# Bayesian Nonparametric Modeling and Inference for Multiple Object Tracking

by

Bahman Moraffah

A Dissertation Presented in Partial Fulfillment
of the Requirement for the Degree
Doctor of Philosophy

Approved July 2019 by the
Graduate Supervisory Committee:

Antonia Papandreou-Suppappola, Chair
Daniel W. Bliss
Christ D. Richmond
Gautam Dasarathy

ARIZONA STATE UNIVERSITY

August 2019

# ABSTRACT

The problem of multiple object tracking seeks to jointly estimate the time-varying cardinality and trajectory of each object. There are numerous challenges that are encountered in tracking multiple objects including a time-varying number of measurements, under varying constraints, and environmental conditions. In this thesis, the proposed statistical methods integrate the use of physical-based models with Bayesian nonparametric methods to address the main challenges in a tracking problem. In particular, Bayesian nonparametric methods are exploited to efficiently and robustly infer object identity and learn time-dependent cardinality; together with Bayesian inference methods, they are also used to associate measurements to objects and estimate the trajectory of objects. These methods differ from the current methods to the core as the existing methods are mainly based on random finite set theory.

The first contribution proposes dependent nonparametric models such as the dependent Dirichlet process and the dependent Pitman-Yor process to capture the inherent time-dependency in the problem at hand. These processes are used as priors for object state distributions to learn dependent information between previous and current time steps. Markov chain Monte Carlo sampling methods exploit the learned information to sample from posterior distributions and update the estimated object parameters.

The second contribution proposes a novel, robust, and fast nonparametric approach based on a diffusion process over infinite random trees to infer information on object cardinality and trajectory. This method follows the hierarchy induced by objects entering and leaving a scene and the time-dependency between unknown object parameters. Markov chain Monte Carlo sampling methods integrate the prior distributions over the infinite random trees with time-dependent diffusion processes to update object states.

The third contribution develops the use of hierarchical models to form a prior for statistically dependent measurements in a single object tracking setup. Dependency among the sensor measurements provides extra information which is incorporated to achieve the optimal tracking performance. The hierarchical Dirichlet process as a prior provides the required flexibility to do inference. Bayesian tracker is integrated with the hierarchical Dirichlet process prior to accurately estimate the object trajectory.

The fourth contribution proposes an approach to model both the multiple dependent objects and multiple dependent measurements. This approach integrates the dependent Dirichlet process modeling over the dependent object with the hierarchical Dirichlet process modeling of the measurements to fully capture the dependency among both object and measurements. Bayesian nonparametric models can successfully associate each measurement to the corresponding object and exploit dependency among them to more accurately infer the trajectory of objects. Markov chain Monte Carlo methods amalgamate the dependent Dirichlet process with the hierarchical Dirichlet process to infer the object identity and object cardinality.

Simulations are exploited to demonstrate the improvement in multiple object tracking performance when compared to approaches that are developed based on random finite set theory.

*To my beloved Dad, Mom, and Sister and in memory of my Grandma*

# ACKNOWLEDGEMENTS

*The enchanting charms of this sublime science reveal*
*only to those who have the courage to go deeply into it.*
*Carl Friedrich Gauss*

*My thesis is simple:*
*probability does not exist.*
*Bruno de Finetti*

*Be approximately right*
*rather than exactly wrong.*
*John Tukey*

*There are three kinds of lies:*
*lies, damned lies, and statistics.*
*Attributed to Benjamin Disraeli by Mark Twain*

*First and foremost, I would like to express my deepest appreciation to my advisor, Professor Antonia Papandreou-Suppappola– for her support, mentorship, encouragement, and inspiration. Her patience, kindness, and perception during my incredibly onerous journey taught me to work hard and never give up. She believed in me and made me believe in myself. I never forget her teaching me how to conquer difficulties to achieve what I wished for. She not only continually and convincingly showed me the sprit of research but also taught me how to be a teacher. I am enormously thankful for her letting me find my passion in research. Without her persistent help, this thesis would not have been possible.*

I also would like to thank my dissertation committee, Professor Daniel Bliss, Professor Christ Richmond, and Professor Gautam Dasarathy for a number of useful discussions over the years. I also would like to thank Dr Oliver Kosut for his valuable comments and discussions. Indeed, my thanks go out to all of the other colleagues with whom I have had the great pleasure of discussing or collaborating on various research topics. I would also like to thank SPAS group at Arizona State University; Dr. Narayan Kovvali, Dr. Alexander Maurer, Dr. John Kota, Cesar Brito, Bindya Venkatesh, Vineet Gattani, Srija Chakraborty, and Subhasish Das. I also want to thank Dr. Seth Suppappola for being an incredible, heartfelt, and caring friend.

I also wish to thank everyone in my research communities for their support and criticism. I wish to thank the department of Electrical Engineering at Arizona State University, specifically Dr. Joseph Palais for supporting me over years, Dr. David Allee for trusting me with a rare opportunity of instructor position for a graduate student over the past two years. I thank Lynn Pratt, Toni Mengert, Jenna Snowberger, Clayton Javurek for their constant help throughout these years.

The successes I have had in my many, many years as a student are mainly due to the love and encouragement of my family. And I am forever in their debt. I cannot thank my parents enough for giving me the opportunity to freely pursue my interests, career path, and otherwise. They have inspired me to continue to strive to be the best version of myself everyday. I am appreciative of my father for supporting me and being there for me. He has given me all the riches that life could offer and without him, I would never be where I stand today. I am indebted to my mother for listening, consoling, and assuring me that everything is going to be fine. I am grateful for everything my parents have done for me through all my years. I also would like to thank my sister, Raha– for being the best sister I could ask for. I have enormously enjoyed and benefited from our late night discussions on trending topics. Also, many *thanks to my uncle who has always encouraged and believe in me even when I did not. Finally, a big shout-out to my friends– thanks for sticking with me and helping me grow.*

# TABLE OF CONTENTS

Page

CHAPTER

# LIST OF TABLES

## LIST OF FIGURES

Figure Page

Figure Page

Figure Page

# Chapter 1

# INTRODUCTION

Multi-object tracking (MOT) refers to the problem of jointly estimating the time-varying cardinality and trajectories of multiple objects from noisy or cluttered measurements. With the development of the Kalman filter (1960), the object tracking problem became an active area of research. This area of research was primarily focused on the problem of single object tracking; however, with new advancements in technology, computational and embedded systems, this problem has rapidly grown to a multi-object tracking problem. The continued growth of multi-object tracking may be attributed to human needs and has drawn enormous attention in recent years. The multi-object tracking problem has found various applications in different areas of research including computer vision [1–4], driver assistance [5, 6], surveillance [7], image processing [8–10], remote sensing [4, 11], robotics  [12], and radar target tracking [13–15] .

## 1.1 Overview of Methods and Challenges

Despite advancement in the field of multi-object tracking, several problems have remained unclear. To see why the MOT problem is challenging, consider an environment in which multiple moving targets use different types of radar on a multimodal system under high clutter and high noise conditions. At each time step, targets can leave the scene and some new targets may come to the scene. Some of the challenges that are imposed by this problem include the time-dependent cardinality of objects, unordered measurements, unknown measurement-to-object association, and the association between object and the estimated object state.

Regardless of these difficulties which makes it typically impossible to directly label the objects, there have been various attempts to address the challenges in the MOT problem [16]. Historically, Bayesian methods have been used to track a single object, however, these methods become extremely complicated, if not impossible, when there is a multiple numbers of objects to track simultaneously. The simplest MOT algorithm is the nearest-neighbor Kalman filter. This technique updates the object state estimate only through the measurements that are in the statistical vicinity of the predicted track. Some variants of this approach include the strongest neighbor filter that considers the signal-to-noise ratio (SNR) to address the association ambiguity, and 2-dimensional (2-D) assignment algorithms in which an assignment problem accounts for the distances between all measurement and all tracks. This method employs a Kalman filter to update; however, it considers data association decisions one scan at each time step and encapsulates all previously collected data by a set of track estimates and their covariances [17].

Methods depending on maximum likelihood (ML) [18] or maximum a posteriori (MAP) estimation [19] have also been developed. These estimation approaches integrate the object labeling uncertainties with multiple hypotheses tracking algorithms [20]. Algorithms developed to perform this task include the Viterbi algorithm [21], the EM algorithm [22], network theoretic algorithms [23], and set partitioning [24, 25].

More than two decades ago, first order approximation models such as the joint probabilistic data association filter (JPDAF) and multiple hypothesis tracking (MHT) were introduced [26, 27]. Some recent developments that have received a great amount of attention are based on the theory of random finite set (RFS) [28]. These RFS based methods include probability hypothesis density filtering (PHDF) and multi-Bernoulli filtering (MB) [29, 30]. These models are used to model and track object states. In an RFS setup, most methods pair objects to their associated estimated state parameters using clustering methods after tracking [31]. In recent studies on RFS theory and its application on multi-object tracking, some new methods such as the labeled multi-Bernoulli filtering, generalized multi-Bernoulli filtering are introduced in which the labeled RFS is exploited to estimate the object tracks and update the trajectory [32].

