection rays, angle of incidence equals the angle of reflection. We also assume that every reflection on a building wall introduces about 20 dB loss in signal energy, this is demonstrated in [9]. Following this assumption, multipath returns from more than two bounces can be ignored as the reflected signal strength is too weak to be detected by the radar. As a result, in this thesis, we only consider signal reflection paths of upto two bounces. Note that when the multipath signal returns are observed, the radar system interprets the time delay as range between the radar and a virtual target, as shown in Fig. 2.2. Fig. 2.2 demonstrates the scenario of single-bounce multipath signal returns.

The measurement equation in terms of range and range-rate as a function of multipath returns is given as follows. The range from the radar to the target after m bounces off Building i is given by

$$r_{m,k,i} = ((x_k - x_R)^2 + [(-1)^{m+1}(2[\frac{m}{2}]_i H - (-1)^{i+1} y_k) - y_R]^2 + (z_k - z_R)^2)^{1/2}, \tag{2.4}$$

where H is the street width. It is assumed that the first bounce was off Building i, $i = 1, 2$, where $[\frac{m}{2}]_1 = \lceil \frac{m}{2} \rceil$ and $[\frac{m}{2}]_2 = \lfloor \frac{m}{2} \rfloor$ [11]. The corresponding range-rate is given

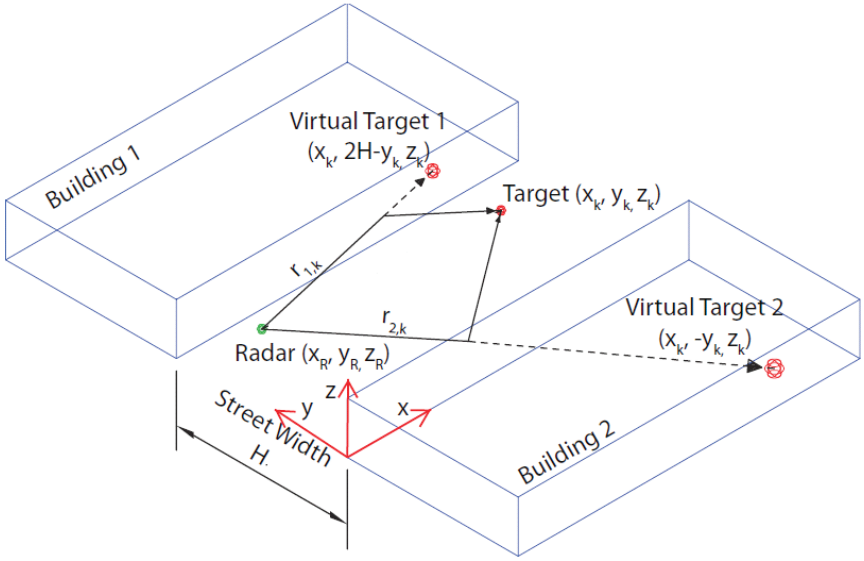

Figure 2.2: 3-D geometry of multipath returns in an urban scene for a target moving between two buildings.

by the derivative of the range in (2.4) with respect to time,

$$\begin{aligned}\dot{r}_{m,k,i} &= \dot{x}_k(x_k - x_R)/r_{i,k} + \dot{y}_k(-1)^{m+i+1}[(-1)^{m+1}(2[\frac{m}{2}]_i H - (-1)^{i+1}y_k) - y_R]/r_{i,k} \\ &+ \dot{z}_k(z_k - z_R)/r_{i,k}. \end{aligned} \tag{2.5}$$

When shadowing or obscuring occurs, no return can be observed by the radar receiver. This situation occurs when the targets travel behind buildings and no LOS or multipath returns form.

The number of range and range-rate measurements depends on both the geometry of the environment and the target location within the environment. Since the target's location is changing with time, we let P_k to be the number of available measurement at time k. The measurement model is thus given by

$$\mathbf{z}_k = h_k(\mathbf{x}_k) + \mathbf{w}_k, \tag{2.6}$$

where $\mathbf{w}_k$ is a zero-mean Gaussian noise process with covariance matrix R_k, and $h_k(\mathbf{x}_k)$ is a matrix of range and range-rate measurements given by

$$h_k(\mathbf{x}_k) = \begin{bmatrix} r_{0,k} & r_{1,k} & \cdots & r_{P_k,k} \\ \dot{r}_{0,k} & \dot{r}_{1,k} & \cdots & \dot{r}_{P_k,k} \end{bmatrix}, \tag{2.7}$$

where $r_{p,k}$ and $\dot{r}_{p,k}$ are the range and range-rate measurements of the pth path at time step k, $p = 1, \cdots, P_k$, respectively, and P_k is the total number of paths at time step k. In this case, the measurement $\mathbf{z}_k$ is given by

$$\mathbf{z}_k = \begin{bmatrix} \mathtt{r}_{0,k} & \mathtt{r}_{1,k} & \cdots & \mathtt{r}_{P_k,k} \\ \dot{\mathtt{r}}_{0,k} & \dot{\mathtt{r}}_{1,k} & \cdots & \dot{\mathtt{r}}_{P_k,k} \end{bmatrix}, \tag{2.8}$$

where $\mathtt{r}_{p,k}$ and $\dot{\mathtt{r}}_{p,k}$ are the noisy version of range and range-rate measurements of the pth path at time step k, $p = 1, \cdots, P_k$, respectively, and P_k is the total number of paths at time step k.

2.3 Clutter Model

The received measurements are assumed to be either target returns (LOS or multipath) or false alarms due to clutter. In order to model the probability density function of the number of false alarms received, the parametric Poisson model is used with spatial density ρ [8]. We consider a validation region of volume V_k at time k, which is a region in the measurement space in which the true measurement will exist with some high probability. Detections within the validation region are associated with the target of interest and are considered valid whereas the remaining detections are discarded. Inside the validation region, the clutter is assumed to be distributed uniformly. Then, the probability that m false alarms are obtained at time k is given by We assume that the number of false alarms follows a Poisson distribution with average ρV, where ρ is the

$$Pr(m_k = m) = \frac{\exp(-\rho V_k)(\rho V_k)^{m_k}}{m_k!}. \tag{2.9}$$

For urban environments, the probability of detection, P_{D}, dynamically varies for a given desirable probability of false alarm P_{FA} since it depends on the signal-to-noise

ratio (SNR_ζ) at the receiver at the region ζ of the predicted target position. Specifically, the probability of detection is given by

$$P_{\mathrm{D}} = P_{\mathrm{FA}}^{\frac{1}{1+\mathrm{SNR}_\zeta}}. \tag{2.10}$$

Chapter 3

Tracking Multiple Targets and Multiple Model Motion

3.1 Multiple Target Tracking Approaches

Conventional multiple target tracking is normally performed by first coupling the observed measurements to the existing tracks using *measurement-to-track* associations and then employing single target tracking algorithms to track each target separately. An example of measurement-to-track association is the nearest neighbor (NN) method [35, 36], which selects a measurement closest to a track if it falls within the track's specified gate. However, when more than one measurement falls within the gate of one or more targets, then the NN algorithm only detects the track that was first considered. The multiple hypothesis tracking (MHT) is a more general and widely used technique that first performs data association on a sequence of measurements and then performs filtering on each data association hypothesis. However, as it is an exhaustive approach, its computational cost is very high [37]. The joint probabilistic data association (JPDA) approach [37] is a computationally cheaper alternative to the MHT as it considers associations that survive after gating and combines those association according to their likelihood. However, JPDA requires the number of targets to be known a priori [37].

3.2 Probability Hypothesis Density Filter

The probability hypothesis density filter (PHDF) is a computational tractable approximation to the optimal multiple-target Bayesian filters. It is based on the use of random finite set (RFS) theory [44], which is frequently associated with multiple target tracking problems. The PHDF consists of two main steps, the prediction step and the update step, that recursively propagate the first order moment or intensity function of the RFS of the targets of interest [42, 44–48]. If more than one target appears at a given time, the PHDF can iteratively estimate the number of targets while avoiding any explicit measurement-to-track association. Note that the PHDF can be implemented using sequential Monte Carlo methods such as particle filtering (PF) [49]; the mixed-Gaussian

PHDF (GM-PHDF) assumes that the intensities are Gaussian mixtures [56].

3.2.1 Random Finite Sets

At a given time k, a target may randomly appear or disappear, resulting in target birth or target death. A target birth can also be modeled by spawning or directly generated from existing targets. As a result, due to spontaneous target birth, death or spawning, the number of targets is continuously changing with time. For a realistic multiple target tracking problem, the duration for which a target is present is unknown, and the received signal at the sensor may not have measurements from all the targets and/or it may have clutter measurements. The PHDF does not require knowledge of the number of targets a priori, and as a result, it can be used to determine the number of targets at each time step.

The main idea in exploiting the RFS approach [29, 43, 44, 58] is to treat the multiple targets as a set-valued state (multiple-target state) and the multiple measurements as a set-valued observation (multiple target observation). The multiple tracking uncertainties can then be characterized using RFS. An RFS is defined as a finite set-valued random sequence that can be completely characterized by a discrete probability distribution and a family of joint probability densities [59]. The number of elements in the set, or cardinality, is characterized by this discrete distribution. For multiple target tracking, the cardinality of the multiple-target state RFS is the random number of targets, and each RFS element indicates the unknown random state of each target.

3.2.2 Probability Hypothesis Density Filtering Algorithm

We assume that a target generates only one observation at each time step k, and that each target generates measurements independently of each other. The state space representation of the ith target, while present at time step k, can be described by the prior density $p(\mathbf{x}_{k,i}|\mathbf{x}_{k-1,i})$ and the likelihood function $p(\mathbf{z}_{k,i}|\mathbf{x}_{k,i})$. Here, $\mathbf{x}_{k,i}$ is the unknown state of the ith target and $\mathbf{z}_{k,i}$ is the corresponding measurement at time step k. Assum-

ing N_k targets at time k, the multiple-target state RFS is given by

$$\mathbf{X}_k = \{\mathbf{x}_{k,1}, \ldots \mathbf{x}_{k,N_k}\}. \tag{3.1}$$

The multiple target measurement RFS at time k is

$$\mathbf{Z}_k = \{\mathbf{z}_{k,1}, \ldots \mathbf{z}_{k,M_k}\}, \tag{3.2}$$

where M_k is the number of measurements at the receiver at time k. Note that there may be more measurements than targets at any given time k since measurements may also be obtained from clutter. The overall prior density and likelihood function, assuming N_k targets and M_k measurements at time k, are given by $p(\mathbf{X}_k|\mathbf{X}_{k-1})$ and $p(\mathbf{Z}_k|\mathbf{X}_k)$, respectively.

Given the multiple-target state RFS $\mathbf{X}_{k-1}$ at time $(k-1)$, then at time k, the multiple-target state RFS $\mathbf{X}_k$ is formed by combining the surviving and spawned target RFS $\mathbf{X}_{k|k-1}^{\text{surv}}$ and $\mathbf{X}_{k|k-1}^{\text{sp}}$, respectively, from the previous time step $(k-1)$, and the spontaneous target birth RFS $\mathbf{X}_k^{\text{birth}}$. Also, due to the presence of clutter, the received multiple-target measurement RFS $\mathbf{Z}_k$ is formed by the combination of two types of measurement RFS: $\mathbf{Z}_k^{\text{target}}$ generated by the existing targets and $\mathbf{Z}_k^{\text{clutter}}$ generated by false alarms or clutter at time k. It is assumed that the clutter RFS is independent of the target measurement RFS and that the target measurement RFS are mutually independent.

The PHDF uses the assumption that the predicted multiple-target posterior density $p(\mathbf{X}_k|\mathbf{Z}_{k-1})$ can be completely characterized by the corresponding intensity function $\lambda(\mathbf{x}_k|\mathbf{Z}_{k-1})$. With this assumption, given the posterior intensity $\lambda(\mathbf{x}_{k-1}|\mathbf{Z}_{k-1})$ at time step $(k-1)$, the predicted intensity can be obtained as

$$\begin{aligned}\lambda(\mathbf{x}_k|\mathbf{Z}_{k-1}) &= \int \left[\mathrm{P}_{k|k-1}^{\text{surv}}(\mathbf{x}_{k-1})\, p(\mathbf{x}_k|\mathbf{x}_{k-1}) + \lambda^{\text{sp}}(\mathbf{x}_k|\mathbf{Z}_{k-1}) \right] \lambda(\mathbf{x}_{k-1}|\mathbf{Z}_{k-1})\, d\mathbf{x}_{k-1} \\ &+ \lambda^{\text{birth}}(\mathbf{x}_k|\mathbf{Z}_k), \end{aligned} \tag{3.3}$$

where $\lambda^{\text{sp}}(\mathbf{x}_k|\mathbf{Z}_{k-1})$ is the intensity of the targets spawned at the previous time step $(k-1)$, $\lambda^{\text{birth}}(\mathbf{x}_k|\mathbf{Z}_k)$ is the intensity of new target births, and $\mathrm{P}_{k|k-1}^{\text{surv}}(\mathbf{x}_{k-1})$ is the probability

that a target present at time step $(k-1)$ will survive to time step k. The posterior intensity is given by

$$\begin{aligned}\lambda(\mathbf{x}_k|\mathbf{Z}_k) &= (1-\mathrm{P}_k^{\mathrm{D}}(\mathbf{x}_k))\,\lambda(\mathbf{x}_k|\mathbf{Z}_{k-1}) \\ &+ \sum_{\mathbf{z}_k\in\mathbf{Z}_k} \frac{\mathrm{P}_k^{\mathrm{D}}(\mathbf{x}_k)\; p(\mathbf{z}_k|\mathbf{x}_k)\;\lambda(\mathbf{x}_k|\mathbf{Z}_{k-1})}{\lambda^{\mathrm{clutter}}(\mathbf{z}_k)+\int \mathrm{P}_k^{\mathrm{D}}(\tilde{\mathbf{x}}_k)\,p(\mathbf{z}_k|\tilde{\mathbf{x}}_k)\,\lambda(\tilde{\mathbf{x}}_k|\mathbf{Z}_{k-1})\,d\tilde{\mathbf{x}}_k} \end{aligned} \tag{3.4}$$

where $\lambda^{\mathrm{clutter}}(\mathbf{z}_k)$ is the clutter intensity and $\mathrm{P}_k^{\mathrm{D}}(\mathbf{x}_k)$ is the probability that a target present at time step k is detected at time k.

Note that, as a Poisson RFS is completely characterized by its intensity function, the clutter RFS $\mathbf{Z}_k^{\mathrm{clutter}}$, the target spawn RFS $\mathbf{X}_{k|k-1}^{\mathrm{sp}}$ and the target birth RFS $\mathbf{X}_{k|k-1}^{\mathrm{birth}}$ are often modeled as Poisson distributed. Specifically, an RFS is Poisson if its cardinality distribution is also Poisson with mean $\hat{N}_k$ and its finite-set elements are independent and identically distributed with probability density $p(\mathbf{X}_k|\mathbf{Z}_k)=\lambda(\mathbf{X}_k|\mathbf{Z}_k)/\hat{N}_k$. As a result, the expected number of $\mathbf{X}_k|\mathbf{Z}_k$ can provide an estimate for the number of targets $\hat{N}_k$ at time k directly from the posterior intensity as $\hat{N}_k=\int\lambda(\mathbf{x}_k|\mathbf{Z}_k)d\mathbf{x}_k$.

3.2.3 Particle Probability Hypothesis Density Filter

Although the PHDF recursion in Equations (3.3) and (3.4) are considerably simpler than those of the multiple-target Bayesian filter, it still requires solving multi-dimensional integrals. In [60, 61], a closed form solution was provided for the Gaussian-mixture PHDF, but this particular filter has some very strict assumptions.

The particle PHDF (PPHDF) [48,53,62–64] implements the PHDF using a particle filter (PF), which allows for nonlinear and non-Gaussian target dynamic models. At each time step, the PPHDF approximates the posterior intensity by a weighted set of particles, and at the end of every recursion, the multiple target states can be estimated using standard clustering techniques such as the k-means clustering algorithm. The number of targets can be easily estimated by summing the particle weights.

The PPHDF has three main steps, similar to the PF. During the prediction step, the predicted intensity function $\lambda(\mathbf{x}_k|\mathbf{Z}_{k-1})$ is obtained; then the posterior intensity

$\lambda(\mathbf{x}_k|\mathbf{Z}_k)$ is computed during the update and resampling steps. The recursion requires an assumed initial intensity function $\lambda(\mathbf{x}_0)$ at time $k = 0$.

For the prediction step, the PPHDF assumes that the posterior intensity function $\lambda(\mathbf{x}_{k-1}|\mathbf{Z}_{k-1})$ can be approximated using L_{k-1} particles $\mathbf{x}_{k-1}^{(i)}$ and associated normalized weights $w_{k-1}^{(i)}$, $i = 1,\ldots,L_{k-1}$ as

$$\lambda(\mathbf{x}_{k-1}|\mathbf{Z}_{k-1}) \approx \sum_{i=1}^{L_{k-1}} w_{k-1}^{(i)} \delta(\mathbf{x}_{k-1} - \mathbf{x}_{k-1}^{(i)}). \tag{3.5}$$

Then, the predicted intensity function $\lambda(\mathbf{x}_k|\mathbf{Z}_{k-1})$ can be approximated as $q_k(\cdot|\mathbf{x}_{k-1}^{(i)}, \mathbf{Z}_k)$ and $p_k(\cdot|\mathbf{Z}_k)$

$$\lambda(\mathbf{x}_k|\mathbf{Z}_{k-1}) \approx \sum_{i=1}^{L_{k-1}+J_k} w_{k|k-1}^{(i)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}), \tag{3.6}$$

where the particles are obtained using two importance sampling densities

$$\mathbf{x}_k^{(i)} \sim q_k(\cdot|\mathbf{x}_{k-1}^{(i)}, \mathbf{Z}_k),\ i = 1,\cdots,L_{k-1}, \tag{3.7}$$

$$\mathbf{x}_k^{(i)} \sim p_k(\cdot|\mathbf{Z}_k),\ i = L_{k-1}+1,\cdots,L_{k-1}+J_k, \tag{3.8}$$

and the weights are computed as

$$w_{k|k-1}^{(i)} = \frac{\phi_{k|k-1}(\mathbf{x}_k^{(i)}, \mathbf{x}_{k-1}^{(i)})\ w_{k-1}^{(i)}}{q_k(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)}, \mathbf{Z}_k)},\ i = 1,\cdots,L_{k-1}, \tag{3.9}$$

$$w_{k|k-1}^{(i)} = \frac{\lambda^{\text{birth}}(\mathbf{x}_k^{(i)}|\mathbf{Z}_k)}{J_k\ p_k(\mathbf{x}_k^{(i)}|\mathbf{Z}_k)},\ i = L_{k-1}+1,\cdots,L_{k-1}+J_k, \tag{3.10}$$

with

$$\phi_{k|k-1}(\mathbf{x}_k^{(i)}, \mathbf{x}_{k-1}^{(i)}) = \mathrm{P}_{k|k-1}^{\text{surv}}(\mathbf{x}_{k-1}^{(i)}) p(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)}) + \lambda^{\text{sp}}(\mathbf{x}_k^{(i)}|\mathbf{Z}_{k-1}). \tag{3.11}$$

and J_k is the total number of particles required to represent the new birth target RFS.

For the update step of the PPHDF, the posterior intensity function at time k is approximated as

$$\lambda(\mathbf{x}_k|\mathbf{Z}_k) \approx \sum_{i=1}^{L_{k-1}+J_k} w_k^{(i)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}), \tag{3.12}$$

where the particle weight is updated as

$$w_k^{(i)} = (1 - \mathrm{P}_k^{\mathrm{D}}(\mathbf{x}_k^{(i)}))\, w_{k|k-1}^{(i)} + \sum_{\mathbf{z}_k \in \mathbf{Z}_k} \frac{\mathrm{P}_k^{\mathrm{D}}(\mathbf{x}_k^{(i)})\; p(\mathbf{z}_k|\mathbf{x}_k^{(i)})\; w_{k|k-1}^{(i)}}{\lambda^{\mathrm{clutter}}(\mathbf{z}_k) + C_k(\mathbf{z}_k)} \tag{3.13}$$

where

$$C_k(\mathbf{z}_k) = \sum_{i=1}^{L_{k-1}+J_k} \mathrm{P}_k^{\mathrm{D}}(\mathbf{x}_k^{(i)})\, p(\mathbf{z}_k|\mathbf{x}_k^{(i)}) w_{k|k-1}^{(i)}. \tag{3.14}$$

In order to avoid particle degeneracy [2], low weight particles are eliminated and high weight particles are multiplied in order to focus on important regions of the intensity function. This is done by resampling and normalizing $L_{k-1} + J_k$ particles to redistribute the weight. The normalization factor is the estimated number of targets, $\hat{N}_k = \lfloor \sum_{i=1}^{L_{k-1}+J_k} w_k^{(i)} \rfloor$, where $\lfloor q \rfloor$ takes the integer smaller or equal to q. The new set of particles and corresponding weights is given by $(\mathbf{x}_k^{(i)}, w_k^{(i)}/\hat{N}_k)$.

3.3 Multiple-Model Particle Filter

In maneuvering target tracking, the target may change its motion model at any time. As conventional particle filters (PFs) are not robust for tracking maneuvering targets with multiple state models, they were combined with the multiple model (MM) approach [1, 53, 65–67]. At every time step, the MM-PF needs to estimate the state model as well as the target state. It is assumed that the target model can switch according to a transitional probability matrix $\Pi = \{\pi_{mn}\}$. The model number $\varpi_k^{(i)}$ of the ith particle $i = 1, \ldots, L_k$ at time k, follows the transitional matrix, where L_k is the number of particles that still exist at time k. Specifically, if at time $k-1$ a particle has model index number $m = \varpi_{k-1}^{(i)}$, then at time index k, the model index transfers to model number n with probability π_{mn}.

The model transition algorithm in Table. 3.1 is used for generating $\varpi_k^{(i)}$ from $\varpi_{k-1}^{(i)}$ according to the transitional matrix Π.

The integrated MM-PF algorithm is outlined in Table. 3.2, where $\{\mathbf{x}_k^{(i)}, \varpi_k^{(i)}\}_{i=1}^{L_k}$ is used to denote the ith particle state, $w_k^{(i)}$ is the ith particle weight at time k, and $\mathbf{x}_k$ and $\mathbf{z}_k$ denotes the single target state vector and measurement vector, respectively. At time k, L_k new model index particles $\varpi_k^{(i)}$, $i = 1, \ldots, L_k$, are generated according to the

Table 3.1: Target Model Transition Algorithm

$(\boldsymbol{\varpi}_k^{(i)},\, L_k) = \text{Transition}(\boldsymbol{\varpi}_{k-1}^{(i)},\, L_k,\, \Pi)$

For $m = 1 : S$ (*For all number of models S*)
  $c_m(0) = 0$ (*Set initial value to zero*)
  For $n = 1 : S$ (*Second loop to calculate cumulative density*)
    $c_m(n) = c_m(n-1) + \boldsymbol{\pi}_{mn}$
  End For
End For
For $i = 1 : L_k$ (*For all number of particles L_k*)
  Draw $u_{(i)} \sim U[0,1]$ (*Draw a random value from uniform distribution*)
  Set $m = \boldsymbol{\varpi}_{k-1}^{(i)}$ (*Previous model number*)
  $n = 1$ (*For current model number n*)
  While$(c_m(n) < u_{(i)})$
    $n = n + 1$
  End For
  Set $\boldsymbol{\varpi}_k^{(i)} = n$
End For

transition matrix, which includes all the possible target state models at time k. Then the PF is applied according to the generated model index in each particle.

For the MM-PF, the optimal importance density $q(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)}, \boldsymbol{\varpi}_k^{(i)}, \mathbf{z}_k)$ is given by [1]

$$q(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)}, \boldsymbol{\varpi}_k^{(i)}, \mathbf{z}_k)_{opt} = p(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)}, \boldsymbol{\varpi}_k^{(i)}, \mathbf{z}_k). \tag{3.15}$$

As this optimal distribution is usually not known, a commonly used sub-optimal importance distribution is the transitional prior [66],

$$q(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)}, \boldsymbol{\varpi}_k^{(i)}, \mathbf{z}_k) = p(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)}, \boldsymbol{\varpi}_k^{(i)}). \tag{3.16}$$

Table 3.2: Multiple Model Particle Filter

$(\mathbf{x}_k^{(i)},\ \boldsymbol{\varpi}_k^{(i)},\ w_k^{(i)},\ L_k) = \text{MM-PF}(\mathbf{x}_{k-1}^{(i)},\ \boldsymbol{\varpi}_{k-1}^{(i)},\ w_{k-1}^{(i)},\ L_k,\ \mathbf{z}_k)$

Model Transition 3.1

$[\{\boldsymbol{\varpi}_k^{(i)}\}_{i=1}^{L_k}] = Tran[\{\boldsymbol{\varpi}_{k-1}^{(i)}\}_{i=1}^{L_k}, \Pi]$

For $i = 1 : L_k$ (L_k *is used to denote the total number of particles*)

Draw $\mathbf{x}_k^{(i)} \sim q(\mathbf{x}_k^{(i)} | \mathbf{x}_{k-1}^{(i)}, \boldsymbol{\varpi}_k^{(i)}, \mathbf{z}_k)$

update the particle weight

$\tilde{w}_k^{(i)} = w_{k-1}^{(i)} \frac{p(\mathbf{z}_k | \mathbf{x}_k^{(i)}, \boldsymbol{\varpi}_k^{(i)}) p(\mathbf{x}_k^{(i)} | \mathbf{x}_{k-1}^{(i)}, \boldsymbol{\varpi}_k^{(i)})}{q(\mathbf{x}_k^{(i)} | \mathbf{x}_{k-1}^{(i)}, \boldsymbol{\varpi}_k^{(i)}, \mathbf{z}_k)}$

End For

$W_{\text{sum}} = \sum_i \tilde{w}_k^{(i)}$ (*Calculate the total weight*)

For $i = 1 : L_k$

$w_k^{(i)} = \tilde{w}_k^{(i)} / W_{\text{sum}}$

End For

Calculate the $\tilde{N}_{eff}$ (*Effective sample size* $\tilde{N}_{eff}$)

$\tilde{N}_{eff} = \frac{1}{\sum_{i=1}^{L_k} (w_k^{(i)})^2}$

If $\tilde{N}_{eff} < N_{threshold}$

Resampling

End If

Chapter 4

Multiple Target Tracking in Urban Terrain

We propose to investigate the multiple-target tracking problem in urban environments by efficiently estimating target states using the particle probability hypothesis density filter (PPHDF) presented in Chapter 3. When tracking in urban terrain, as discussed in Chapter 2, multipath returns need to be exploited due to the lack of line-of-sight (LOS) returns as well as the presence of multipath interference and high clutter. When there is also a need to simultaneously track multiple targets, the problem of extracting multiple target state information from an increased number of multipath returns becomes even more difficult, and the dynamic tracking system model needs to be appropriately modeled.

4.1 Dynamic System Formulation

The formulation of a dynamic Urban TErrain Multiple-Target Tracking (UTE-MTT) system entails the integration of the multipath exploitation (ME) system formulation in Chapter 2 with the multiple-target tracking (MTT) system formulation in Chapter 3. For the ME system, the formulation of the dynamic state space formulation is given by Equation (2.1), allowing for both constant velocity and coordinated turn motion models, and Equations (2.2)-(2.8). In Equations (2.1)-(2.8), $\mathbf{x}_k$ is the unknown state of a single target and $\mathbf{z}_k$ is the measurement corresponding to that target. For all general purposes, we can let replace $\mathbf{x}_k$ with $\mathbf{x}_{k,1}$ and $\mathbf{z}_k$ with $\mathbf{z}_{k,1}$ in (2.1)-(2.8) to emphasize that this ME-based formulation is for a single target.

Extending to multiple targets in the urban environment, the multiple-target state RFS can be given as in (3.1) by $\mathbf{X}_k = \{\mathbf{x}_{k,1}, \ldots \mathbf{x}_{k,N_k}\}$, where, at time k, $\mathbf{x}_{k,i}$ is the unknown state of the ith target and the number of targets being tracked is assumed to be N_k. Note that $\mathbf{x}_{k,i}$ follows the state model in (2.1), with the two possible motion models.

The multiple-target measurement RFS with M_k measurements at time k is given

by $\mathbf{Z}_k = \{\mathbf{z}_{k,1}, \ldots \mathbf{z}_{k,M_k}\}$. However, unlike in Equation (3.2) where each measurement corresponds to a single range and range rate pair, the ith measurement $\mathbf{z}_{k,i}$ at time k may be the result of $P_{k,i}$ possible measurement paths, and thus $P_{k,i}$ possible range and range rate pairs, as in Equations (2.6)-(2.8). Note that in Chapter 3, we assumed that a target generates only one observation at each time step k. For the multiple-target urban tracking problem, we assume that the $P_{k,i}$ paths for the ith target are arranged together in one vector. Such an assumption requires prior knowledge about the data received at the receiver. Note, however, that our proposed modified PPHDF, to be presented later in this chapter, does not need to satisfy this assumption.

4.2 Development of PHDF with Multipath Exploitation

With the multiple-target state RFS and multiple-target measurement RFS appropriately specified for multiple model state and multiple target measurements, the predicted intensity function $\lambda(\mathbf{x}_k|\mathbf{Z}_{k-1})$ and posterior intensity function $\lambda(\mathbf{x}_k|\mathbf{Z}_k)$ are given by Equations (3.3) and (3.4), respectively. For the PF implementation, following similar steps as with the PPHD for a single target in Chapter 3.2, we first obtain the posterior intensity $\lambda(\mathbf{x}_{k-1}|\mathbf{Z}_{k-1})$ using particles and weights at time step $(k-1)$ as in Equation (3.5) with $\mathbf{x}_k$ representing the states of all N_k targets. The particles are given by

$$\mathbf{x}_k^{(i)}, \rho_k^{(i)} \sim q_k\left(\cdot|\mathbf{x}_{k-1}^{(i)}, \mathbf{Z}_k, \varpi_k^{(i)}\right), \quad i = 1, \cdots, L_{k-1} \tag{4.1}$$

$$\mathbf{x}_k^{(i)}, \rho_k^{(i)} \sim p_k(\cdot|\mathbf{Z}_k), \quad i = L_{k-1}+1, \cdots, L_{k-1}+J_k \tag{4.2}$$

and their corresponding weights can be computed using

$$w_{k|k-1}^{(i)} = \frac{\phi_{k|k-1}\left(\mathbf{x}_k^{(i)}, \mathbf{x}_{k-1}^{(i)}\right) w_{k-1}^{(i)}}{q_k\left(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)}, \mathbf{Z}_k, \varpi_k^{(i)}\right)} \quad i = 1, \cdots, L_{k-1} \tag{4.3}$$

$$w_{k|k-1}^{(i)} = \frac{\lambda^{\text{birth}}\left(\mathbf{x}_k^{(i)}|\mathbf{Z}_k\right)}{J_k\, p_k\left(\mathbf{x}_k^{(i)}|\mathbf{Z}_k\right)}, \quad i = L_{k-1}+1, \cdots, L_{k-1}+J_k, \tag{4.4}$$

where $\phi_{k|k-1}(\mathbf{x}_k^{(i)}, \mathbf{x}_{k-1}^{(i)})$ is defined in (3.11), $\rho_k^{(i)}$ is the ith region index particle, and $\varpi_k^{(i)}$ is the ith model index particle at time k. In the modified PPHDF, the new parameter

$\rho_k^{(i)}$ indicates at the step k which physical region on the road map where the ith particle $\mathbf{x}_k^{(i)}$ is.