Despite success in MOT algorithms through RFS methods, their use is more suited for the small number of objects; these methods are computationally expensive and do not perform in high noise conditions. These methods are often too slow and cannot robustly and efficiently estimate the trajectories simultaneously.

Bayesian nonparametrics is the area of Bayesian statistics in which the finite-dimensional parametric prior distributions of classical Bayesian statistics are replaced with stochastic processes. In practice, however, two stochastic processes—the Gaussian process and the Dirichlet process— are the most used processes in this context due to their flexibility. Bayesian nonparametric methods have recently become very popular in various research areas. Advances in computing the posterior distributions have turned this area of Bayesian statistics to a feasible and reliable field of study; Markov chain Monte Carlo (MCMC), and variational Bayes (VB) sampling methods are amongst the popular sampling approaches that facilitate computation of the posterior distribution.

Bayesian nonparametric models have recently been introduced to the problem of multi-object tracking [33]. For example, a hierarchical Dirichlet process on the modes is employed to provide a prior over the unknown number of unobserved modes for tracking with maneuvering [34, 35] and a generalized Pólya scheme is employed to track multiple objects [36, 37].

## 1.2 Contributions

In this work, we mainly focus on constructing robust Bayesian nonparametric priors with some desired properties over multiple object tracking. We develop efficient Bayesian inference methods to sample from posterior distributions. To this end, we propose several approaches to improve both the prediction and update performance of multi-object tracking. In the first approach, we primarily concentrate on the construction of dependent prior models over objects for which the marginal distributions have well known nonparametric distributions. The second approach constructs an inexpensive nonparametric method based on an infinite random tree and diffusion processes. The third approach, informations of multiple sensors is exploited through a hierarchical nonparametric modeling over the dependent measurements received from multiple sensors to track a single object. Lastly, we propose a Bayesian nonparametric modeling for multiple object tracking with multiple dependent measurements. We integrate the proposed dependent Dirichlet process prior over the object states with the hierarchical Dirichlet process prior over the dependent measurements to successfully associate each measurement to the corresponding object and to more accurately estimate the cardinality of objects at each time step.

### *1.2.1 Dependent Bayeisan Nonparametric Modeling and Identity Learning for Multiple Object Tracking*

Our contributions mainly encompass tracking multiple objects with unknown, time-dependent cardinality and identity using measurements received from multiple sensors.

We propose a class of time-dependent distributions for multi-object tracking problem that exploits a dependent Dirichlet process as the prior on the object state parameters to infer the trajectory of each object. We propose MCMC methods to do inference. The problem of multi-object tracking becomes even more challenging when the unordered measurements have a large number of false alarms due to high noise [13, 24]. In general, we aim to accurately and robustly estimate the trajectory of each object and learn the cardinality of time-varying objects at any time step. There are various practical examples: dependent Dirichlet process to model the time-dependent targets in a radar tracking problem, locating specific cognitive and behavioral information in different regions in the brain by tracking multiple neural dipole sources using patient-dependent electroencephalography (EEG) recordings which include interference from physiologic and extra-physiologic artifacts. We simulate a multi-object tracking problem to exhibit the advantages of Bayesian nonparametric models to infer and estimate the tracks.

We also construct another class of time-dependent distributions that can be used to tack multiple objects. The family of dependent Pitman-Yor (DPY) process is proposed to model the state prior in multiple object tracking. This process is shown to be more flexible and a better match than the dependent Dirichlet process in tracking a time-varying number of objects. The DPY model directly incorporates learning multiple parameters from correlated information. This prior not only obtains the full dependency amongst the objects but may also be integrated with a Dirichlet process mixture model to accurately estimate time-dependent object cardinality, to provide object labeling, and to identify object-to-measurement association. We provide an MCMC sampling method to do inference and track the trajectory of each object. Simulations are used to demonstrate that the proposed nonparametric model effectively traces the objects and extends to learning the object cardinality based on the received measurements.

### *1.2.2 Random Infinite Tree and Dependent Poisson Diffusion Process for Multiple Object Tracking*

Tracking a time-varying number of objects using unordered sets of measurements can be a challenging and computationally intensive problem; most methods require the pairing of objects to their associated estimated state parameters after tracking. However, the main challenge is how to robustly associate objects on a new scene with previously estimated objects. We propose a new approach that links random graph theory, Bayesian nonparametrics, and multi-object tracking to track multiple objects at each time step using previously tracked objects. This model utilizes diffusion processes to construct an evolutionary process. This method efficiently estimates the object trajectory along with object identification at each step by tracing the paths on a random tree. This method is not only robust but also inexpensive since it directly takes advantage of information learned at the previous time step to evolve objects. Searching over random trees produces the trajectory of each object at each time step. We also study the performance of the proposed method. Empirical results on a dataset containing five objects demonstrate the benefits of this graph-based model, and thus the advantages of inference algorithms derived from nonparametric models.

### *1.2.3 Bayesian Nonparametrics for Dependent Measurements*

We investigate a single object tracking using multiple dependent measurements provided from multiple dependent sensors. We consider a multimodal dependent framework for integration of complementary information in analyzing a scene. We develop a method based on the hierarchical Dirichlet process to group the dependent measurements such that the sensor information and the dependency among the measurements are preserved. The Hierarchical Dirichlet process to group the dependent measurements improves the performance of the tracker. This method clusters measurements that are collected by each sensor and estimates joint density of dependent measurements. We show through simulations that assuming dependency among the sensor measurement may improve the tracker in the sense that mean square error (MSE) of the tracker is much smaller than that of with no dependency assumption.

### *1.2.4 Bayesian Nonparametrics for Multiple Dependent Measurements and Multiple Object Tracking*

We extend the multi-object tracking to include statistically dependent measurements from multiple sensors by proposing a dependent Dirichlet process prior over the object state parameters and a hierarchical model to take advantage of the additional information provided by multimodal dependent measurements to improve tracking performance. This model fully captures the dependency among objects and measurements and can robustly associate each measurement to the corresponding object and accurately infer the trajectory of objects by exploit dependency among measurements. We demonstrate through simulations that taking the dependency among the measurements and information provided by multiple dependent sensors into account may improve the tracking procedure. Simulations also show that assuming dependent measurements may improve the object cardinality at each time step.

## 1.3 Organization

This dissertation is organized as follows. Chapter 2 surveys a broad range of Bayesian nonparametric and inferential methods upon which models in the thesis are constructed. In Chapter 3, a class of dependent nonparametric models is proposed for which the marginal distribution follows a Dirichlet process. Gibbs sampler for this model to sample from the posterior is also provided. We discuss the consistency and contraction rate of this nonparametric process. Chapter 4 generalizes the model introduced in Chapter 3 to a family of dependent processes where the marginal distribution is a two-parameter Poisson-Dirichlet process (Pitman-Yor process). This model benefits from the power law property of the Pitman-Yor process, and therefore it is more suited for the multiple object tracking. In Chapter 5, we propose a new approach to introduce a class of dependent processes over random trees. This model accurately and efficiently estimates the trajectory by tracing the paths on the infinite random trees. In Chapter 6, we investigate the multi-object tracking problem when multiple sensors provide dependent measurements. We utilize the information obtained through the dependency of measurements to accurately and robustly track each object. In Chapter 7, we conclude by summarizing the contributions of this work and outline directions for future research. The acronyms and notation used throughout the dissertation are summarized in the following tables.

## 1.4 List of Symbols

### General Notation

| Symbols | Definition |
|---|---|
| $\|\| \cdot \|\|$ , $\|\| \cdot \|\|_2$ | $L_2$ Distance |
| $\|\| \cdot \|\|_{\mathrm{TV}}$ | Total Variation Distance |
| $\mathrm{KL}(p, q)$ | Kullback-Leibler Distance between Densities $p$ and $q$ |
| $\mathrm{KL}(\Pi)$ | Kullback-Leibler Support of Prior $\Pi$ |
| $d_H(p, q)$ | Hellinger Distance between Densities $p$ and $q$ |
| $\mathcal{H}_\kappa$ | $\kappa$-smoothed Holder Space |
| $\delta_\theta(A)$ | Indicator Function |
| $\mathbb{1}_\theta(A)$ | Indicator Function |
| $x^n$ | Collection of Random Variables $\{x_1, \ldots, x_n\}$ |
| $\mathcal{X}^n$ | Sample Space of n-dimensional Vector |
| $\mathbb{R}^n$ | Vector Space of Real-valued n-dimensional Vector |
| $I$ | Identity Matrix |
| $p(x)$ | Probability Density Function (p.d.f) of random variable $x$ |
| $p(x\|y)$ | Conditional Probability Density Function of Random Variable of $x$ Given Random Variable $y$ |
| $P_x, \mathbb{P}_x$ | Distribution of $x$ whose density is $p(x)$ |
| $P_{x\|z}$ | Conditional Distribution of Random Variables $x$ given $z$ |
| $\mu \ll \nu$ | $\mu$ Absolutely Continuous with respect to $\nu$ |
| i.i.d. | Independently and Identically Distributed |
| $x_1, x_2 \cdots \overset{\text{i.i.d.}}{\sim} P_x$ | Random Variables $x_1, x_2 \ldots$ Drawn i.i.d. from Distribution $P_x$ |
| $\mathbb{E}_\theta[\cdot]$ | Expected Value for Fixed Parameter $\theta$ |
| $\mathrm{Card}(A)$, $\#A$ | Cardinality of A |