The update intensity can be approximated using (3.12), with the weights obtained as

$$w_{k|k-1}^{(i)} = \left[1 - \mathrm{P}_k^{\mathrm{D}}(\mathbf{x}_k^{(i)}) + \sum_{\mathbf{z}_k \in \mathbf{Z}_k} \frac{\psi_{k,\mathbf{z}_k}(\mathbf{x}_k^{(i)}, \rho_k^{(i)})}{\lambda^{\mathrm{clutter}}(\mathbf{z}_k) + C_k(\mathbf{z}_k)}\right] w_{k|k-1}^{(i)} \tag{4.5}$$

where

$$C_k(\mathbf{z}_k) = \sum_{j=1}^{L_{k-1}+J_k} \psi_{k,\mathbf{z}_k}\left(\mathbf{x}_k^{(j)}, \rho_k^{(j)}\right) w_{k|k-1}^{(j)} \tag{4.6}$$

and $\psi_{k,\mathbf{z}_k}(\mathbf{x}_k^{(j)}, \rho_k^{(j)}) = \mathrm{P}_k^{\mathrm{D}}(\mathbf{x}_k^{(j)}) g_k(\mathbf{z}_k | \mathbf{x}_k^{(j)}, \rho_k^{(j)})$.

The PPHDF algorithm is described in Table 4.1.

At the initialization step, particles are generated randomly according to a uniform distribution. The initial number of particles should be very large to ensure that some of the particles survive after the update stage. As a result, the computational intensity of the initialization step is high.

In the PPHDF algorithm, the predicted state particles are drawn according to (3.7) and (3.8). As the state model changes depending on the target's motion, another set of particles, $\varpi_k^{(i)}$, need to be included to indicate the model index at time k for the ith state particle. The advantage of the multiple-model method is demonstrated using a comparison between the multiple-model algorithm and the single-model algorithm. In Figure 4.1, as we can see, the particles generated by the single-model lead to an inaccurate estimate, especially when the target suddenly changes its motion. When the multiple-model is used, the resulting particles are shown in Figure 4.2. Unlike the single-model case, the resulting particles are distributed around the true state value. As a result, the track performance is expected to be improved when a multiple-model estimator is used.

Similar to particle filters, the degeneracy problem exists for the PPHDF. A mea-

Table 4.1: Multiple-Model Multiple-Target PPHDF

Step 0.(Initialization at time index $k = 0$)

For $i = 1 : L_0$ Sample the particles

$\mathbf{x}_k^{(i)} \sim$ Uniform distribution

Set the particle weight

$w_k^{(i)} = 1/L_0$

set model index number

$\varpi_0^{(i)} = 1$

End For

Set $k = 1$

Step 1.(Prediction Stage, when $k \geq 1$)

update the model index number

$(\varpi_k^{(i)},\ L_k) = \text{Transition}(\varpi_{k-1}^{(i)},\ L_k,\ \Pi)$

For $i = 1 : L_{k-1}$

$\mathbf{x}_k^{(i)} \sim q(\cdot|\mathbf{x}_{k-1}^{(i)}, \varpi_k^{(i)}, \mathbf{Z}_k)$

$w_{k|k-1}^{(i)} = \frac{\phi_{k|k-1}(\mathbf{x}_k^{(i)}, \mathbf{x}_{k-1}^{(i)}) w_{k-1}^{(i)}}{q(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)}, \varpi_k^{(i)}, \mathbf{Z}_k)}$

End For

For $i = L_{k-1} + 1 : L_{k-1} + J_k$

$\mathbf{x}_k^{(i)} \sim p(\cdot|\mathbf{Z}_k)$

$w_{k|k-1}^{(i)} = \frac{1}{J_k} \frac{\lambda^{\text{birth}}(\mathbf{x}_k^{(i)}|\mathbf{Z}_k)}{p(\mathbf{x}_k^{(i)}|\mathbf{Z}_k)}$

End For

The total number of particles $N_{\text{particle}} = L_{k-1} + J_k$

Step 2.(Update Stage, when $k \geq 1$)

For $\mathbf{z}_k \in \mathbf{Z}_k$

$< w_{k|k-1}, \psi_{k,\mathbf{z}_k} > = \sum_{i=1}^{N_{\text{particle}}} \psi_{k,\mathbf{z}_k}(\mathbf{x}_k^{(i)}) w_{k|k-1}^{(i)}$

End For

For $i = 1 : N_{\text{particle}}$

Calculated updated weight

$w_k^{(i)} = [1 - \text{P}_k^{\text{D}}(\mathbf{x}_k^{(i)}) + \sum_{\mathbf{z}_k \in \mathbf{Z}_k} \frac{\psi_{k,\mathbf{z}_k}(\mathbf{x}_k^{(i)})}{\lambda^{\text{clutter}}(\mathbf{z}_k) + < w_{k|k-1}, \psi_{k,\mathbf{z}_k} >}] w_{k|k-1}^{(i)}$

End For

Step 3.(Resampling, when $k \geq 1$)

$\hat{N}_k = \lfloor (\sum_{i=1}^{N} w_k^{(i)}) \rfloor$

Resample $\{w_k^{(i)}, \mathbf{x}_k^{(i)}\}_{i=1}^{N_{\text{particle}}}$ to $\{w_k^{(i)}, \mathbf{x}_k^{(i)}\}_{i=1}^{L_k}$

Step 4.(Target Estimation, when $k \geq 1$)

K-mean Clustering algorithm

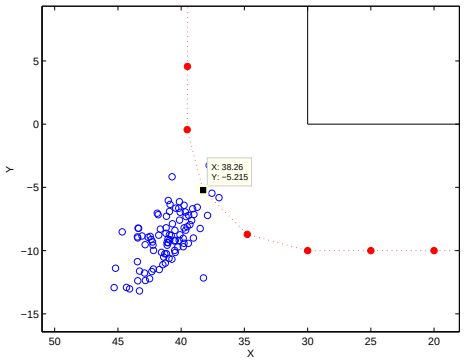

Figure 4.1: Particle (blue circles) distribution using a single-model algorithm. The true target is shown as a black square.

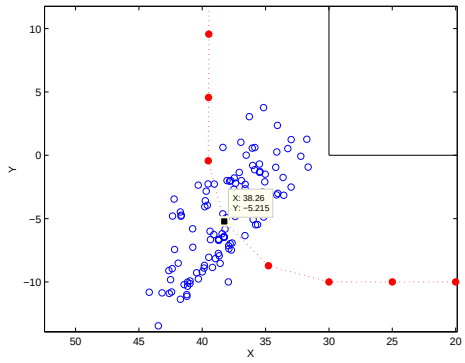

Figure 4.2: Particle (blue circles) distribution using a multiple-model algorithm. The true target is shown as a black square.

sure of degeneracy is the effective sample size $\tilde{N}_{\text{eff}}$ [1] given by

$$\tilde{N}_{\text{eff}} = \frac{1}{\sum_{i=1}^{N_{\text{particle}}} (w_k^{(i)})^2}, \tag{4.7}$$

where $w_k^{(i)}$ is the normalized weight calculated in (4.12). We can see that $1 \leq \tilde{N}_{\text{eff}} \leq N_{\text{particle}}$, and a small $\tilde{N}_{\text{eff}}$ indicates serious degeneracy. We perform resampling at the end of each iteration to remove the particles with low weights and duplicate the particles with high weights After resampling all the particles have equal weight.

From the particle representation of the posterior intensity after resampling, the states of the individual targets are estimated using clustering, which can be performed using the k-means algorithm [68]. The k-means clustering technique partitions the given particle representation into the number of clusters, which is given by the integer approximation of the expected number of targets. And the center of each cluster indicates a local maximum of the intensity function and hence gives the state estimate of a target.

4.3 Modified Particle Probability Hypothesis Density Filter

Although the PPHDF avoids conventional data association, it requires a prior knowledge of any path-to-measurement associations. Specifically, at the receiver, when the radar observes all the range-range rate pairs, it is assumed that it can successfully dis-

tinguish which range-range rate pair corresponding to what path. In practice, however, the matched filter may not receive information such as range and range rate, in any particular order. Although, by having the road map information, the above assumption can be achieved, it still involves a lot of quite pre-process work.

We propose a modified PPHDF (MPPHDF) that can perform the path or region to measurement association and include it the algorithm steps. This is done by including a region parameter for the ith particle at time k, $\rho_k^{(i)}$. In this case, the new prediction process can be updated as follow,

$$\mathbf{x}_k^{(i)}, \rho_k^{(i)} \sim q_k(\cdot|\mathbf{x}_{k-1}^{(i)}, \mathbf{Z}_k, \varpi_k^{(i)}),\ i = 1, \cdots, L_{k-1} \tag{4.8}$$

where L_{k-1} is defined in (3.5). While for particles corresponding to the new born targets,

$$\mathbf{x}_k^{(i)}, \rho_k^{(i)} \sim p_k(\cdot|\mathbf{Z}_k),\ i = L_{k-1}+1, \cdots, L_{k-1}+J_k \tag{4.9}$$

In addition, from the road map given in Fig. 4.3, the surveillant area is divided into 8 regions. In MPPHDF, each particle corresponds to a vector $\mathbf{x}_k^{(i)} = [x_k^{(i)}, \dot{x}_k^{(i)}, y_k^{(i)}, \dot{y}_k^{(i)}]^T$, the following table gives a criterion to calculate region index.

Table 4.2: Region index criterion

case 1	$(x_k^{(i)} + y_k^{(i)} \leq 0)$ $(30 \leq x_k^{(i)} + y_k^{(i)} \leq 60) \cap (x_k^{(i)} - y_k^{(i)} \leq -30) \cap (y_k^{(i)} \leq 60)$ $(130 \leq x_k^{(i)} + y_k^{(i)} \leq 160) \cap (130 \leq x_k^{(i)} - y_k^{(i)})$
case 2	$(0 \leq x_k^{(i)} + y_k^{(i)} \leq 30) \cap (x_k^{(i)} \leq 0)$
case 3	$(60 \leq x_k^{(i)} + y_k^{(i)} \leq 160) \cap (-30 \leq x_k^{(i)} - y_k^{(i)}) \cap (30 \leq y_k^{(i)} \leq 60)$
case 4	$(160 \leq x_k^{(i)} + y_k^{(i)} \leq 190) \cap (x_k^{(i)} - y_k^{(i)} \leq 130) \cap (x_k^{(i)} \leq 160) \cap (y_k^{(i)} \leq 60)$
case 5	$(160 \leq x_k^{(i)} + y_k^{(i)}) \cap (130 \leq x_k^{(i)} - y_k^{(i)}) \cap (x_k^{(i)} \leq 160)$
case 6	$(0 \leq x_k^{(i)} + y_k^{(i)} \leq 130) \cap (-50 \leq y_k^{(i)} \leq 0)$
case 7	$(60 \leq x_k^{(i)} + y_k^{(i)}) \cap (x_k^{(i)} - y_k^{(i)} \leq -30) \cap (y_k^{(i)} \leq 60)$
case 8	$(x_k^{(i)} + y_k^{(i)} \leq 60) \cap (-30 \leq x_k^{(i)} - y_k^{(i)}) \cap (30 \leq y_k^{(i)})$ $(x_k^{(i)} + y_k^{(i)} \leq 160) \cap (x_k^{(i)} - y_k^{(i)} \leq 130) \cap (130 \leq x_k^{(i)})$

After the predicted particles are generated, the corresponding particle weight is propagated based on following equations.

$$w_{k|k-1}^{(i)} = \frac{\phi_{k|k-1}(\mathbf{x}_k^{(i)}, \mathbf{x}_{k-1}^{(i)}) w_{k-1}^{(i)}}{q_k(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)}, \mathbf{Z}_k)},\ i = 1, \cdots, L_{k-1}, \tag{4.10}$$

$$w_{k|k-1}^{(i)} = \frac{\lambda^{\text{birth}}(\mathbf{x}_k^{(i)}|\mathbf{Z}_k)}{J_k p_k(\mathbf{x}_k^{(i)}|\mathbf{Z}_k)},\ i = L_{k-1}+1, \cdots, L_{k-1}+J_k, \tag{4.11}$$

where ϕ is defined in (3.11).

When the radar observes all the range-range rate pairs, the proposed filter updates the particle weights according to the following equations,

$$w_{k|k-1}^{(i)} = [1 - \mathrm{P}_k^{\mathrm{D}}(\mathbf{x}_k^{(i)}) + \sum_{\mathbf{z}_k \in \mathbf{Z}_k} \frac{\psi_{k,\mathbf{z}_k}(\mathbf{x}_k^{(i)}, \rho_k^{(i)})}{\lambda^{\text{clutter}}(\mathbf{z}_k) + C_k(\mathbf{z}_k)}] w_{k|k-1}^{(i)} \tag{4.12}$$

$$C_k(\mathbf{z}_k) = \sum_{j=1}^{L_{k-1}+J_k} \psi_{k,\mathbf{z}_k}(\mathbf{x}_k^{(j)}, \rho_k^{(j)}) w_{k|k-1}^{(j)} \tag{4.13}$$

where $\psi_{k,\mathbf{z}_k}(\mathbf{x}_k^{(j)}, \rho_k^{(j)}) = \mathrm{P}_k^{\mathrm{D}}(\mathbf{x}_k^{(j)}) g_k(\mathbf{z}_k|\mathbf{x}_k^{(j)}, \rho_k^{(j)})$.

The difference between conventional PPHDF technique and modified PPHDF (MPPHDF) is the measurement in this case becomes unordered, then PHDF can not be directly applied to get the updated particle weight as described in (**??**) and (**??**). The proposed filtering algorithm first predict the measurement of each particle $\mathbf{x}_k^{(i)}$ by using the parameter $\rho_k^{(i)}$, and the table below shows the relationship between region index and predicted measurements.

Table 4.3: Region index criterion

case 1	LOS
case 2	LOS + One-bounce
case 3	One-bounce
case 4	Two One-bounce
case 5	LOS + One-bounce
case 6	LOS + One-bounce
case 7	LOS + One-bounce
case 8	Shadowing

Once a range-range rate pair is observed, the likelihood of ith range-range rate pair, i.e., $\mathbf{z}_{k,i}$ given jth particle, i.e., $\mathbf{x}_k^{(j)}$ is calculated as follow,

First, the filter generates another prediction random finite set according to jth particle $\mathbf{x}_k^{(j)}$ and its region index parameter $\rho_k^{(j)}$, denotes as $\mathbf{R}_k^{(j)}$, whose elements are all the possible range-range rate pairs corresponding to the particle $\mathbf{x}_k^{(j)}$. For example, if the particle with the region parameter indicates it is currently in the LOS plus one-bounce region, the filter then checks the measurement equation for this particle, and predicts all the possible observations, i.e., observation pair from LOS, observation from one-bounce and observation from LOS-one-bounce. Then, the filter calculates the likelihood of each element in the generated prediction random finite set given $\mathbf{z}_{k,i}$ individually and selects the largest one as the likelihood of ith range-range rate pair $\mathbf{z}_{k,i}$ given jth particle $\mathbf{x}_k^j$. Then, the filter updates particle weights using (4.12) and (4.13).

After the updated particle weight is obtained, the resampling process is applied to handle the degeneracy problem, which is the same process as described in previous chapter. From the particle representation of posterior intensity after resampling, the states of the individual targets are estimated via clustering algorithm. In our proposed filtering algorithm, the PHDF no longer provides ability to estimate number of targets, because the measurement equation here is different from regular one (with the assumption of knowing which rang-range rate pair corresponding to LOS or multipath). In this case, the clustering algorithm should be modified to estimate both the number of targets and target states. The new k-means clustering algorithm first sets up a threshold, and starts the standard k-means clustering algorithm with cluster number to be 1. At the end of clustering, the average error distance is calculated and compared with the threshold, if the average distance is within the threshold, the new clustering algorithm stops and claims the number of targets is equal to the number of clusters, the estimated target states are the centroid of each cluster, or the cluster number is increase by 1 and the standard k-means algorithm is applied again with the new cluster number.

The proposed new MPPHDF algorithm is shown in Table 4.4

4.4 Numerical Simulation Setup

In order to demonstrate the performance of our proposed algorithm, we provide simulations for tracking multiple targets in urban terrain. We will demonstrate both the PPHDF, that assumes knowledge of path-to-measurement association, as well as the modified PPHDF that does not assume such prior knowledge. In both cases, the PPHDF can recursively calculate the total number of targets at each time step.