| **Symbols** | **Definition** |
|---|---|
| $s_k$ | $k$-dimensional Unit Simplex |
| $s_\infty$ | Infinite-dimensional Unit Simplex |
| $o_P(a_n)$ | Sequence of Random Variables $a_n$ Approaching Zero in Probability $P$ |
| $O_P(a_n)$ | Sequence of Random Variables $a_n$ Bounded in Probability $P$ |
| $H(X), H(p)$ | Shannon Entropy of Random Variable $X \sim p$ |
| $\mathrm{D}(\epsilon, \Theta, d)$ | $\epsilon$-packing Number of $\Theta$ with respect to Distance $d$ |
| $\mathrm{N}(\epsilon, \Theta, d)$ | $\epsilon$-covering Number of $\Theta$ with respect to Distance $d$ |
| $\mathrm{N}_{[]}(\epsilon, \Theta, d)$ | $\epsilon$-bracketing Number of $\Theta$ with respect to Distance $d$ |
| $\log \mathrm{N}_{[]}(\epsilon, \Theta, d)$ | Entropy |
| $\mathcal{U}nif([a, b])$ | Uniform Distribution on $[a, b]$ |
| $\mathcal{N}(\mu, \Sigma)$ | Normal Distribution with Mean $\mu$ and Covariance Matrix $\Sigma$ |
| $\mathcal{NIW}(\mu_0, \lambda, \nu, \Psi)$ | Normal-inverse-Wishart Distribution with parameters $\mu_0 \in \mathbb{R}^N, \lambda \in \mathbb{R}^+, \nu \in \mathbb{R}$, and $\Psi \in \mathbb{R}^{N \times N}$ |
| $\mathrm{Po}(\lambda)$ | Poisson Distribution with Mean $\lambda$ |
| $\Gamma(a, b)$ | Gamma Distribution with Shape Parameter $a$ and Rate $b$ |
| $\mathrm{Beta}(a, b)$ | Beta Distribution with Parameters $a, b > 0$ |
| $\mathrm{Mult}(n; \pi_1, \ldots, \pi_K)$ | Multinomial Distribution with Parameters $n$, $\pi_k > 0$ and $\sum_{k=1}^{K} \pi_k = 1$ |
| $\mathrm{Dir}(\alpha_1, \ldots, \alpha_K)$ | Dirichlet Distribution with Parameters $\alpha_k > 0$ and $\sum_{k=1}^{K} \alpha_k = 1$ |
| $\mathrm{DP}(\alpha, H)$ | Dirichlet Process with Hyperparameter $\alpha$ and Base Distribution $H$ |
| $\mathcal{PY}(d, \alpha, H)$ | Pitman-Yor Process with Hyperparameters: Discount Parameter $d$, Concentration Parameter $\alpha$, and Base Distribution $H$ |

### Multiple Object Tracking

| **Symbols** | **Definition** |
|---|---|
| $N_k$ | Number of Objects at Time Step $k$ |
| $M_k$ | Number of Measurements at Time Step $k$ |
| $\mathbf{x}_{\ell,k}$ | $\ell$th Object State Vector at Time Step $k$ |
| $\mathbf{X}_k$ | Set of All Objects at Time Step $k$ |
| $\mathbf{X}_k^{-i}$ | $\mathbf{X}_k \setminus \{\mathbf{x}_{i,k}\}$ |
| $\mathbf{X}_{1,k}^{\ell}$ | Collection of $\{\mathbf{X}_{1,k}, \ldots, X_{\ell,k}\}$ |
| $\mathbb{Q}_{\boldsymbol{\theta}}(\cdot,\cdot)$ | Probability Transition Kernel given parameters $\theta$ |
| $P_{k\|k-1}$ | Probability of Remaining in the Scene from time $(k-1)$ to $k$ |
| $\mathbf{z}_{l,k}$ | $l$th Measurement Vector at Time Step $k$ |
| $\mathbf{Z}_k$ | Set of all Measurements at Time Step $k$ |
| $\mathscr{Z}$ | Measurements Space |
| $\mathbf{Z}_k^i$ | Set of all Measurements Received from $i$th Sensor at Time Step $k$ |
| $\boldsymbol{\theta}_{\ell,k}$ | $\ell$th Object Parameter at Time Step $k$ |
| $\Theta_k$ | Set of All Parameters at Time Step $k$ |
| $\Theta_k^{\star}$ | Set of All Unique Parameters at Time Step $k$ |
| $\nu(\cdot,\cdot)$, $\xi(\cdot,\cdot)$ | Transition Kernels |
| $\mathcal{C}_k$ | Cluster Assignment up to Time Step $k$ |

# Chapter 2

# BAYESIAN NONPARAMETRIC AND INFERENCE MODELS

This chapter outlines the background necessary for subsequent developments in this thesis. Bayesian nonparametric models provide a flexible statistical model selection method as well as a method to choose a model at an appropriate level of complexity for a variety of problems in statistics, computer science, and electrical engineering. The primary focus of this thesis is on problems that arise in multi-object tracking and how to address them through Bayesian nonparametric models. In this chapter, we briefly discuss two main nonparametric models and discuss their basic properties. In Section 2.1, we provide a comprehensive analysis of distributions that play an integral role in Bayesian statistics; highlighting the importance of conjugate priors in Bayesian analysis. In Section 2.2, we describe the significance of Bayesian nonparametrics. In the subsequent sections, we study the main nonparametric models from which we construct our novel models in this thesis. Section 2.3 discusses the Dirichlet process and its properties and the generalized Dirichlet process. The two-parameter Poisson-Dirichlet process (Pitman-Yor Process) is studied in detail in Section 2.4. Bayesian inferential methods should be adapted to be able to make inference in the nonparametric models. The invention of Markov chain Monte Carlo (MCMC) methods enables us to do inference in high-dimensional datasets. In Section 2.5, we discuss core inferential methods; Monte Carlo methods and variational Bayes methods to achieve flexible and robust inferential methods in infinite-dimensional spaces. We propose novel inferential models over infinite-dimensional spaces that are mainly based on these two methods. These models are adapted to provide a tractable analysis of Bayesian nonparametric models.

## 2.1 Analysis of Distributions

### 2.1.1 Exponential Family

In this section, we introduce a class of parametric distributions; this family of distributions include the Gaussian, multinomial, Poisson, Beta and many other distributions. For a random variable $x \in \mathcal{X}$, an exponential family of distributions are distributions whose densities (given $\theta$) follow

$$p(x|\theta) = h(x) \exp\{\theta^T T(x) - A(\theta)\} \tag{2.1}$$

where the parameter vector $\theta$ is often called the family's natural or canonical parameters, $h(x)$ is a nonnegative reference measure. $T(x)$ is the sufficient statistics for the exponential family. The cumulant function $A(\theta)$ is a logarithm of a normalizer and defined as

$$A(\theta) = \log \int h(x) \exp\{\theta^T T(x)\} \nu(dx) \tag{2.2}$$

for a deterministic measure $\nu(\cdot)$. The exponential family is well defined if the integral in Equation (2.2) is finite. The set of canonical parameters for which Equation (2.2) is finite defines the natural parameter space and mathematically formulated as

$$\mathcal{C} = \{\theta : A(\theta) < \infty\}. \tag{2.3}$$

We restrict our definition to the exponential family that is regular, meaning $\mathcal{C}$ is a nonempty open set. As a case in point, the Gaussian, Poisson, Beta, and gamma distributions fall into this category. It is straightforward to see that the convexity of $A(\theta)$ in $\theta$ results in the convexity of $\mathcal{C}$, and if the family is minimal, then $A(\theta)$ is strictly convex [38]. There is a close relationship between the derivatives of the cumulant function and the moments of sufficient statistics [38–40]; it can be easily shown that

$$\begin{aligned} \frac{\partial A}{\partial \theta^T} &= \mathbb{E}[T(x)] \\ \frac{\partial^2 A}{\partial \theta \partial \theta^T} &= \mathrm{Var}[T(x)]. \end{aligned} \tag{2.4}$$

where $\mathbb{E}[\cdot]$ and $\mathrm{Var}[\cdot]$ denote statistical expectation and variance, respectively.