Our numerical simulations are based on the three-dimensional test bench environment depicted in Figure 4.3 The targets considered in the simulations are ground

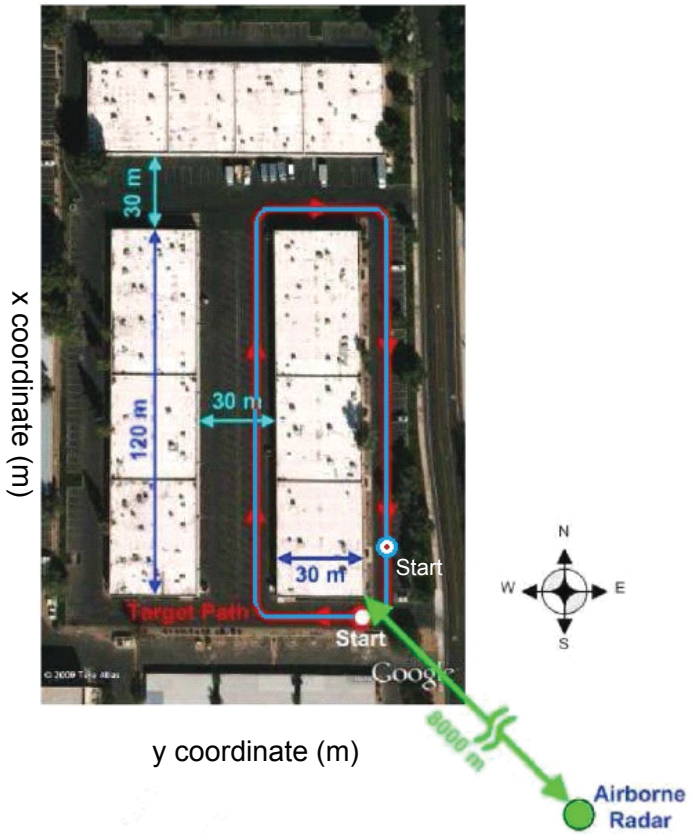

Figure 4.3: The road map of the test bench urban environment to be used for the PPHDF and MMPHDF simulations.

Table 4.4: Modified Multiple-Model Multiple-target Particle Probability Hypothesis Density Filter

Step 0.(Initialization at time index $k = 0$)

For $i = 1 : L_0$

Sample the particles

$\mathbf{x}_k^{(i)} \sim$ Gaussian distribution

Set the particle weight

$w_k^{(i)} = 1/L_0$

set model index number

$\varpi_0^{(i)} = 1$

End For

Set $k = 1$

Step 1.(Prediction Stage, when $k \geq 1$)

update the model index number

$[\{\varpi_k^{(i)}\}_{i=1}^{L_{k-1}}] = Tran[\{\varpi_{k-1}^{(i)}\}_{i=1}^{L_{k-1}}, \Pi]$

For $i = 1 : L_{k-1}$

$\mathbf{x}_k^{(i)}, \rho_k^{(i)} \sim q(\cdot|\mathbf{x}_{k-1}^{(i)}, \varpi_k^{(i)}, \mathbf{Z}_k)$

$w_{k|k-1}^{(i)} = \frac{\phi_k(\mathbf{x}_k^{(i)}, \mathbf{x}_{k-1}^{(i)}) w_{k-1}^{(i)}}{q(\mathbf{x}_k^{(i)}|\mathbf{x}_{k-1}^{(i)}, \mathbf{Z}_k)}$

End For

For $i = L_{k-1} + 1 : L_{k-1} + J_k$

$\mathbf{x}_k^{(i)}, \rho_k^{(i)} \sim p(\cdot|\mathbf{Z}_k)$

$w_{k|k-1}^{(i)} = \frac{1}{J_k} \frac{\lambda^{\text{birth}}(\mathbf{x}_k)|\mathbf{Z}_k}{p_k(\mathbf{x}_k^{(i)}|\mathbf{Z}_k)}$

End For

The total number of particles $N_{particle} = L_{k-1} + J_k$

Step 2.(Update Stage, when $k \geq 1$)

For $\mathbf{z}_k \in \mathscr{Z}_k$

$< w_{k|k-1}, \psi_{k,\mathbf{z}_k} > = \sum_{i=1}^{N_particle} \psi_{k,\mathbf{z}_k}(\mathbf{x}_k^{(i)}, \rho_k^{(i)}) w_{k|k-1}^{(i)}$

End For

For $i = 1 : N_{particle}$

Calculated updated weight

$w_k^{(i)} = [1 - \mathrm{P}_k^{\mathrm{D}}(\mathbf{x}_k^{(i)}) + \sum_{\mathbf{z}_k \in \mathbf{Z}_k} \frac{\psi_{k,\mathbf{z}_k}(\mathbf{x}_k^{(i)}, \rho_k^{(i)})}{\lambda^{\text{clutter}}(\mathbf{z}_k) + < w_{k|k-1}, \psi_{k,\mathbf{z}_k} >}] w_{k|k-1}^{(i)}$

End For

Step 3.(Resampling, when $k \geq 1$)

Resample $\{w_k^{(i)}, \mathbf{x}_k^{(i)}\}_{i=1}^{N_{particle}}$ to $\{w_k^{(i)}, \mathbf{x}_k^{(i)}\}_{i=1}^{L_k}$

Step 4.(Target Estimation, when $k \geq 1$)

Modified K-means Clustering algorithm

vehicles, and thus move on a 2-D plane. In Figure 4.3, there are three buildings, and an airborne radar is located approximately 8000 m southeast of the scene, which is about 1400 m in height. As a result, when either of the ground vehicles travels between the buildings, the LOS signal returns are lost.

According to the locations of the three buildings and the radar, Figure 4.4 illustrates the measurement map of different regions such as LOS, multipath and shadowing [11].

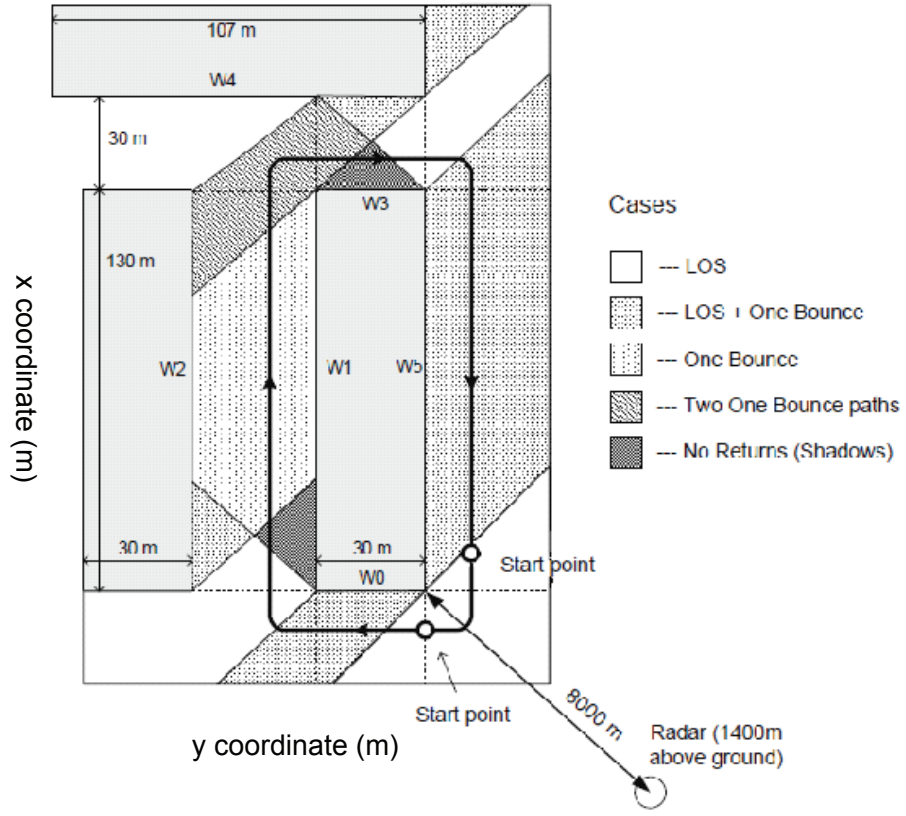

Figure 4.4: Available measurement map for the simulated urban terrain [11].

As shown in Figure 4.4, the urban scene contains five different types of regions including LOS regions, one-bounce regions, two one-bounce regions, and shadowing regions. When the ground vehicle moves in to a specific region, the corresponding mea-

surement pattern is used to generate radar measurements. Moreover, all the simulations are performed based on this test bench.

4.4.1 Two Targets in Urban Terrain

We assume that there are two ground vehicles, whose loop trajectory and starting points are marked in Figure 4.3. The target state is denoted by $\mathbf{x}_k = [x_k, \dot{x}_k, y_k, \dot{y}_k]^T$ with its position (x_k, y_k) and velocity $(\dot{x}_k, \dot{y}_k)$ in Cartesian coordinates. The measurements consist of noisy range and range-rate observations.

Each target can switch between the following linear state model

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{n}_k, \tag{4.14}$$

where $\mathbf{F}$ can represent a constant velocity model and given by ith model index $\varpi_k^{(i)} = 1$

$$\mathbf{F} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Or $\mathbf{F}$ can represent a constant turning model and given by ith model index $\varpi_k^{(i)} = 2$

$$\mathbf{F} = \begin{bmatrix} 1 & sin(\omega)/\omega & 0 & -(1-cos(\omega))/\omega \\ 0 & cos(\omega) & 0 & -sin(\omega) \\ 0 & (1-cos(\omega))/\omega & 1 & sin(\omega)/\omega \\ 0 & sin(\omega) & 0 & cos(\omega) \end{bmatrix}.$$

where $\omega = -2$ is the angular turning rate, $\mathbf{v}_k$ is assumed to be a zero-mean, white Gaussian sequence with covariance matrix:

$$\mathbf{Q} = q \begin{bmatrix} \frac{1}{3} & \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{3} & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & 1 \end{bmatrix}$$

where $q = 0.04$ is a coefficient determining the process noise intensity. We assume that the probability of target survial is 0.95. When implementing the PPHDF, we assign each target 2000 particles.

In this example, we assume that there is no spawning and that no new targets appear. The clutter RFS is considered to be Poisson, and the pdf of each clutter is assumed to be uniformly distributed in the region $[-50, 150] \times [-100, 50]$. Moreover, the clutter density is assumed to be 3.33×10^{-4}, which results in an average rate of 10 points per scan. At the end of each iteration, the target states are estimated using the standard k-means clustering algorithm.

The simulation result is shown in Fig. 4.5.

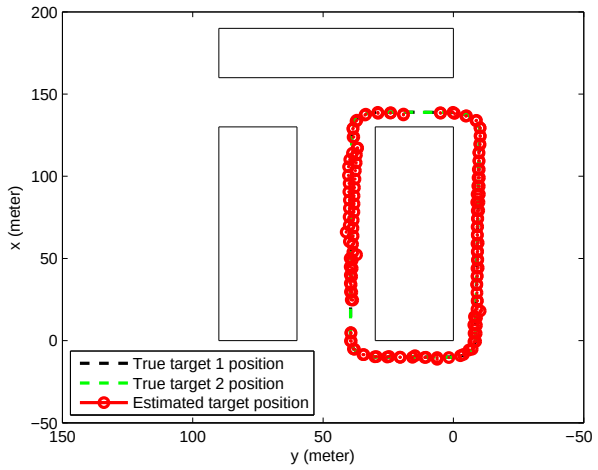

Figure 4.5: Tracking result of two targets moving in the same direction in the (x,y) plane using the PPHDF.

The simulation results for (x,y) position and velocity coordinates are separately shown in Figures 4.6 to 4.9.

The above simulation results in Figures 4.5 to 4.9 demonstrate that the multiple model PPHDF provides reasonably accurate tracking results for the multiple targets moving in urban terrain. Note that, as expected due to no signal returns, the target track is lost when a target enters the shadowing region.

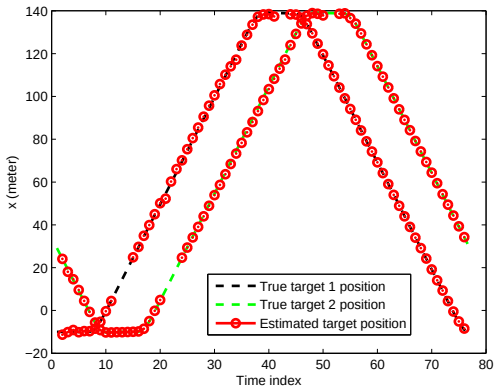

Figure 4.6: PPHDF estimated x-coordinate positions of two targets moving in the same direction.

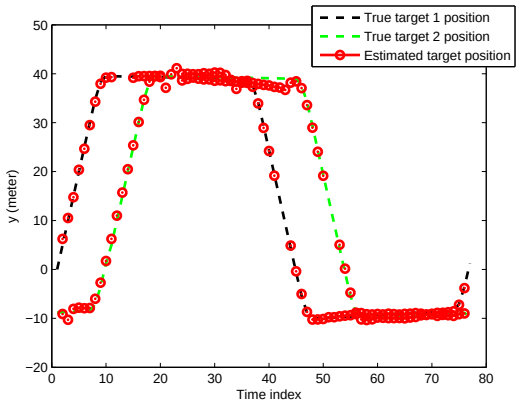

Figure 4.7: PPHDF estimated y-coordinate positions of two targets moving in the same direction.

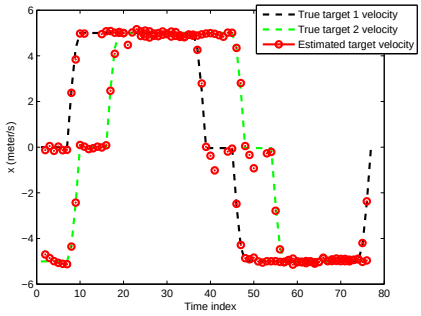

Figure 4.8: PPHDF estimated x-coordinate velocity of two targets moving in the same direction.

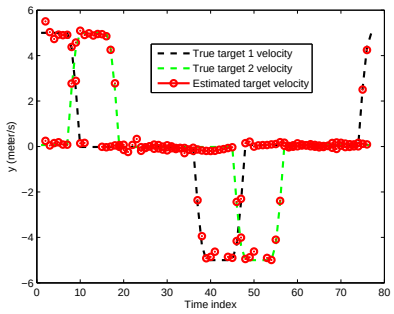

Figure 4.9: PPHDF estimated y-coordinate velocity of two targets moving in the same direction.

The mean-square error (MSE) performance based on 200 Monte Carlo simulation is shown in Figure 4.10 to Figure 4.13.

We note that the MSE error is small when target a is in the LOS region or LOS plus one-bounce region, and it becomes larger when a target is in the one-bounce or two one-bounce regions. The error grows large when a target is in the shadow region.

The estimated number of targets with time is shown in Figure 4.14, which is the average of 300 Monte Carlo simulations.

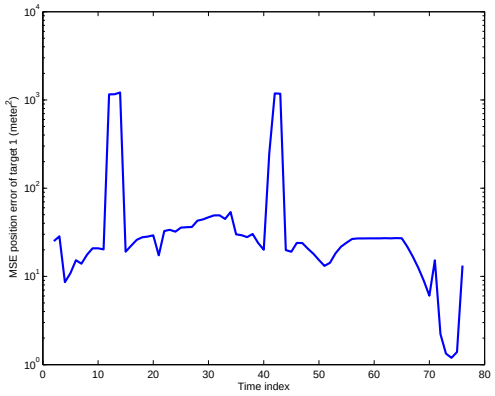

Figure 4.10: MSE position error of tracking result of two targets moving in the same direction of target 1 using MM-PPHDF.

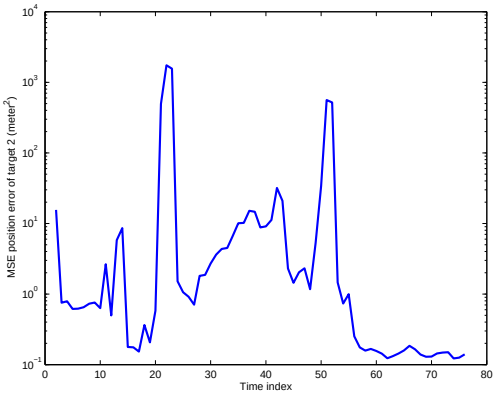

Figure 4.11: MSE position error of tracking result of two targets moving in the same direction of target 2 using MM-PPHDF.

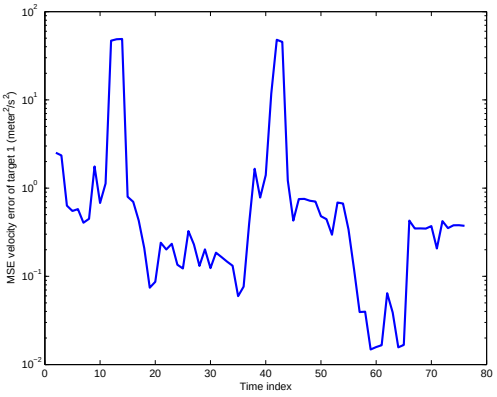

Figure 4.12: MSE velocity error of tracking result of two targets moving in the same direction of target 1 using MM-PPHDF.

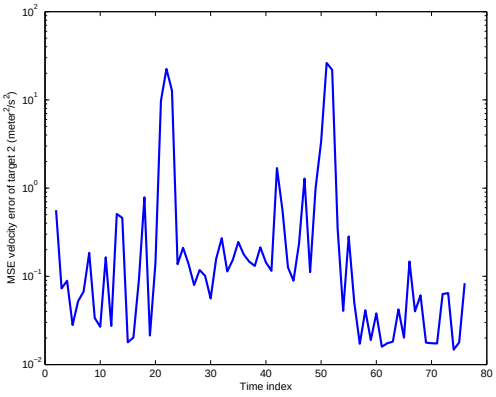

Figure 4.13: MSE velocity error of tracking result of two targets moving in the same direction of target 2 using MM-PPHDF.

4.4.2 Time-varying Number of Targets in Urban Terrain

Previously, example 1 basically illustrates the multiple model PPHDF can solve fixed targets moving in same direction. In order to further investigate the performance of multiple model PPHDF, we consider the following case, when the number of targets are time varying. Assuming initially ($k = 1$) there are two ground vehicles, which are moving in the same direction. Then, at $k = 5$, a new target born and moves in the opposite direction. Moreover, when at $k = 12$, one target dies, and $k = 15$, another target

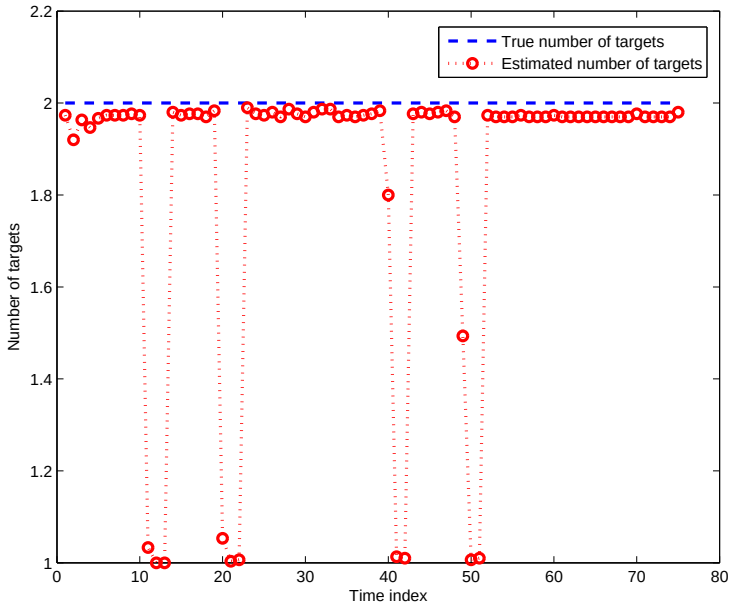

Figure 4.14: PPHDF estimated number of targets when 2 targets are moving in the same direction

dies. The road map as well as the targets trajectory can be found in Fig. 4.15 The target state is denoted by $\mathbf{x}_k = [x_k, \dot{x}_k, y_k, \dot{y}_k]^T$ with its position (x_k, y_k) and velocity $(\dot{x}_k, \dot{y}_k)$ in Cartesian coordinates and the measurements are noisy version of rangeCrange-rate measurements.

Each target can switch between the following linear state model

$$\mathbf{x}_k = \mathbf{F}_k \mathbf{x}_{k-1} + \mathbf{n}_k. \tag{4.15}$$

When at time k, the ith model index $\varpi_k^{(i)} = 1$, which indicates $\mathbf{F}_k$ is a NCV model given by

$$\mathbf{F}_k = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

When at time k, the ith model index $\varpi_k^{(i)} = 2$, which indicates $\mathbf{F}_k$ is a CT model

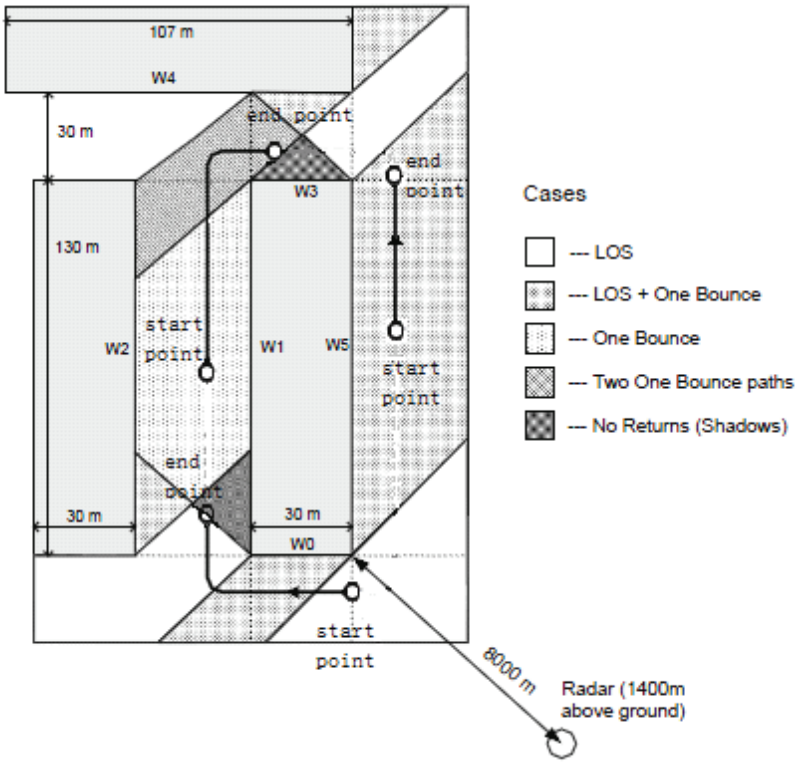

Figure 4.15: The road map of the test bench urban environment and trajectory of 3 targets using MM-PPHDF.

represented by

$$\mathbf{F}_2 = \begin{bmatrix} 1 & sin(\omega)/\omega & 0 & -(1-cos(\omega))/\omega \\ 0 & cos(\omega) & 0 & -sin(\omega) \\ 0 & (1-cos(\omega))/\omega & 1 & sin(\omega)/\omega \\ 0 & sin(\omega) & 0 & cos(\omega) \end{bmatrix}.$$

where $\omega = -2$ is the angular turning rate.

Moreover, a third model is needed for the opposite vehicle to make turning,

which is represented by

$$\mathbf{F}_k = \begin{bmatrix} 1 & sin(\omega)/\omega & 0 & -(1-cos(\omega))/\omega \\ 0 & cos(\omega) & 0 & -sin(\omega) \\ 0 & (1-cos(\omega))/\omega & 1 & sin(\omega)/\omega \\ 0 & sin(\omega) & 0 & cos(\omega) \end{bmatrix}.$$

where $\omega = 2$ is the angular turning rate, $\mathbf{v}_k$ is assumed to be a zero-mean, white Gaussian sequence with covariance matrix:

$$\mathbf{Q} = q \begin{bmatrix} \frac{1}{3} & \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{3} & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & 1 \end{bmatrix}$$

where $q = 0.04$ is a coefficient determining the process noise intensity. Especially, the surviving probability is assumed to be 0.95. When implementing PPHDF, we assign totally 4500 particles for tracking.

Beside, we assume that there are no spawning but new born target are assumed to following a Gaussian distribution in the simulation. Besides the clutter random finite set is considered to be Poisson random finite set, and uniformly distributed in the region $[-50,150] \times [-100,50]$. Moreover, the clutter density is assumed to be 3.33×10^{-4}, which results in an average rate of 10 points per scan. At the end of each iteration, the target state are estimated by using the standard k-means clustering algorithm.

The simulation result is shown in Fig. 4.16.

And the simulation results of x-coordinate positions and y-coordinate positions separately are shown in Fig. 4.17 and Fig. 4.18.

Besides, the estimation results of x-coordinate velocity and y-coordinate velocity are shown in Fig. 4.19 and Fig. 4.20.

The above simulation results from Fig. 4.16 to Fig. 4.20 are given by the multiple model PPHDF with the assumption of knowing which range-range pair corresponding to LOS or multipath signal return. It shows that the multiple model PPHDF

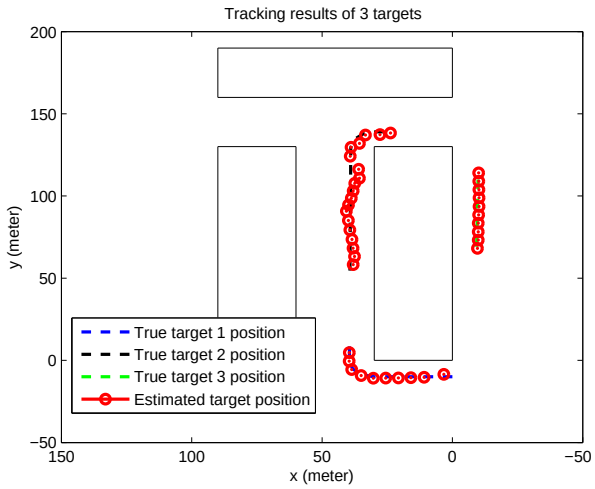

Figure 4.16: Tracking result of time varying targets moving in XY planes using MM-PPHDF.

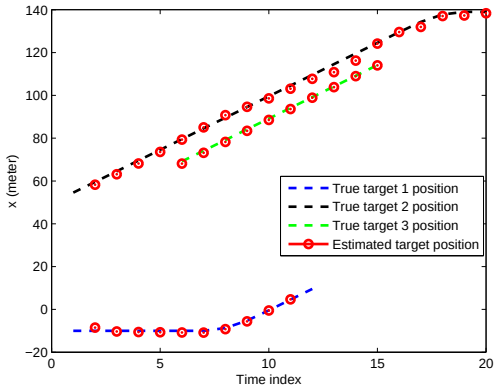

Figure 4.17: Tracking result of time varying targets: x-coordinate position using MM-PPHDF.

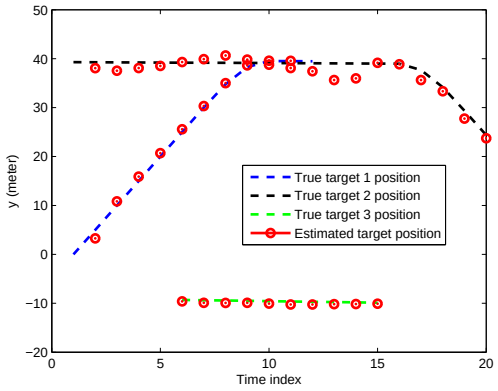

Figure 4.18: Tracking result of time varying targets: y-coordinate position using MM-PPHDF.

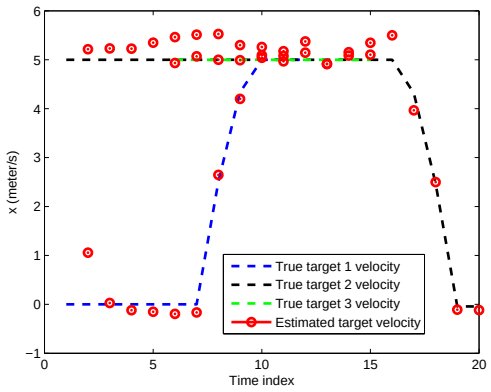

Figure 4.19: Tracking result of time varying targets: x-coordinate velocity using MM-PPHDF.

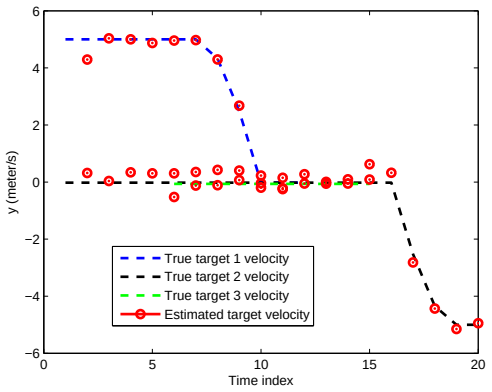

Figure 4.20: Tracking result of time varying targets: y-coordinate velocity using MM-PPHDF.

provides reasonably accurate estimation of urban multiple targets tracks with almost free of false tracks. In addition, because in the shadowing region, there is no signal return, so the target track is lost when target enters the shadowing region.

From all the above simulations, it is not hard to see, the multiple model PPHDF can efficiently overcome the problem brought by time varying number of targets and multiple model.

Moreover, the mean square error (MSE) performance of 100 Monte Carlo simulation is shown in Fig. 4.21 to Fig. 4.26.

The above Monte Carlo simulation results from Fig. 4.21 to Fig. 4.26 are given by the multiple model PPHDF with the assumption of knowing which range-range pair corresponding to LOS or multipath signal return. From which, it is shown the MSE error is small when target is in the LOS region or LOS plus one-bounce region, and becomes larger when target in one-bounce or two one-bounce area, in addition, the error grows larger when target is in shadowing region. Furthermore, in Fig. 4.23 and 4.24, there is a jump of MSE error at time $k = 12$, and the reason is in this case, the target 1 has entered the shadowing region, the particles in the shadowing region can effect the clustering for target 3, which makes the MSE error grows larger in this case. Besides, the MSE error decrease to normal at the next time step.

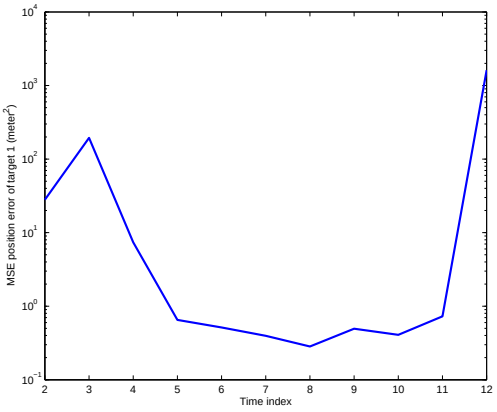

Figure 4.21: MSE position error of tracking result of time varying targets: target 1 using MM-PPHDF.

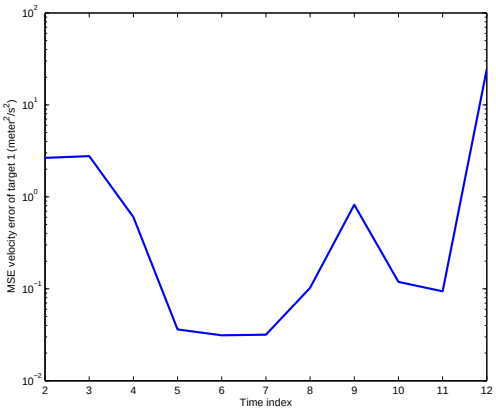

Figure 4.22: MSE velocity error of tracking result of time varying targets: target 1 using MM-PPHDF.

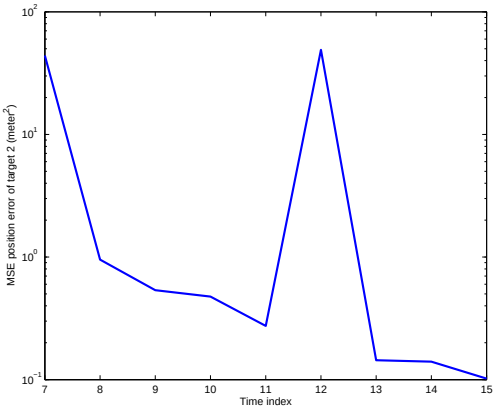

Figure 4.23: MSE position error of tracking result of time varying targets: target 2 using MM-PPHDF.