### Maximum Likelihood Estimator

In this section, we study the maximum likelihood (MLE) estimator of $\mu := \mathbb{E}[T(x)]$ as a function of the canonical parameter $\theta$. Assuming $x_1, \ldots, x_N \sim p(x|\theta)$[1] and using Equation (2.1), the log-likelihood is

$$\ell(\theta) = \log\Big(\prod_{j=1}^{N} h(x_j)\Big) + \theta^T\Big(\sum_{j-1}^{N} T(x_j)\Big) - NA(\theta). \tag{2.5}$$

Taking the partial derivative of the Equation (2.5) with respect to $\theta$ and setting the result to zero yields the unbiased maximum likelihood estimator of $\theta$ as

$$\hat{\theta}_{\mathrm{MLE}} = \frac{1}{N}\sum_{j=1}^{N} T(x_j). \tag{2.6}$$

It is shown that $\hat{\theta}_{\mathrm{MLE}}$ is an unbiased estimator and attains the Cramér-Rao lower bound, i.e., the Fisher information is

$$\ell(\theta) = \frac{1}{\mathrm{Var}[T(x)]}.$$

assuming that samples $x_1, \ldots, x_N \sim \tilde{p}$ are drawn independently and identically (i.i.d.) distributed, empirical density $p^*(x)$ is

$$p^*(x) = \frac{1}{N}\sum_{j=1}^{N} \delta_{x_j}(x). \tag{2.7}$$

[1] Note that $x \sim P$ or equivalently $x \sim p$ indicates that random variable $x$ is drawn from a distribution $P$ whose density is $p$. Notation $x|P \sim P$ displays the conditioning on a distribution.

where $\delta_{x_j}(x) = \delta(x - x_j)$ is the delta function, defined to be 1 if $x = x_j$ and zero if $x \neq x_j$.

There is a close correspondence between the maximizing the likelihood and minimizing the Kullback-Leibler (KL) distance. This correspondence often provides an alternative approach to optimize the likelihood function. In particular, the KL distance between densities $p^*$ and $p_\theta$ is given by

$$\begin{aligned} \mathrm{KL}(p^*, p_\theta) &= \sum_x p^*(x) \log \frac{p^*(x)}{p(x|\theta)} \\ &= \sum_x p^*(x) \log p^*(x) - \sum_x p^*(x) \log p(x|\theta) \\ &= -H(p^*) - \sum_x \frac{1}{N} \sum_{j=1}^{N} \delta_{x_j}(x) \log p(x|\theta) \qquad (2.8) \\ &= -H(p^*) - \frac{1}{N} \sum_{j=1}^{N} \log p(x_j|\theta) \\ &= -H(p^*) - \frac{1}{N} \ell(\theta) \end{aligned}$$

where $H(p^*)$ is the entropy of $X$ with respect to the empirical density $p^*$ and is not a function of $\theta$. Moreover, Equation (2.8) shows that

$$\hat{\theta}_{\mathrm{MLE}} = \arg\max_\theta \ell(\theta) = \arg\min_\theta KL(p^*, p_\theta). \qquad (2.9)$$

### Bayesian Inference

So far, we treated the parameters fixed but unknown. In this section, we develop a Bayesian inference method by treating the parameters as random. A comprehensive, detailed version of this topic can be found in [40, 41].

Assume $x_1, \ldots x_N \sim p(x|\theta)$ where $p(x|\theta)$ is the canonical exponential family with parameter $\theta$. We assume a prior $p(\theta|\gamma)$ on the parameter $\theta$ with hyperparameters $\gamma$.

By Bayes' rule, the posterior distribution equals to

$$p(\theta|\{x_j\}_{j=1}^N, \gamma) \propto p(\theta|\gamma) \prod_{j=1}^N p(x_j|\theta). \tag{2.10}$$

Typically in Bayesian statistics, a prior $p(\gamma)$ is placed over the hyperparameter $\gamma$. In practice, however, $\gamma$ is often estimated using a frequentist method. In particular,

$$\hat{\gamma} = \arg\max_{\gamma} p(x_1, \ldots, x_N|\gamma). \tag{2.11}$$

However, this optimization problem is not tractable. The solution to find the optimal $\gamma$ is often computed via leave-one-out cross validation.

In many applications, statistical models are particularly used to predict new observations. For a new observation $x_{\text{new}}$, the predictive distribution is given by

$$p(x_{\text{new}}|\{x_j\}_{j=1}^N, \gamma) = \int_{\mathcal{C}} p(x_{\text{new}}|\theta) p(\theta|\{x_j\}_{j=1}^N, \gamma) d\theta. \tag{2.12}$$

Often, Equation (2.12) is intractable, in which case we approximate the parameters using the maximum a posteriori (MAP) estimator,

$$\hat{\theta}_{\text{MAP}} = \arg\max_{\theta} p(\theta|\{x_j\}_{j=1}^N, \gamma). \tag{2.13}$$

We study cases for which Equation (2.12) is tractable. This class of distributions are called conjugate prior.

**Definition:** A family of distribution, $\mathcal{F}$, is called *conjugate prior* for likelihood $p(x|\theta)$ if for every prior $p(\theta) \in \mathcal{F}$, the posterior $p(\theta|x) \in \mathcal{F}$.

For the exponential family with density as in Equation (2.1), the likelihood for independent samples $x_1, \ldots, x_N$ is given by

$$p(x_1, \ldots, x_N|\theta) = \big(\prod_{j=1}^N h(x_j)\big) \exp\Big(\theta^T \big(\sum_{j=1}^N T(x_j)\big) - NA(\theta)\Big). \tag{2.14}$$

**Propositions 1.** A conjugate family for the likelihood $p(x_1, \ldots, x_N|\theta)$ is

$$p(\theta|\tau, \zeta) = Z(\tau, \zeta) \exp\Big(\tau^T\theta - \zeta A(\theta)\Big) \tag{2.15}$$

where $Z(\tau, \zeta)$ is the normalizer. Then, the posterior distribution is

$$p\Big(\theta|\tau + \sum_{j=1}^{N} T(x_j), \zeta + N\Big). \tag{2.16}$$

Given Proposition 1, we can compute the predictive likelihood as follows:

$$p(x_{\text{new}}|\{x_j\}_{j=1}^N, \gamma) = \frac{Z(\tau + \sum_{j=1}^{N} T(x_j), \zeta + N)}{Z(\tau + \sum_{j=1}^{N} T(x_j) + T(x_{\text{new}}), \zeta + N + 1)}. \tag{2.17}$$

(see [40, 42] for a detailed proof). It can be shown that the posterior expectation of $\mu = \mathbb{E}[T(x)]$ is a convex combination of the prior expectation and the maximum likelihood estimate.

### *2.1.2 Multinomial Distribution and Dirichlet Distribution*

#### Multinomial Distribution

Consider a random variable $x$ taking $K$ possible categorical outcomes, i.e., the outcome space is $\mathcal{X} = \{1, 2, \ldots, K\}$. Suppose each category is selected with probability $\pi_k = \mathbb{P}(x = k)$. The distribution that characterizes random variable $x$ given $\pi_k$, $k = 1, \ldots, K$ has the following probability mass function

$$p(x|\pi_1, \ldots, \pi_K) = \prod_{k=1}^{K} \pi_k^{\mathbb{1}_x(k)}, \qquad \mathbb{1}_x(k) = \begin{cases} 1 & \text{if} \quad x = k \\ 0 & \text{otherwise} \end{cases}. \tag{2.18}$$

Let $x_k$ be the random variable which counts the number of observations selecting category $k$. Define random vector $\mathbf{x} = (x_1, \ldots, x_K)$ to be the vector of counts such that $\sum\limits_{k=1}^{K} x_k = n$. Random vector $\mathbf{x}$ has a multinomial distribution if its probability mass function follows [40, 43]

$$p(\mathbf{x}) = p(x_1, \dots, x_K) = \Big(\frac{n!}{\prod_k x_k!} \prod_{k=1}^{K} \pi_k^{x_k}\Big) \mathbb{1}_{[\sum_k x_k]}(n). \tag{2.19}$$

For $K = 2$ is simplifies to the binomial distribution. One can observe that the parameters of a multinomial distribution lie in a $K-1$ dimensional simplex

$$\Pi_{K-1} = \{\pi \in \mathbb{R}^K : 0 \leq \pi_k \leq 1, \sum_k \pi_k = 1\}. \tag{2.20}$$

It is simple to show that if $\mathbf{x} \sim$Mult$(n; \pi_1, \dots, \pi_K)$, then

$$\begin{aligned} \mathbb{E}[x_k] &= n\pi_k \\ \mathrm{Var}(x_k) &= n\pi_k(1-\pi_k). \end{aligned} \tag{2.21}$$

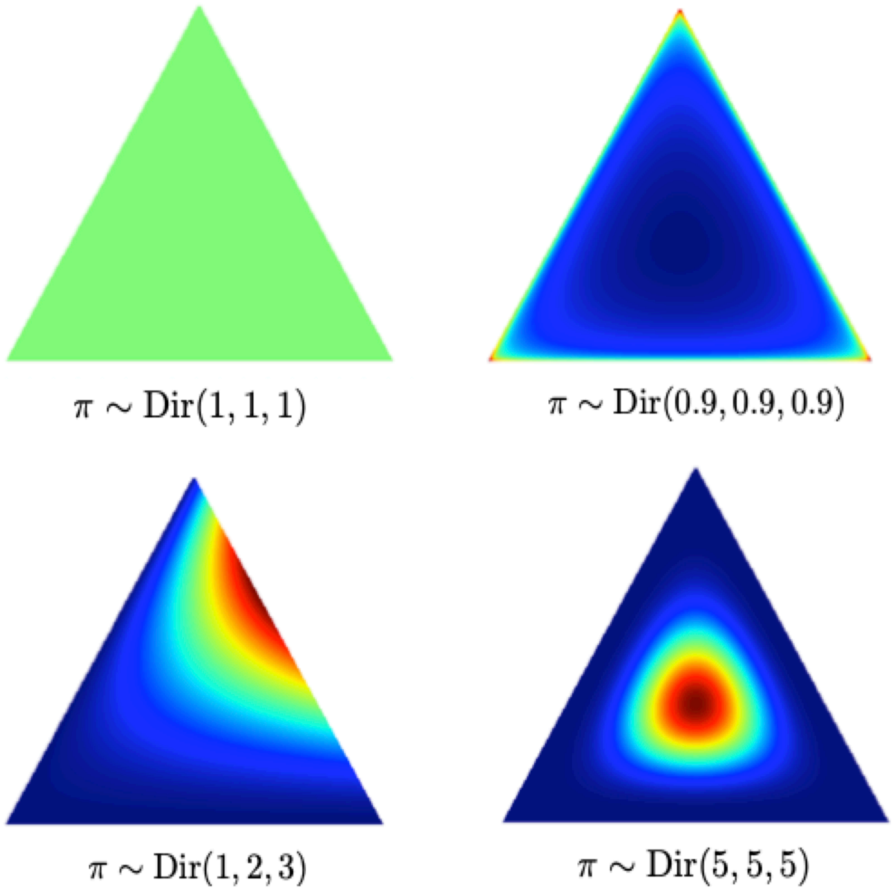
The multinomial distribution defines a regular exponential family since it may be re-written as

$$\begin{aligned} p(\mathbf{x}) &= \Big(\frac{n!}{\prod_k x_k!} \exp\Big\{\sum_{k=1}^{K} x_k \log \pi_k\Big\}\Big) \mathbb{1}_{[\sum_k x_k]}(n) \\ &= \Big(\frac{n!}{\prod_k x_k!} \exp\Big\{\sum_{k=1}^{K-1} x_k \log \pi_k + \big(1 - \sum_{k=1}^{K-1} x_k\big) \log\big(1 - \sum_{k=1}^{K-1} \pi_k\big)\Big\}\Big) \mathbb{1}_{[\sum_k x_k]}(n) \\ &= \Big(\frac{n!}{\prod_k x_k!} \exp\Big\{\sum_{k=1}^{K-1} \log\Big(\frac{\pi_k}{\Pi_k}\Big) x_k + \log\big(1 - \sum_{k=1}^{K-1} \pi_k\big)\Big\}\Big) \mathbb{1}_{[\sum_k x_k]}(n) \end{aligned} \tag{2.22}$$

where $\Pi_K = 1 - \sum_{k=1}^{K-1} \pi_k$ and hence, it follows the exponential family with canonical parameters $\theta_k = \log\big(\frac{\pi_k}{\Pi_K}\big)$ and cumulant $A(\theta_1, \dots, \theta_K) = -\log\big(1 - \sum_{k=1}^{K-1} \pi_k\big)$. Using $\theta_k = \log\big(\frac{\pi_k}{\Pi_K}\big)$, we can re-write the cumulant $A(\theta_1, \dots, \theta_K) = \log\big(\sum\limits_{k=1}^{K} \exp(\theta_k)\big)$. The maximum likelihood estimator of the multinomial parameters is $\hat{\pi}_k = x_k/n$.

### Dirichlet Distribution

The Dirichlet distribution [40, 42] is a class of distributions which is the conjugate prior for the multinomial distribution. The Dirichlet distribution with parameters



$\pi \sim \text{Dir}(1,1,1)$

$\pi \sim \text{Dir}(0.9,0.9,0.9)$

$\pi \sim \text{Dir}(1,2,3)$

$\pi \sim \text{Dir}(5,5,5)$

Figure 2.1: Dirichlet Distribution as Uniform Prior (Top, Left), Prior Favoring Sparse Multinomial Distribution(Top, Right), Biased Prior(Bottom, Left), and Unbiased unimodal prior (Bottom, Right).

$(\alpha_1, \ldots, \alpha_K)$ is denoted by $\text{Dir}(\alpha_1, \ldots, \alpha_K)$ and has the probability density function

$$p(\pi_1, \ldots, \pi_K | \alpha) = \frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} \pi_k^{\alpha_k - 1} \tag{2.23}$$

where $\pi = (\pi_1, \ldots, \pi_K) \in \Pi_{K-1}$. When $K = 2$, Dirichlet distribution is known as Beta distribution. We discuss Beta distribution in the next section in detail.

**Propositions 2.** If $\pi \sim \text{Dir}(\alpha_1, \ldots, \alpha_K)$, and $\alpha_0 = \sum_k \alpha_k$, then

a) $\mathbb{E}[\pi_k] = \frac{\alpha_k}{\alpha_0}$

b) $\mathrm{Var}(\pi_k) = \frac{\alpha_k(\alpha_0-\alpha_k)}{\alpha_0^2(\alpha_0+1)}$

c) $\mathrm{Cov}(\pi_j, \pi_k) = \frac{-\alpha_j\alpha_k}{\alpha_0^2(\alpha_0+1)}, \qquad j \neq k$

d) Aggregation property: The combination of a subset of categories is also Dirichlet, for example, if $\pi \sim \mathrm{Dir}(\alpha_1, \ldots, \alpha_{K-1}, \alpha_K)$, then $(\pi_1, \ldots, \pi_{K-1} + \pi_K) \sim \mathrm{Dir}(\alpha_1, \ldots, \alpha_{K-1} + \alpha_K)$

e) Marginal distribution of any individual $\pi_k$ has a Beta density, i.e.,

$$\pi_k \sim \mathrm{Beta}(\alpha_k, \alpha_0 - \alpha_k)$$

f) Multinomial and Dirichlet distributions are conjugate priors. The posterior distribution has a Dirichlet distribution. If we have $N$ observations $\{x^n\}_{n=1}^N$ from a multinomial distribution, then the posterior distribution is

$$p(\pi|\{x^k\}_{n=1}^N, \alpha) \sim \mathrm{Dir}\Big(\alpha_1 + \sum_n \mathbb{1}_{x^n}(1), \ldots, \alpha_K + \sum_n \mathbb{1}_{x^n}(K)\Big). \tag{2.24}$$

Figure 2.1 displays a Dirichlet distribution for different values of $\alpha$ and $K = 3$ on the simplex $\Pi_2 = (\pi_1, \pi_2, 1 - \pi_1 - \pi_2)$.

### *2.1.3 Beta Distribution*

The Beta distribution is a class of continuous probability distributions defined on $[0, 1]$. It is parametrized by parameters $a, b > 0$, and it is a special case of the Dirichlet distribution. A random variable $x \sim \mathrm{Beta}(a, b)$, then it has distribution of

$$P(dx|a, b) = \frac{1}{\beta(a, b)} x^{a-1}(1 - x)^{b-1} dx \tag{2.25}$$

where $\beta(a, b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$ and $\Gamma(\cdot)$ is the gamma function. The probability density function for different values of $(a, b)$ is depicted in Figure 2.2.

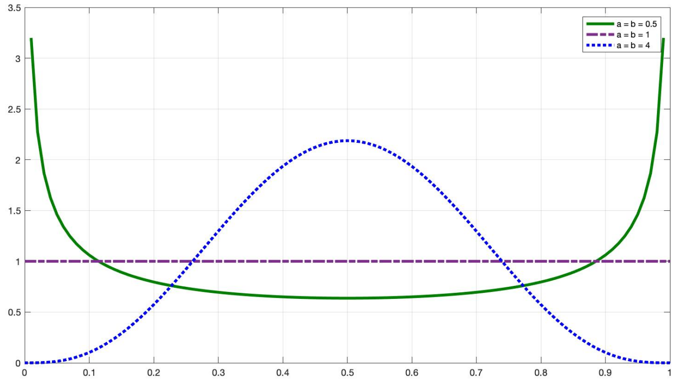


Figure 2.2: Beta Probability Density Function.

If $x \sim \text{Beta}(a, b)$, then

$$\begin{aligned}
\mathbb{E}(x) &= \frac{a}{a+b} \\
\text{Var}(x) &= \frac{ab}{(a+b)^2(a+b+1)} \\
\mathbb{E}[\ln X] &= \frac{\partial \ln \Gamma(a)}{\partial a} - \frac{\partial \ln \Gamma(a+b)}{\partial a} = \psi(a) - \psi(a+b)
\end{aligned} \tag{2.26}$$

where $\psi$ is digamma function.