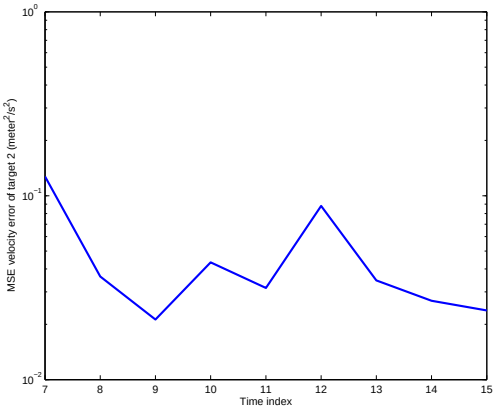

Figure 4.24: MSE velocity error of tracking result of time varying targets: target 2 using MM-PPHDF.

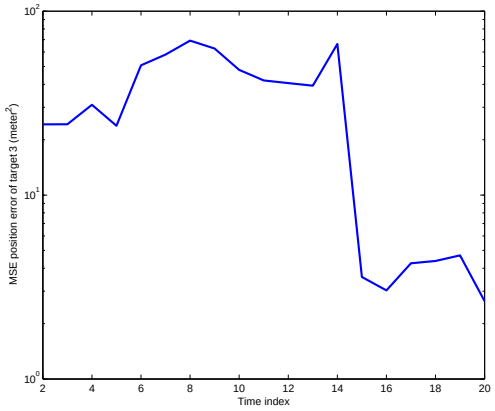

Figure 4.25: MSE position error of tracking result of time varying targets: target 3 using MM-PPHDF.

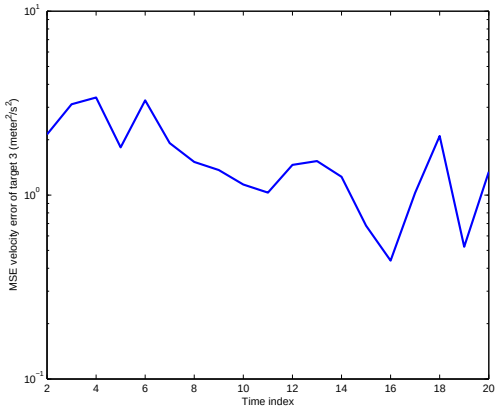

Figure 4.26: MSE velocity error of tracking result of time varying targets: target 3 using MM-PPHDF.

From Fig. 4.16 to Fig. 4.26, the simulation results demonstrate the multiple model PPHDF can accurately handle the multiple target tracking problem in urban terrain with the assumption of knowing which rang-range rate corresponding to LOS or multipath signal return.

Moreover, the simulation results of targets number is shown in Fig. 4.27, which is the average of 300 Monte Carlo simulations.

From Fig. 4.27, it is not hard to see, besides the shadowing region, in most case, there is no miss tracking.

4.4.3 Two Targets in Urban Terrain With Path-to-Measurement Association

Assuming there are two ground vehicles, whose starting points are marked in Fig. 4.3. Besides, they are moving in a loop trajectory, as marked in Fig. 4.3. The target state is denoted by $\mathbf{x}_k = [x_k, \dot{x}_k, y_k, \dot{y}_k]^T$ with its position (x_k, y_k) and velocity $(\dot{x}_k, \dot{y}_k)$ in Cartesian coordinates and the measurements are noisy version of rangeCrange-rate measurements.

Each target can switch between the following linear state model

$$\mathbf{x}_k = \mathbf{F}_k \mathbf{x}_{k-1} + \mathbf{n}_k. \tag{4.16}$$

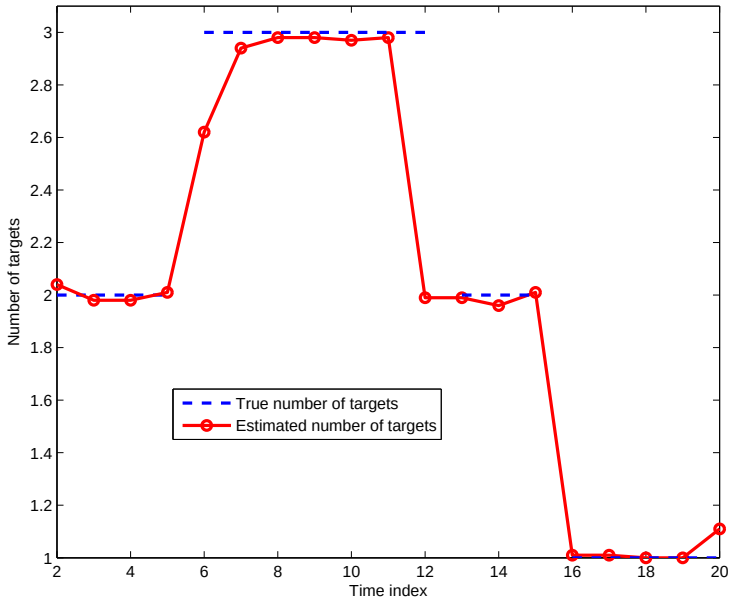

Figure 4.27: Number of targets of time varying targets using MM-PPHDF.

When at time k, the ith model index $\varpi_k^{(i)} = 1$, which indicates $\mathbf{F}_k$ is a NCV model given by

$$\mathbf{F}_k = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

When at time k, the ith model index $\varpi_k^{(i)} = 2$, which indicates $\mathbf{F}_k$ is a CT model represented by

$$\mathbf{F}_k = \begin{bmatrix} 1 & sin(\omega)/\omega & 0 & -(1-cos(\omega))/\omega \\ 0 & cos(\omega) & 0 & -sin(\omega) \\ 0 & (1-cos(\omega))/\omega & 1 & sin(\omega)/\omega \\ 0 & sin(\omega) & 0 & cos(\omega) \end{bmatrix}.$$

where $\omega = -2$ is the angular turning rate, $\mathbf{v}_k$ is assumed to be a zero-mean, white

Gaussian sequence with covariance matrix:

$$\mathbf{Q} = q \begin{bmatrix} \frac{1}{3} & \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{3} & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & 1 \end{bmatrix}$$

where $q = 0.04$ is a coefficient determining the process noise intensity. Especially, the surviving probability is assumed to be 0.95. When implementing MPPHDF, we assign each target 2000 particles.

Beside, we assume that there are no spawning and no new born target in the simulation. Besides the clutter random finite set is considered to be Poisson random finite set, and uniformly distributed in the region $[-50, 150] \times [-100, 50]$. Moreover, the clutter density is assumed to be 3.33×10^{-4}, which results in an average rate of 10 points per scan. At the end of each iteration, the target state are estimated by using the standard k-means clustering algorithm.

The simulation result is shown in Fig. 4.28.

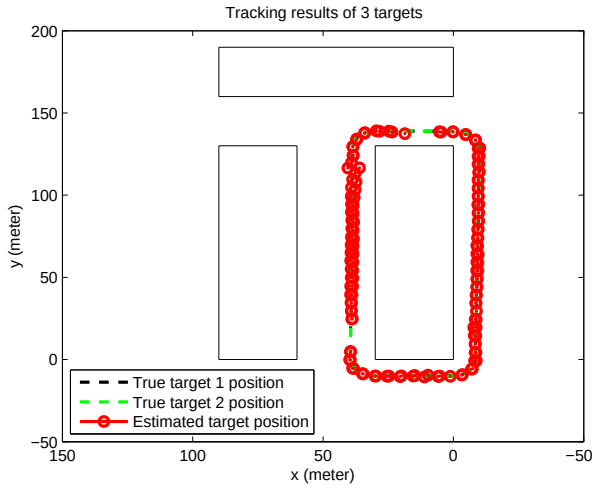

Figure 4.28: Tracking result of two targets moving in the same direction in XY planes using modified algorithm.

And the simulation results of x-coordinate positions and y-coordinate positions

separately are shown in Fig. 4.29 and Fig. 4.30.

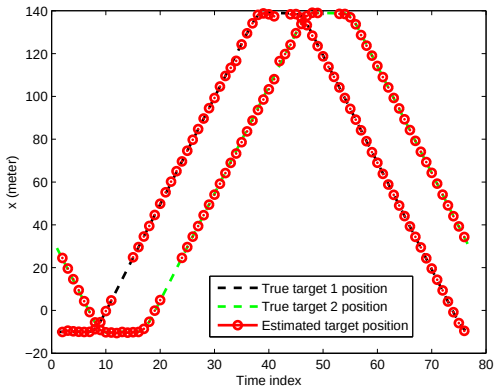

Figure 4.29: Tracking result of two targets moving in the same direction of x-coordinate position using modified algorithm.

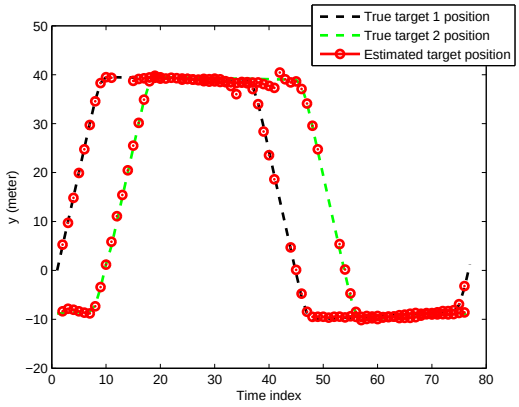

Figure 4.30: Tracking result of two targets moving in the same direction of y-coordinate position using modified algorithm.

Besides, the estimation results of x-coordinate velocity and y-coordinate velocity are shown in Fig. 4.31 and Fig. 4.32.

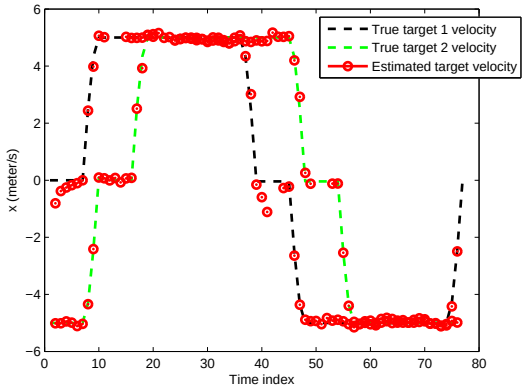

Figure 4.31: Tracking result of two targets moving in the same direction of x-coordinate velocity using modified algorithm.

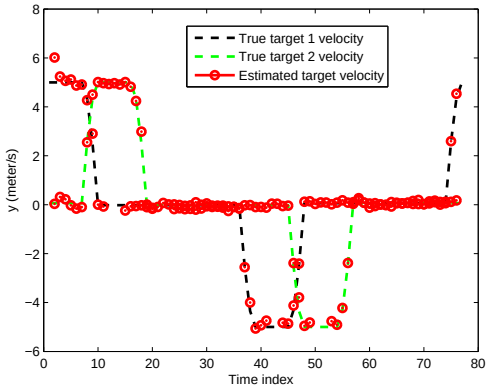

Figure 4.32: Tracking result of two targets moving in the same direction of y-coordinate velocity using modified algorithm.

The above simulation results from Fig. 4.28 to Fig. 4.32 are given by the multiple model MPPHDF with the assumption of knowing which range-range pair corresponding to LOS or multipath signal return. It shows that the multiple model MP-

PHDF provides reasonably accurate estimation of urban multiple targets tracks with almost free of false tracks. In addition, because in the shadowing region, there is no signal return, so the target track is lost when target enters the shadowing region.

Moreover, the mean square error (MSE) performance of 300 Monte Carlo simulation is shown in Fig. 4.33 to Fig. 4.36.

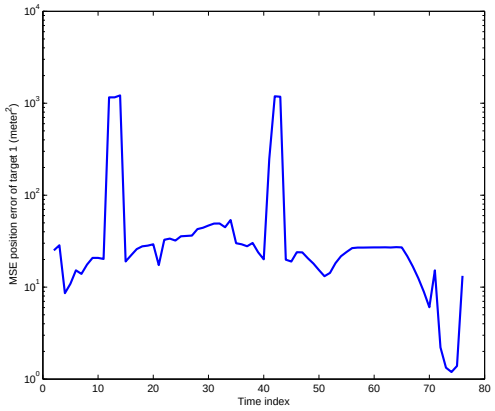

Figure 4.33: MSE position error of tracking result of two targets moving in the same direction of target 1 using modified algorithm.

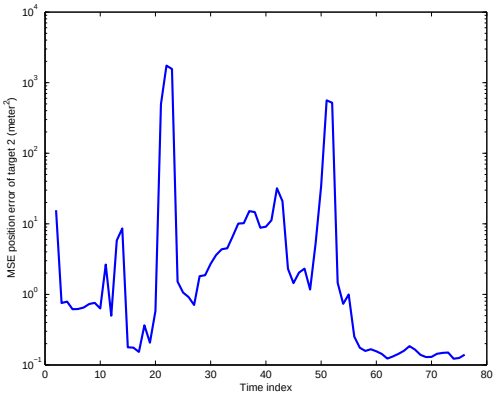

Figure 4.34: MSE position error of tracking result of two targets moving in the same direction of target 2 using modified algorithm.

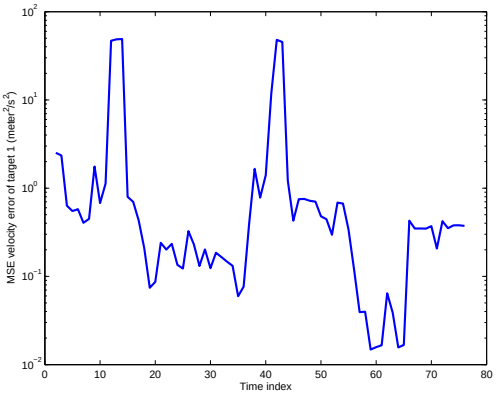

Figure 4.35: MSE velocity error of tracking result of two targets moving in the same direction of target 1 using modified algorithm.

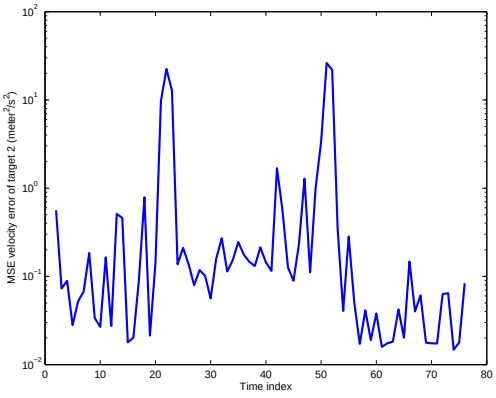

Figure 4.36: MSE velocity error of tracking result of two targets moving in the same direction of target 2 using modified algorithm.

The above Monte Carlo simulation results from Fig. 4.33 to Fig. 4.36 are given

by the MPPHDF with the assumption of knowing which range-range pair corresponding to LOS or multipath signal return. From which, it is shown the MSE error is small when target is in the LOS region or LOS plus one-bounce region, and becomes larger when target in one-bounce or two one-bounce area, in addition, the error grows large when target is in shadowing region.

From Fig. 4.28 to Fig. 4.36, the simulation results demonstrate the MPPHDF can accurately handle the multiple target tracking problem in urban terrain with the assumption of knowing which rang-range rate corresponding to LOS or multipath signal return.

Moreover, the simulation results of targets number is shown in Fig. 4.37, which is the average of 300 Monte Carlo simulations.

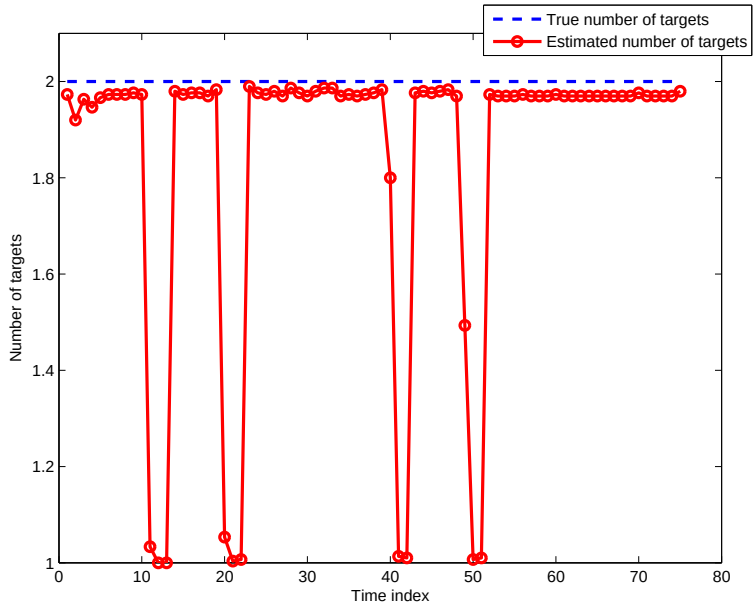

Figure 4.37: Number of targets of two targets moving in the same direction using modified algorithm.