The Beta distribution is the conjugate prior for the Bernoulli, binomial, negative binomial, and geometric distribution. The moment generating function of the Beta distribution is given by

$$\mathbb{E}[e^{\lambda X}] = 1 + \sum_{k=1}^{\infty} \big( \prod_{i=0}^{k-1} \frac{a+j}{a+b+j} \big) \frac{\lambda^k}{k!}. \tag{2.27}$$

### *2.1.4 Gamma Distribution*

The two-parameter family of continuous probability distributions is called Gamma distribution and denoted by $x \sim \Gamma(\alpha, \beta)$ if, for $\alpha, \beta > 0$, the density follows

$$P(dx|\alpha, \beta) = \frac{\beta^{\alpha}}{\Gamma(\alpha)} x^{\alpha-1} \exp\left(-\beta x\right) \mathbb{1}_x[0, \infty] dx. \tag{2.28}$$

where $\Gamma(\cdot)$ is the gamma function. Gamma distribution is exponential family for natural parameters $\theta = [\alpha - 1, -\beta]^T$. If $x \sim \Gamma(\alpha, \beta)$, then

$$\begin{aligned} \mathbb{E}[x] &= \frac{\alpha}{\beta} \\ \mathrm{Var}[x] &= \frac{\alpha}{\beta^2}. \end{aligned} \tag{2.29}$$

Suppose $x_j \sim \Gamma(\alpha_j, \beta)$ for $j = 1, 2, \ldots, N$, then

$$\sum_{j=1}^{N} x_j \sim \Gamma(\sum_{j=1}^{N} \alpha_j, \beta).$$

**Propositions 3.** Assume $x \sim \Gamma(\alpha, \beta)$, then $y = \frac{1}{x}$ is distributed as the inverse-Gamma distribution whose density follows

$$P(dy|\alpha, \beta) = \frac{\beta^{\alpha})}{\Gamma(\alpha)} y^{-\alpha-1} \exp{(\frac{-\beta}{y})} \mathbb{1}_y[0, \infty] dy \tag{2.30}$$

and denoted by $\mathrm{IG}(\alpha, \beta)$. The mean and variance of $y \sim \mathrm{IG}(\alpha, \beta)$ is

$$\begin{aligned} \mathbb{E}[y] =& \frac{\beta}{\alpha - 1}, \qquad \alpha > 1 \\ \mathrm{Var}(y) =& \frac{\beta^2}{(\alpha-1)^2(\alpha-2)}, \qquad \alpha > 2. \end{aligned} \tag{2.31}$$

### *2.1.5 Student's t-Distribution*

There are two ways to derive the Student's t-distribution; first, as conjugate prior for the variance of the Gaussian distribution; second, square root of a Gamma random variable [44]. Let $x \sim \mathcal{N}(\mu, \sigma^2)$ and assume that $\mu$ is fixed and known. Assuming a $\Gamma(\alpha, \beta)$ prior over precision parameter $\tau = 1/\sigma^2$ results in the marginal density that has Student's t-distribution. The Student's t-distribution follows

$$P(dx|\mu, \alpha, \beta) = \frac{\Gamma(\alpha + 1/2)}{\Gamma(\alpha)(2\pi\beta)^{1/2}} \frac{1}{(1 + \frac{1}{2\beta}(x-\mu)^2)^{\alpha+1/2}} dx \tag{2.32}$$

where $\Gamma(\cdot)$ is gamma function. It is easy to see if $\alpha = 2$, then Student's t-distribution is the Cauchy distribution and if $\alpha \to \infty$, then the limiting distribution is a Gaussian distribution. It is common to define $\nu = \alpha/2$ and $\lambda = \alpha/\beta$ and re-write the density. A detailed discussion on the second method is discussed in detail in [39, 40, 44].

### *2.1.6 Normal-Inverse-Wishart Distribution*

A d-dimensional random variable $x$ taking values in $\mathcal{X} = \mathbb{R}^d$ has a Gaussian distribution [40, 44] with mean $\mu$ and covariance matrix $\Sigma$ if the distribution follows

$$P(dx|\mu, \Sigma) = \frac{1}{2\pi)^{d/2}|\Sigma|^{1/2}} \exp\Big\{ -\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\Big\}dx. \tag{2.33}$$

and is denoted by $\mathcal{N}(\mu, \Sigma)$. The maximum likelihood estimator for the parameters upon receiving $N$ observations $\{x^n\}_{n=1}^N$ are

$$\begin{aligned} \hat{\mu} =& \frac{1}{N}\sum_{n=1}^{N} x^n \\ \hat{\Sigma} =& \frac{1}{N}\sum_{n=1}^{N}(x^n - \hat{\mu})(x^n - \hat{\mu})^T. \end{aligned} \tag{2.34}$$

It is easy to confirm that the Gaussian distribution is in the class of exponential families with canonical parameters $\theta = (\Sigma^{-1}\mu, \Sigma^{-1})$ and sufficient statistics $T(x) = [\hat{\mu}, \hat{\Sigma}]$.

As suggested earlier, assuming conjugate priors leads to a tracktable posterior and facilitates inference. The conjugate prior for the covariance matrix of a Gaussian distribution with known mean has inverse-Wishart distribution [40, 45]. The inverse-Wishart distribution is the multivariate generalization of the inverse-Gamma distribution studied in Section 2.1.4. A d-dimensional inverse-Wishart distribution with parameters $\nu, \mathbf{\Psi}$ is denoted by $\mathcal{IW}(\nu, \mathbf{\Psi})$ and equals

$$P(d\Sigma|\nu, \mathbf{\Psi}) = \frac{\mathbb{S}^{-\nu/2}}{2^{\nu d/2\Gamma_d(\nu/2)}}|\Sigma|^{(\nu+d+1)/2} \exp\big\{ -\frac{1}{2}tr(\mathbf{\Psi}\Sigma^{-1})\big\} \tag{2.35}$$

The mean and mode for $x \sim \mathcal{IW}(\nu, \mathbf{\Psi})$ are respectively

$$\begin{aligned} \mathbb{E}[x] =& \frac{\mathbf{\Psi}}{\nu - d - 1}, \qquad \nu > d + 1 \\ \arg\max_{\Sigma} \mathcal{IW}(\Sigma; \nu, \mathbf{\Psi}) =& \frac{\mathbf{\Psi}}{\nu + d + 1}. \end{aligned} \tag{2.36}$$

If both mean and covariance matrix are unknown, the Normal-inverse-Wishart distribution provides the conjugate prior. To this end, we first draw a covariance matrix from an inverse-Wishart prior, $\Sigma \sim \mathcal{IW}(\Sigma; \nu, \mathbf{\Psi})$. Conditioning upon the covariance matrix $\Sigma$ and a scale hyperparameter $\lambda$, we then draw the mean from a normal distribution, i.e., $\mu|\mu_0, \Sigma, \lambda \sim \mathcal{N}(\mu; \mu_0, \Sigma/\lambda)$, where $\mu_0$ is the expected mean. Note that $\lambda$ may be interpreted as the pseudo observations to scale the observations. We denote the joint distribution by $\mathcal{NIW}(\mu_0, \lambda, \nu, \mathbf{\Psi})$ which equals

$$P(d\mu, d\Sigma|\mu_0, \lambda, \nu, \mathbf{\Psi}) = \mathcal{N}\big(\mu; \mu_0, \frac{\Sigma}{\lambda}\big) \times \mathcal{IW}(\Sigma; \nu, \mathbf{\Psi}) d\mu d\Sigma. \tag{2.37}$$

### Posterior Distribution

Suppose $N$ observations $\{x^n\}_{n=1}^N$ are drawn from a Gaussian distribution $\mathcal{N}(\mu, \Sigma)$. Assume a normal-inverse-Wishart distribution $\mathcal{NIW}(\mu_0, \lambda, \nu, \mathbf{\Psi})$ as a prior on $\mu, \Sigma$. The posterior distribution is also a normal-inverse-Wishart distribution with updated hyperparameters, $\mathcal{NIW}(\hat{\mu}, \hat{\lambda}, \hat{\nu}, \hat{\mathbf{\Psi}})$ [43, 45]. These hyperparameters can be computed as

$$\begin{aligned} \hat{\mu} =& \frac{\lambda \mu_0 + \sum\limits_{n=1}^{N} x^n}{\lambda + N} \\ \hat{\lambda} =& \lambda + N \\ \hat{\nu} =& \nu + N \\ \hat{\mathbf{\Psi}} =& \mathbf{\Psi} + \mathbb{S} + \frac{\lambda N}{\lambda + N} (\bar{x} - \mu_0)(\bar{x} - \mu_0)^T \end{aligned} \tag{2.38}$$

where $\bar{x} = \sum_n x^n$ and $\mathbb{S} = \sum_n (x^n - \bar{x})(x^n - \bar{x})^T$ are the sample mean and covariance matrix, respectively.