From Fig. 4.37, it is not hard to see, besides the shadowing region, in most case, there is no miss tracking.

4.4.4 *Time-varying Number of Targets in Urban Terrain With Path-to-Measurement Association*

Previously, example 1 basically illustrates the MPPHDF can solve fixed targets moving in same direction. In order to further investigate the performance of MPPHDF, we consider the following case, when the number of targets are time varying. Assuming initially ($k = 1$) there are two ground vehicles, which are moving in the same direction. Then, at $k = 5$, a new target born and moves in the opposite direction. Moreover, when at $k = 12$, one target dies, and $k = 15$, another target dies. The road map as well as the targets trajectory can be found in Fig. 4.38 The target state is denoted by $\mathbf{x}_k =$

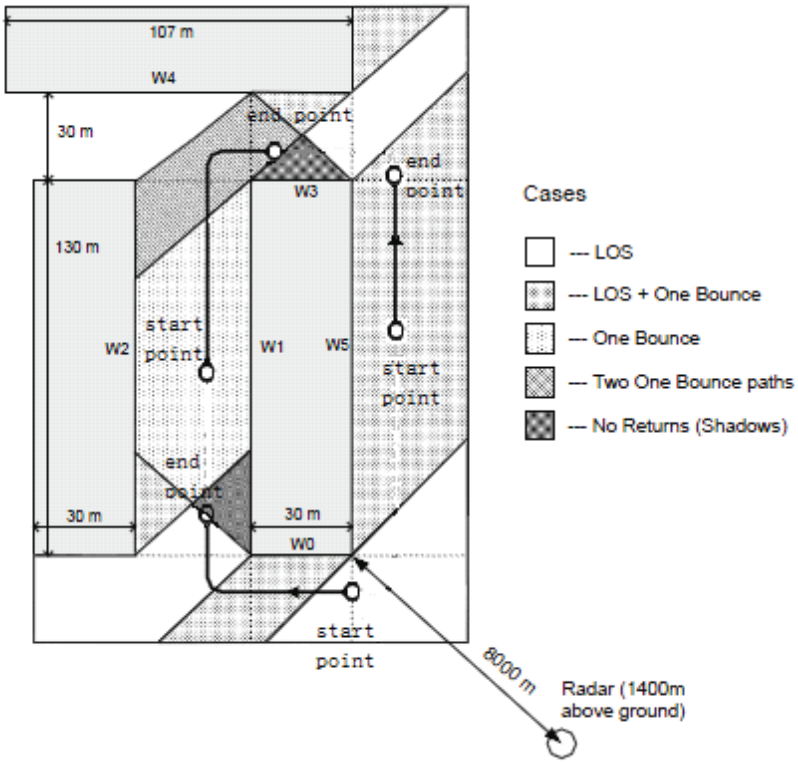

Figure 4.38: The road map of the test bench urban environment and trajectory of 3 targets.

$[x_k, \dot{x}_k, y_k, \dot{y}_k]^T$ with its position (x_k, y_k) and velocity $(\dot{x}_k, \dot{y}_k)$ in Cartesian coordinates and the measurements are noisy version of rangeCrange-rate measurements.

Each target can swith between the following linear state model

$$\mathbf{x}_k = \mathbf{F}_k \mathbf{x}_{k-1} + \mathbf{n}_k. \tag{4.17}$$

When at time k, the ith model index $\varpi_k^{(i)} = 1$, which indicates $\mathbf{F}_k$ is a NCV model given by

$$\mathbf{F}_k = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

When at time k, the ith model index $\varpi_k^{(i)} = 2$, which indicates $\mathbf{F}_k$ is a CT model represented by

$$\mathbf{F}_2 = \begin{bmatrix} 1 & sin(\omega)/\omega & 0 & -(1-cos(\omega))/\omega \\ 0 & cos(\omega) & 0 & -sin(\omega) \\ 0 & (1-cos(\omega))/\omega & 1 & sin(\omega)/\omega \\ 0 & sin(\omega) & 0 & cos(\omega) \end{bmatrix}.$$

where $\omega = -2$ is the angular turning rate.

Moreover, a third model is needed for the opposite vehicle to make turning, which is represented by

$$\mathbf{F}_k = \begin{bmatrix} 1 & sin(\omega)/\omega & 0 & -(1-cos(\omega))/\omega \\ 0 & cos(\omega) & 0 & -sin(\omega) \\ 0 & (1-cos(\omega))/\omega & 1 & sin(\omega)/\omega \\ 0 & sin(\omega) & 0 & cos(\omega) \end{bmatrix}.$$

where $\omega = 2$ is the angular turning rate, $\mathbf{v}_k$ is assumed to be a zero-mean, white Gaus-

sian sequence with covariance matrix:

$$\mathbf{Q} = q \begin{bmatrix} \frac{1}{3} & \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{3} & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & 1 \end{bmatrix}$$

where $q = 0.04$ is a coefficient determining the process noise intensity. Especially, the surviving probability is assumed to be 0.95. When implementing MPPHDF, we assign totally 4500 particles for tracking.

Beside, we assume that there are no spawning but new born target are assumed to following a Gaussian distribution in the simulation. Besides the clutter random finite set is considered to be Poisson random finite set, and uniformly distributed in the region $[-50, 150] \times [-100, 50]$. Moreover, the clutter density is assumed to be 3.33×10^{-4}, which results in an average rate of 10 points per scan. At the end of each iteration, the target state are estimated by using the standard k-means clustering algorithm.

The simulation result is shown in Fig. 4.39.

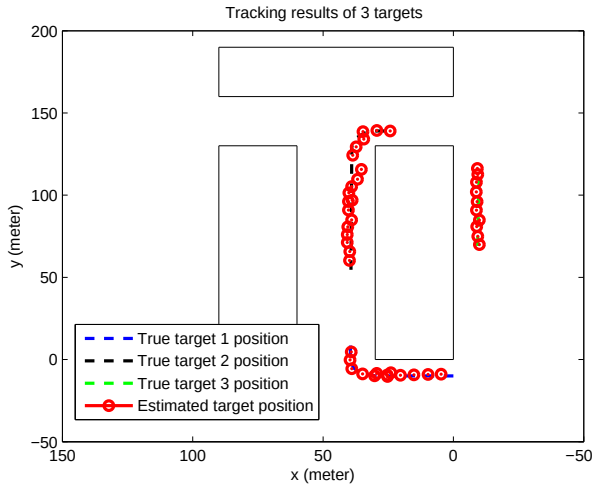

Figure 4.39: Tracking result of time varying targets moving in XY planes using modified algorithm.

And the simulation results of x-coordinate positions and y-coordinate positions

separately are shown in Fig. 4.40 and Fig. 4.41.

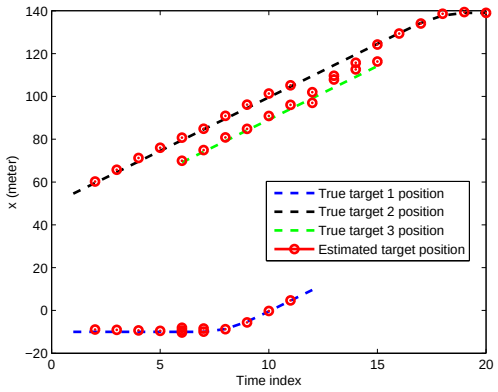

Figure 4.40: Tracking result of time varying targets: x-coordinate position using modified algorithm.

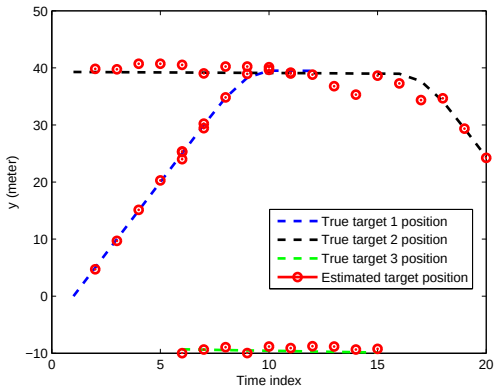

Figure 4.41: Tracking result of time varying targets: y-coordinate position using modified algorithm.

Besides, the estimation results of x-coordinate velocity and y-coordinate velocity are shown in Fig. 4.42 and Fig. 4.43.

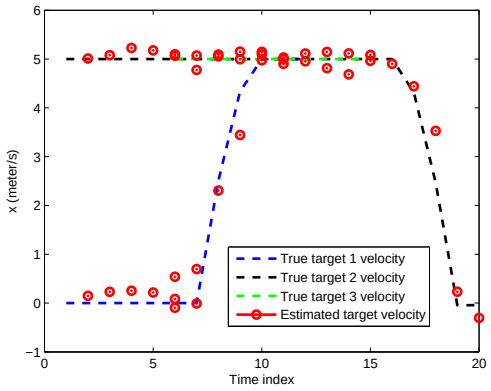

Figure 4.42: Tracking result of time varying targets: x-coordinate velocity using modified algorithm.

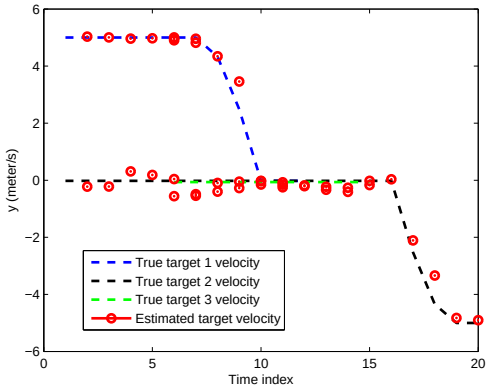

Figure 4.43: Tracking result of time varying targets: y-coordinate velocity using modified algorithm.

The above simulation results from Fig. 4.39 to Fig. 4.43 are given by the MPPHDF with the assumption of knowing which range-range pair corresponding to LOS or multipath signal return. It shows that the MPPHDF provides reasonably accurate estimation of urban multiple targets tracks with almost free of false tracks. In addition,

because in the shadowing region, there is no signal return, so the target track is lost when target enters the shadowing region.

From all the above simulations, it is not hard to see, the MPPHDF can efficiently overcome the problem brought by time varying number of targets and multiple model.

Moreover, the mean square error (MSE) performance of 100 Monte Carlo simulation is shown in Fig. 4.44 to Fig. 4.49.

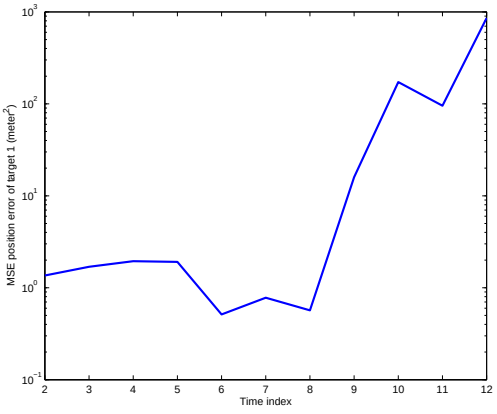

Figure 4.44: MSE position error of tracking result of time varying targets: target 1 using modified algorithm.

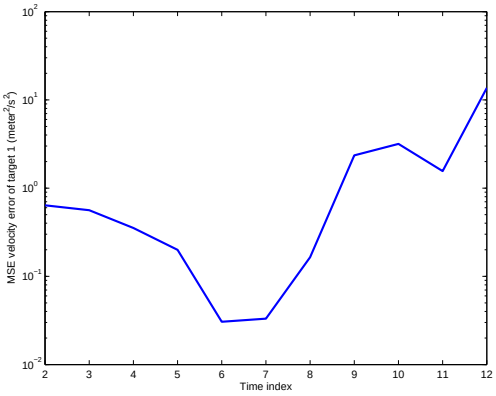

Figure 4.45: MSE velocity error of tracking result of time varying targets: target 1 using modified algorithm.

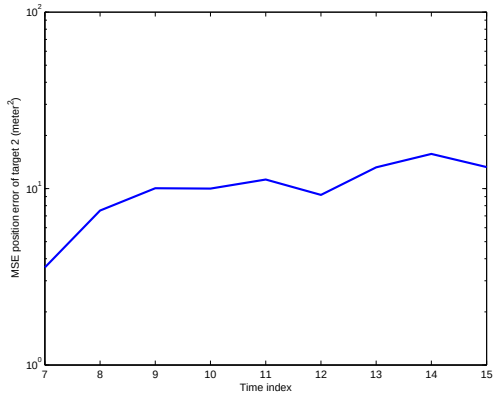

Figure 4.46: MSE position error of tracking result of time varying targets: target 2 using modified algorithm.

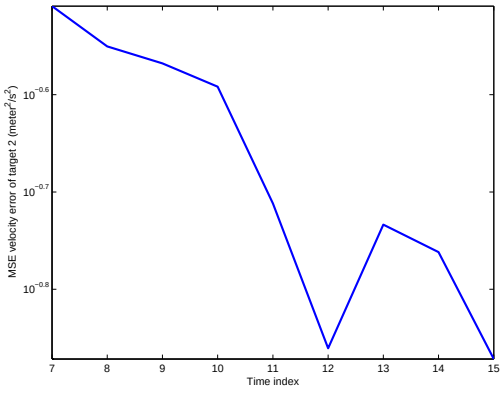

Figure 4.47: MSE velocity error of tracking result of time varying targets: target 2 using modified algorithm.

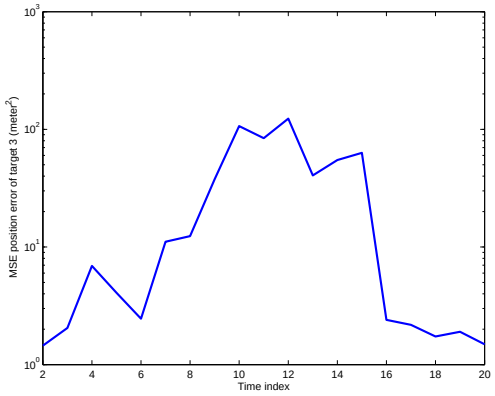

Figure 4.48: MSE position error of tracking result of time varying targets: target 3 using modified algorithm.

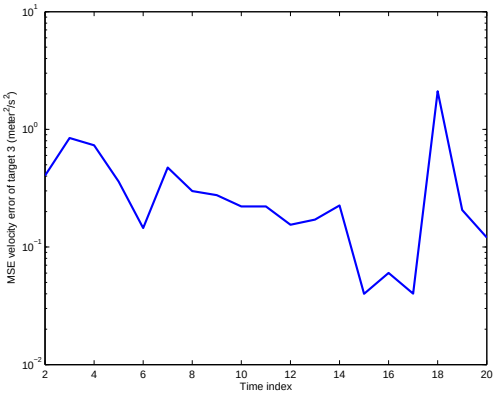

Figure 4.49: MSE velocity error of tracking result of time varying targets: target 3 using modified algorithm.

The above Monte Carlo simulation results from Fig. 4.44 to Fig. 4.49 are given by the MPPHDF with the assumption of knowing which range-range pair corresponding to LOS or multipath signal return. From which, it is shown the MSE error is small when target is in the LOS region or LOS plus one-bounce region, and becomes larger when target in one-bounce or two one-bounce area, in addition, the error grows larger when target is in shadowing region. Furthermore, in Fig. 4.46 and 4.47, there is a jump of MSE error at time $k = 12$, and the reason is in this case, the target 1 has entered the shadowing region, the particles in the shadowing region can effect the clustering for target 3, which makes the MSE error grows larger in this case. Besides, the MSE error decrease to normal at the next time step. In addition, from Fig. 4.48 and 4.49, the MSE error grows larger from time $k = 10$ to $k = 14$, the reason is that for MPPHDF, the information of which noisy range-range rate pair corresponding to LOS or multipath is lost, moreover, the target is in the one-bounce region, i.e., LOS is not available, and, measurements from other targets which has the close value can effect the tracking result, and the MSE decrease to normal when all the other targets die at $k = 15$.