### Predictive Distribution

Marginalizing over the parameters of normal-inverse-Wishart, the predictive distribution of a new observation $x_{\text{new}}$ has a multivariate Student's t-distribution with $(\bar{\nu} - d + 1)$ degrees of freedom [43, 45]. Suppose that the normal-inverse-Wishart is proper, $\bar{nu} > d + 1$, the posterior density has finite covariance and is approximated by

$$p(x_{\text{new}}|\{x^n\}_{n=1}^N, \mu_0, \lambda, \nu, \mathbf{\Psi}) \approx \mathcal{N}(x_{\text{new}}; \hat{\nu}, \frac{(\hat{\lambda}+1)\hat{\nu}}{\hat{\lambda}(\hat{\nu}-d-1)}\hat{\mathbf{\Psi}}). \tag{2.39}$$

This approximation is the moment-matched Gaussian approximation of the posterior distribution [42, 43].

## 2.2 Introduction to Bayesian Nonparametrics

In traditional Bayesian statistics, upon receiving data $x$, with likelihood $\mathcal{L}(x|\theta)$, the Bayes formula assumes a prior $\pi(\theta)$ over the parameters and computes a posterior distribution. Therefore, a Bayesian model consists of a prior $\pi(\theta)$ on the parameters, and the likelihood $\mathcal{L}(x|\theta)$ as a function of parameters. The data is assumed to be generated in the following manner:

$$\begin{aligned} \theta &\sim \pi(\theta) \\ x_i &\sim \mathcal{L}(\cdot|\theta) \qquad j = 1, \ldots, n \end{aligned} \tag{2.40}$$

This model implies that the data is conditionally i.i.d. rather than i.i.d. Using Bayes' theorem, the posterior density is then computed as:

$$\pi(\theta|x) = \frac{\pi(\theta)\mathcal{L}(x|\theta)}{\int \pi(\theta')\mathcal{L}(x|\theta')d\theta'}. \tag{2.41}$$

The value of the parameter often remains uncertain given a finite number of observations, and Bayesian statistics uses the posterior distribution to express this uncertainty. However, in order to compute the posterior density in Equation (2.41), we require all densities to be well-defined with respect to a suitable measure. In particular, the space of parameters $\Theta$ is assumed to be finite-dimensional. The requirement to use Bayes formula is not often met if the dimension of space of parameters, $\Theta$, is infinite, and thus computing the posterior density using Bayes' formula is impossible [46]. As such, Bayesian nonparametric models fall into this category since their space of parameters is assumed to be infinite-dimensional.

The area of Bayesian nonparametrics has become more popular since as the number and size of the datasets grow, we can learn increasingly more complex information from data. This property makes Bayesian nonparametric modeling extremely appealing to the practitioners. Furthermore, the de Finetti's theorem for an exchangeable sequence of data provides a probabilistic justification for employing Bayesian nonparametric models.

**Defenition:** A sequence of random variables is *infinitely exchangeable* if the distribution is invariant for any finite sequence, i.e., for any $n$ and permutation $\sigma$

$$P(x_1 \in A_1, \ldots, x_n \in A_n) = P(x_{\sigma(1)} \in A_1, \ldots, x_{\sigma(n)} \in A_n) \tag{2.42}$$

**Theorem 1.** *(de Finetti's Theorem)* A sequence $x_1, x_2, \ldots$ is infinitely exchangeable if and only if for all $n$ and some distribution G

$$P(x_1 \in A_1, \ldots, x_n \in A_n) = \int_\theta \prod_{j=1}^{n} P(x_j \in A_j|\theta)G(d\theta).$$

This theorem explicitly guarantees that there is a random measure $G$ from which parameters are drawn such that, given parameters, data points are conditionally independent of one another.

In general, Bayesian nonparametrics answers the following questions:

A. How do we construct a prior on an infinite dimensional set?

B. How do we compute the posterior? How do we draw random samples from the posterior?

C. What are the properties of the posterior? Is the posterior consistent? What is the posterior rate of convergence?

In the next sections, we introduce a family probability measures over the space of probability measures. We first introduce the Dirichlet process and different methods to construct it. We introduce a probability distribution on partitions known as the Chinese restaurant process, and we show that the exchangeability property of the Chinese restaurant process leads to the Dirichlet process. We then study the two-parameter Poisson-Dirichlet distributions (also known as Pitmna-Yor process) and compare it to the Dirichlet process. It is worth mentioning that Bayesian inference methods do not necessarily coincide with that of frequentist. Also, Bayesian models do not necessarily have properties like consistency or optimal rates of convergence.

## 2.3 Dirichlet Process

To do Bayesian nonparametric inference, we need to put a prior $\pi$ on infinite dimensional space. The most popular Bayesian nonparametric model over the space of distributions is the Dirichlet process. The Dirichlet process first appeared in a paper by Ferguson [47]. A prior over an infinite dimension leaves open the question as to whether such a process actually exists. In [47], Ferguson makes use of Kolmogorov extension theorem to prove the existence of such processes. Such a construction, however, encounters a measure-theoretic difficulty that requires certain topological conditions to be placed on the space of parameters (space of distributions). Sethuraman provides a constructive definition of the Dirichlet process that removes the restrictions of the original definition [48]. Aldous later introduced a distribution over

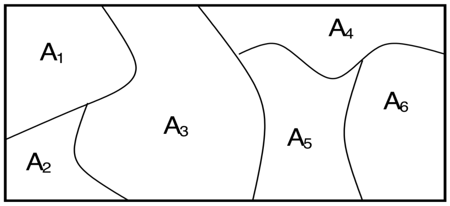


Figure 2.3: Partition of the Parameter Space.

partitions where underlying distribution is the Dirichlet process [49]. Blackwell and McQueen introduced another equivalent definition of the Dirichlet process based on Pólya urn scheme [50].

In this work, we only study the following representations:

- Ferguson definition of Dirichlet process [47]
- Stick-breaking process [48]
- Chinese restaurant process [49]
- Blackwell-MacQueen process (Pólya urn scheme) [50]

### *2.3.1 Ferguson Definition of Dirichlet process*

Ferguson presents a class of priors that have a large support for which given the data, the posteriors can be computed analytically.

**Definition:** *Dirichlet process* is a random probability measure over the space $\Theta$ satisfying:

- Let $A_1, \ldots, A_n$ be a partition of the Polish space $\Theta$ as shown in Figure 2.3. Let $G \sim DP(\alpha, H)$ be a realization of a Dirichlet process with concentration parameter $\alpha$, and base distribution $H$, then

  a) $G$ is a random measure

b) $G$ is discrete with probability one

c) The vector $(G(A_1), \ldots, G(A_n))$ is a probability vector

d) $(G(A_1), \ldots, G(A_n)) \sim \text{Dirichlet}(\alpha H(A_1), \ldots, \alpha H(A_n))$

It is clear that $G$ is a random measure therefore $G(A)$ is a random variable given an event $A$. It is straightforward from the definition to prove the following:

$$\begin{aligned} \mathbb{E}[G(A)] &= H(A) \\ \text{Var}[G(A)] &= \frac{H(A)(1 - H(A))}{\alpha + 1} \end{aligned} \tag{2.43}$$

### *2.3.2 Posterior Distribution of Dirichlet Process*

In this section, we study the problem of obtaining the posterior distribution of a Dirichlet process. We assume a model as in Equation (2.40) with a Dirichlet process as the prior. Ferguson shows that given the model in Equation (2.40), the posterior distribution also follows a Dirichlet process [47].

**Theorem 2.** *Considering the following hierarchy*

$$\begin{aligned} &G \sim DP(\alpha, H) \\ &\theta_j | G \sim G \qquad\qquad j = 1, \ldots n \end{aligned} \tag{2.44}$$

*then the posterior distribution is* $DP(\alpha + n, \frac{1}{\alpha+n} \sum \delta_{\theta_i} + \frac{\alpha}{\alpha+n} H)$.

### *2.3.3 A Constructive Method: Stick-Breaking Construction*

In spite of the fact that the Ferguson's definition of the Dirichlet process is well-defined, it is not truly practical. In [48], Sethuraman provides a practical way of drawing from the Dirichlet process. It is shown that the following hierarchy presents

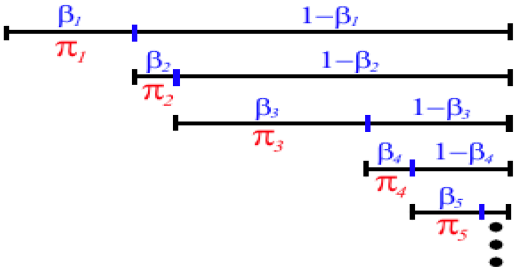


Figure 2.4: Stick-brealking Process.

a single draw from DP$(\alpha, H)$:

$$\begin{aligned} \theta_j &\overset{i.i.d.}{\sim} H \\ \pi_j &\sim \text{GEM}(\alpha), \\ G &= \sum_{j=1}^{\infty} \pi_j \delta_{\theta_j} \end{aligned} \tag{2.45}$$

where GEM$(\alpha)$ is Griffiths-Engen-McCloskey distribution defined as

$$\beta_j \overset{i.i.d.}{\sim} \text{Beta}(1, \alpha)$$

$$\pi_j = \beta_j \prod_{i=1}^{j-1} (1 - \beta_i)$$

This process is often known as stick-breaking process. Intuitively speaking, the construction of the weights $\pi_j$, $j = 1, 2, \dots$ resembles breaking off a unit length stick. In particular, given a unit length stick, $\pi_1$ is obtained by breaking the stick at a random point $\beta_1$. We choose $\beta_2$ at random and select the second weight $\pi_2$ from the remaining of the stick. The process continues and generates the whole sequence of weights $\pi_j$, $j = 1, 2, \dots$. This procedure is depicted in Figure 2.4.