From Fig. 4.39 to Fig. 4.49, the simulation results demonstrate the MPPHDF can accurately handle the multiple target tracking problem in urban terrain with the as-

sumption of knowing which rang-range rate corresponding to LOS or multipath signal return.

Moreover, the simulation results of targets number is shown in Fig. 4.50, which is the average of 300 Monte Carlo simulations.

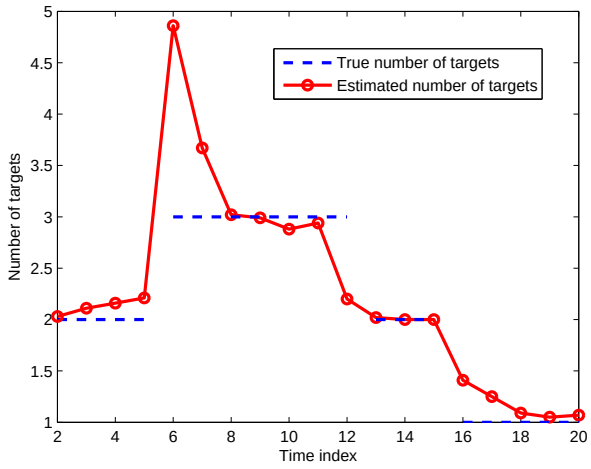

Figure 4.50: Number of targets of time varying targets using modified algorithm.

From Fig. 4.50, it is not hard to see, besides the shadowing region, in most case, there is no miss tracking.

Chapter 5

Conclusions and Future Work

The tracking performance of conventional tracking systems begins to deteriorate or fail when they are used for target tracking in the urban environments. As many warfare scenarios happen in urban environments, radar tracking systems need to be modified and improved to maintain their performance in such environment. During the past decade, lots of research work has been accomplished in the area of target tracking in urban environments, however, due to computationally intensive brought by conventional data association techniques as well as the requirement of very strict conditions for efficient performance, multiple target tracking (MTT) problem in urban environment was not extensively discussed.

The state-space model for target tracking in urban environment is introduced in Chapter 2. In urban environments, single state model is not sufficient for describing a maneuvering target, the more robust state equations are comprised of different maneuvering models, i.e., multiple dynamic state models. Based on, a 3-D urban environment multi-path geometry, we derive the corresponding multiple measurement equations for different scenarios such as LOS, multi-path, shadowing, and also the clutter model in urban terrain scenario is presented.

Chapter 3 describe the probability hypothesis density filter (PHDF), which is based on the theory of random finite sets. Moreover, unlike data association methods, it can thus be used to estimate the number of targets as well as their corresponding tracks, when combined with clustering algorithm. The particle filter is used to implement the PHD due to the additional degrees of nonlinearity in the multipath.

Furthermore, in Chapter 4, the multiple-model estimator is combined with the PPHDF algorithm to overcome the problems resulted from multiple target state models. Besides, a modified multi-model multi-target PPHDF is shown to provide the capability of automatically and adaptively estimating the measurement types available at each

time step. Specifically, the new algorithm allows measurement-to-track nonlinearity associations such that the best matched measurement can be used at each time step, resulting in improved radar coverage and scene visibility. In addition, numerical simulations including both two targets moving in the same trajectory and time varying number of targets moving in same or opposite directions are used to demonstrate the effectiveness of the multi-model multi-target PPHDF as well as the modified algorithm in improving tracking performance, both for tracking multiple targets and targets in clutter.

As we can see from the simulation results, the estimated trajectories of the both multi-model multi-target PPHDF and modified algorithm are closely match with true target states. When the target moves from a LOS region to a multipath region or a shadowing region, the estimated error increases, and the MSE decreases as it travels back to the LOS region.

In summary, an algorithm named multi-model multi-target PPHDF is proposed and its performance is investigated. We modified the PPHDF techniques with multiple model estimator, and design the urban measurement model to successfully achieve multiple target tracking in urban environments.

Our study focus on the general PHDF for multiple target tracking in the urban environments. The scope of further study on PHDF in multi-target tracking includes.

1. In this thesis, we do not considered the adaptive waveform design for MPPHDF. In future, the waveform design can be added into the urban MTT problem.

2. In this thesis, the proposed MPPHDF uses the first moment information for estimating the target number. However another technique known as CPHD technique considers higher order information, which is expected to improve the performance of estimating target number.

REFERENCES

[1] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Boston, MA: Artech House, 2004.

[2] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[3] [Online]. Available: http://www.cs.unc.edu/ welch/media/pdf/ kalman intro.pdf

[4] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*. New York: Academic Press, 1988.

[5] E. J. Baranoski, "Urban operations: The new frontier for radar," in *DARPA Systems and Technology Symposium, DARPA Special Projects Office*, Anaheim, CA, USA, 2005, pp. 155–159.

[6] P. R. Barbosa, E. K. P. Chong, S. Suvarova, and B. Moran, "Multitarget-multisensor tracking in an urban environment: A closed-loop approach," in *Proceedings of SPIE -The International Society for Optical Engineering*, vol. 6969, 2008.

[7] T. Trueblood, "Multipath exploitation radar for tracking in urban terrain," Master's thesis, Arizona State University, 2009.

[8] J. L. Krolik, J. Farrell, and A. Steinhardt, "Exploiting multipath propagation for GMTI in urban environments," in *Proceedings of IEEE Conference Radar*, 2006, pp. 65–68.

[9] [Online]. Available: http://www.darpa.mil/STO/solicitations/baa09-01/presentations/MER Industry Day.pdf

[10] M. Mertens and M. Ulmke, "Precision GMTI tracking using road constraints with visibility information and a refined sensor model," in *IEEE Radar Conference*, 2008, pp. 1–6.

[11] B. Chakraborty, Y. Li, J. J. Zhang, T. Trueblood, A. Papandreou-Suppappola, and D. Morrell, "Multipath exploitation with adaptive waveform design for tracking in urban terrain," in *Proceedings of IEEE International Conference on Acoustics Speech and Signal Processing*, 2010, pp. 3894–3897.

[12] B. Rigling, "Urban RF multipath mitigation," *Radar, Sonar Navigation, IET*, vol. 2, no. 6, pp. 419 –425, December 2008.

[13] A. Papandreou-Suppappola, J. Zhang, B. Chakraborty, Y. Li, D. Morrell, and S. P. Sira, *Waveform Diversity for Advanced Radar Systems*, A. D. F. Givin and L. Patton, Eds. IET Peter Peregrinus, 2011.

[14] B. Chakraborty, J. J. Zhang, A. Papandreou-Suppappola, and D. Morrell, "Urban terrain tracking in high clutter with waveform-agility," in *International Conference on Acoustics, Speech and Signal Processing*, Prague, May 2011.

[15] ——, "Waveform-agile MIMO radar for urban terrain tracking." Sedona AZ: IEEE Digital Signal Processing Workshop, 2011.

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

[17] Y. Li, S. P. Sira, A. Papandreou-Suppappola, D. Cochran, and L. L. Scharf, "Maximizing detection performance with waveform design for sensing in heavy sea clutter," in *IEEE 14th Workshop on Statistical Signal Processing*, 2007, pp. 249–253.

[18] J. Li and P. Stoica, "MIMO radar with colocated antennas," *IEEE Signal Processing Magazine*, vol. 24, no. 5, pp. 106–114, 2007.

[19] J. Li, L. Xu, P. Stoica, K. W. Forsythe, and D. W. Bliss, "Range compression and waveform optimization for MIMO radar: A Cramer-Rao bound based study," *IEEE Transactions on Signal Processing*, vol. 56, no. 1, pp. 218–232, 2008.

[20] L. Xu, J. Li, P. Stoica, K. W. Forsythe, and D. W. Bliss, "Waveform optimization for MIMO radar: A Cramer-Rao bound based study," in *IEEE International Conference on Acoustic, Speech and Signal Processing*, vol. 2, 2007.

[21] A. M. Haimovich, R. S. Blum, and L. J. Cimini, "MIMO radar with widely separated antennas," *IEEE Signal Processing Magazine*, vol. 25, no. 1, pp. 116–129, 2008.

[22] E. Fishler, A. Haimovich, R. S. Blum, J. Cimini, L. J., D. Chizhik, and R. A. Valenzuela, "Spatial diversity in radars-models and detection performance," *IEEE Transactions on Signal Processing*, vol. 54, no. 3, pp. 823–838, 2006.

[23] J. Li and P. Stoica, *MIMO Radar Signal Processing*. Wiley, 2009.

[24] G. W. Pulford and R. J. Evans, "A multipath data association tracker for over-the-horizon radar," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 4, pp. 1165–1183, 1998.

[25] T. Sathyan, D. Humphrey, and M. Hedley, "Target tracking in multipath environments-An algorithm inspired by data association," in *12th International Conference on Information Fusion,*, 2009, pp. 1650–1657.

[26] S. Blackman, *Multiple Target Tracking with Radar Applications*, Norwood, Ed. Artech House, 1986.

[27] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*, Norwood, Ed. Artech House, 1999.

[28] I. J. Cox and S. L. Hingorani, "An efficient implementation of Reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 2, pp. 138–150, 1996.

[29] I. Goodman, R. P. S. Mahler, and H. Nguyen, *Mathematics of Data Fusion*. Springer, 1997.

[30] L. D. Stone, C. A. Barlow, and T. L. Corwin, *Bayesian Multiple Target Tracking*. Norwood, MA: Artech House, 1999.

[31] Y. Bar-Shalom and X. R. Li, *Multitarget-Multisensor Tracking: Principles and Techniques.* Storrs: YBS Publishing, 1995.

[32] T. Kurien, *Multitarget-Multisensor Tracking: Advanced Applications*. Norwood, MA: Artech House, 1990.

[33] D. Reid, "An algorithm for tracking multiple targets," *IEEE Transactions on Automatic Control*, vol. 24, no. 6, pp. 843–854, 1979.

[34] R. A. Singer, "Estimating optimal tracking filter performance for manned maneuvering targets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-6, no. 4, pp. 473–483, 1970.

[35] D. B. Reid, "The application of multiple target tracking theory to ocean surveillance," in *IEEE Conference on Decision and Control*, vol. 18, 1979, pp. 1046–1052.

[36] D. J. Salmond, D. Fisher, and N. J. Gordon, "Tracking in the presence of intermittent spurious objects and clutter," in *SPIE Conference on Signal and Data Processing of Small Targets*, 1998.

[37] H. A. P. Blom and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with Markovian switching coefficients," *IEEE Transactions on Automatic Control*, vol. 33, no. 8, pp. 780–783, 1988.

[38] Y. Bar-Shalom, F. Daum, and J. Huang, "The probabilistic data association filter," *IEEE Control Systems*, vol. 29, no. 6, pp. 82–100, 2009.

[39] T. Kirubarajan and Y. Bar-Shalom, "Probabilistic data association techniques for target tracking in clutter," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 536–557, 2004.

[40] T. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Sonar tracking of multiple targets using joint probabilistic data association," *IEEE Journal of Oceanic Engineering*, vol. 8, no. 3, pp. 173–184, 1983.

[41] R. P. S. Mahler, *An Introduction to Multisource-Multitarget statistics and Its Applications.* Lockheed Martin Technical Monograph, March 2000.

[42] ——, *Random Set Theory for Target Tracking and Identification, Data Fusion Handbook*, D. Hall and J. Llinas, Eds. CRC Press, 2001.

[43] ——, *Random Sets: Theory and Applications*. Springer, 1997.

[44] ——, "Multitarget Bayes filtering via first-order multitarget moments," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1152–1178, 2003.

[45] ——, "Multitarget moments and their applications to multitarget tracking," in *Proceedings of the Workshop on Estimation, Tracking and Fusion*, Monterey, CA, May, 2001, pp. 134–166.

[46] ——, ""statistics 101" for multisensor, multitarget data fusion," *IEEE Aerospace and Electronic Systems Magazine*, vol. 19, no. 1, pp. 53–64, 2004.

[47] H. Sidenbladh and S.-L. Wirkander, "Tracking random sets of vehicles in terrain," in *Proceeding of Conference on Computer Vision and Pattern Recognition Workshop*, 2003, p. 98.

[48] H. Sidenbladh, "Multi-target particle filtering for the probability hypothesis density," in *International Conference on Information Fusion*, 2003, pp. 800–806.

[49] D. E. Clark, K. Panta, and B.-N. Vo, "The GM-PHD filter multiple target tracker," in *Proceedings of 9th International Conference on Information Fusion*, 2006, pp. 1–8.

[50] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo implementation of the PHD filter for multi-target tracking," in *Proceedings of the Sixth International Conference of Information Fusion*, 2003, pp. 792–799.

[51] D. E. Clark and J. Bell, "Multi-target state estimation and track continuity for the particle PHD filter," *IEEE Transactions on Aerospace and Electronic Systems,*, vol. 43, no. 4, pp. 1441 –1453, 2007.

[52] K. Panta, B.-N. Vo, and S. Singh, "Improved probability hypothesis density (PHD) filter for multitarget tracking," in *Proceedings IEEE Third International Conference on Intelligent Sensing and Information Processing*, 2005, pp. 213–218.

[53] T. Zajic and R. P. S. Mahler, "A particle-systems implementation of the PHD multitarget tracking filter," in *Proceedings of SPIE, Signal Processing, Sensor Fusion and Target Recognition*, 2003, pp. 291–299.

[54] M. Tobias and A. D. Lanterman, "Probability hypothesis density-based multitarget tracking with bistatic range and Doppler observations," in *IEE Radar, Sonar and Navigation*, vol. 152, no. 3, 2005, pp. 195–205.

[55] K. Panta, B.-N. Vo, S. Singh, and A. Doucet, "Probability hypothesis density filter versus multiple hypothesis tracking," in *Proceedings of SPIE*, vol. 5429, 2004, pp. 284–295.

[56] K. Panta, B.-N. Vo, and S. Singh, "Novel data association schemes for the probability hypothesis density filter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 2, p. 556, 2007.

[57] X.-R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. Part I: Dynamic models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, 2003.

[58] R. Mahler, "An introduction to multisource-multitarget statistics and applications," Lockheed Martin Technical Monograph, Tech. Rep., 2000.

[59] D. Daley and D. Vere-Jones., *An Introduction to the Theory of Point Process*, G. Berlin, Ed. Springer Verlag, 1988.

[60] B.-N. Vo, A. Pasha, and H. D. Tuan, "A Gaussian mixture PHD filter for nonlinear jump Markov models," in *Proceedings of 45th IEEE Conference on Decision and Control*, 2006, pp. 3162–3167.

[61] S. A. Pasha, B.-N. Vo, H. D. Tuan, and W.-K. Ma, "A Gaussian mixture PHD filter for jump Markov system models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 3, pp. 919–936, 2009.

[62] B.-N. Vo, S. Singh, and A. Doucet, "Random finite sets and sequential Monte Carlo methods in multi-target tracking," in *Proceedings of IEEE Radar Conference*, 2003, pp. 486–491.

[63] D. E. Clark and J. Bell, "Bayesian multiple target tracking in forward scan sonar images using the PHD filter," *IEE Proceedings of Radar, Sonar and Navigation*, vol. 152, no. 5, pp. 327–334, 2005.

[64] K. Punithakumar, T. Kirubarajan, and A. Sinha, "A multiple model probability hypothesis density filter for tracking maneuvering targets." in *Signal and Data Processing of Small Targets, O. E. Drummond (Ed.), Proceedings of SPIE*, vol. 5428, 2004, pp. 113–121.

[65] Y. Boers and J. N. Driessen, "Interacting multiple model particle filter," *IEE Proceedings -Radar, Sonar and Navigation*, vol. 150, no. 5, pp. 344–349, 2003.

[66] S. McGinnity and G. W. Irwin, "Multiple model bootstrap filter for maneuvering target tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 36, no. 3, pp. 1006–1012, 2000.

[67] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan, "Interacting multiple model methods in target tracking: A survey," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 1, pp. 103–123, 1998.

[68] J. Gu, J. Zhou, and X. Chen, "An enhancement of k-means clustering algorithm," in *Proceedings of 2009 International Conference on Business Intelligence and Financial Engineering*, 2009, pp. 237–240.