**Remark1:** It is straightforward to confirm that $G$ drawn based on stick-breaking process is a random probability measure probability and is discrete with probability one. Figure 2.5 shows a draw from a Dirichlet process with Gaussian mean.

**Remark2:** Weights $\pi = (\pi_1, \pi_2, \dots,)$ is a probability measure, i.e., $\sum \pi_j = 1$.

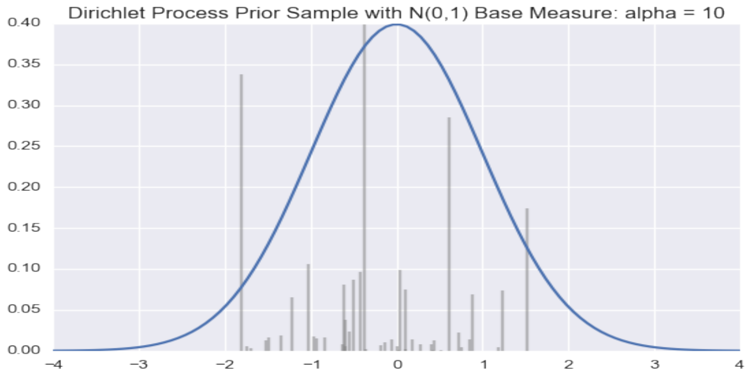


Figure 2.5: A Draw from the Dirichlet Process with Gaussian Mean and $\alpha = 10$.

**Remark3:** The weights $\pi_j$, $j = 1, 2, \ldots$ are decreasing on average but not strictly. Ordering the weights leads to the Poisson-Dirichlet process [51]. However, ordering the weights make this computationally intractable.

### *2.3.4 Dirichlet Process Mixture Model*

The Dirichlet process presents a discrete random measure which does not have densities. Therefore, it is not an appropriate prior to estimate the density. Instead, we can use a generalization of the Dirichlet process to do density estimation.

Suppose $x_1, x_2, \ldots$ are drawn independently and identically from a distribution $P$ whose density is $p$. The goal is to employ a Dirichlet process to estimate $p$. To estimate $p$, we place a Dirichlet process on the space of the parameters and draw parameters from the mean of the Dirichlet process. Each parameter may be selected with some probability according to GEM($\alpha$) and form an infinite mixture model. The infinite mixture model is known as the Dirichlet process mixture model. This infinite mixture is the same as the random distribution $P \sim \text{DP}(\alpha, H)$ which had the form $P = \sum_{j=1}^{\infty} \pi_j \delta_{\theta_j}$ except that the point mass distributions $\delta_{\theta_j}$ is smoothed out by

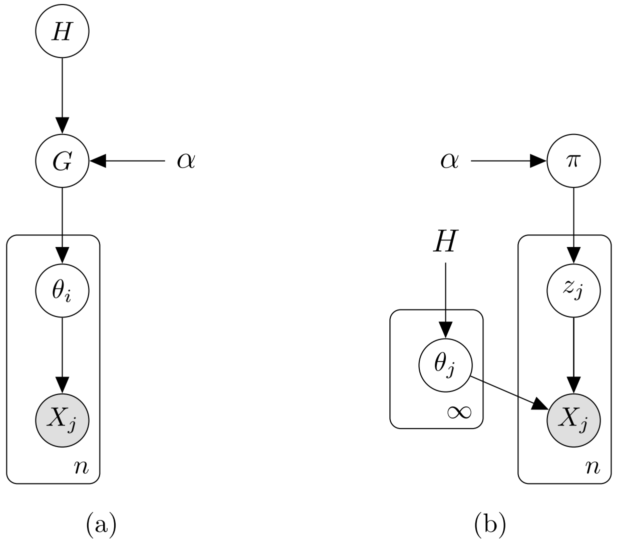


Figure 2.6: (a) Graphical Model Representing the Dirichlet Process Mixture Model. (b) Graphical Model Representing the Dirichlet Process Mixture Model Where $G$ in Marginalized.

densities $p(\cdot|\theta_j)$. The infinite mixture is modeled by the following hierarchy

$$
\begin{aligned}
&G|\alpha, H \sim \mathrm{DP}(\alpha, H) \\
&\theta_j|G \overset{i.i.d.}{\sim} G \\
&x_j|\theta_j \sim p(\cdot|\theta_j).
\end{aligned}
$$

This hierarchy is shown in Figure 2.6a. By marginalizing $G$, we obtain a model that depends only on the mean and the concentration parameter of the Dirichlet process. This representation of the infinite mixture model is given by

$$
\begin{aligned}
&\pi|\alpha \sim \mathrm{GEM}(\alpha) \\
&\theta_j|H \overset{i.i.d.}{\sim} H \\
&z_j|\pi \sim \mathrm{Cat}(\pi) \\
&x_j|\Theta, z_j \sim p(\cdot|\theta_{z_j}),
\end{aligned}
$$

where Cat($\pi$) is a categorical distribution with parameter $\pi$. The indictor variable

$z$ assigns the appropriate probability to each of the infinite parameters drawn from the base distribution of the Dirichlet process. The graphical model representing this hierarchy is depicted in Figure 2.6b. Bayesian inference methods such as Markov chain Monte Carlo(MCMC) or variational Bayes methods are provided to do inference [52, 53].

### *2.3.5 Dirichlet Process and Clustering: Chinese Restaurant Process*

Consider a Chinese restaurant with infinitely many tables. The first customer comes into the restaurant and sits at the first table with probability one. As customers enter the restaurant, they choose an occupied table with probability proportional to the number of customers that are already seated at the table or choose a new table with probability proportional to $\alpha$. This analogy leads to the Chinese restaurant process.

For a fixed $\alpha > 0$ and for every $n \in \mathbb{N}$, the Chinese restaurant process, CRP($\alpha$), is a distribution over all partitions of the set $[n] := \{1, 2, ..., n\}$. A draw from the CRP($\alpha$), $\rho \sim$ CRP($\alpha$), provides a partition on $[n]$. Subsets of the partition and data points are referred to as tables/clusters and customers, respectively. The Chinese restaurant process is mathematically constructed as follows; each customer comes into the restaurant and picks a table at random with probability:

$$\begin{aligned} \mathbb{P}(\text{Choose table}\, \mathcal{C}) &= \frac{n_{\mathcal{C}}}{\alpha + \sum_{\rho} n_{\mathcal{C}}} \\ \mathbb{P}(\text{Choose a new table}) &= \frac{\alpha}{\alpha + \sum_{\rho} n_{\mathcal{C}}} \end{aligned} \tag{2.46}$$

where $n_{\mathcal{C}}$ is the number of customers at the table $\mathcal{C}$. This process is depicted in Figure 2.7. The CRP is an example of the preferential attachment which is proven to be exchangeable.

**Definition:** A random partition is called exchangeable if its distribution is invariant

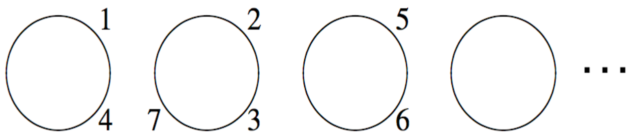


Figure 2.7: Chinese Restaurant Process.

under permutation. Equivalently, a random partition is exchangeable if there is a symmetric function $p$ such that probability of each partition only depends on the size of each subset, i.e., for the random partition $\rho = \{\mathcal{C}_1, \dots, \mathcal{C}_k\}$

$$\mathbb{P}(\rho) = p(|\mathcal{C}_1|, \dots, |\mathcal{C}_k|). \tag{2.47}$$

The function $p$ is called the *exchangeable partition probability function* (EPPF).

The Chinese restaurant process is partition exchangeable, therefore, for a partition $\rho = \{\mathcal{C}_1, \dots, \mathcal{C}_k\}$ the EPPF induced by the distribution over the partition is given by

$$\mathbb{P}\big(|\mathcal{C}_1|, \dots, |\mathcal{C}_k| \big| \alpha\big) = \frac{\alpha^K}{\alpha^{[n]}} \prod_j (|\mathcal{C}_j| - 1)! \tag{2.48}$$

where $\alpha^{[n]} = \alpha(1+\alpha)\dots(\alpha+n-1)$. The Chinese restaurant process is not sequence exchangeable (de Finetti's theorem). However, there is close relationship between the partition exchangeability and sequence exchangeability. A sequence exchangeable is constructed as follows:

- For each $\mathcal{C} \in \rho$ define $\theta^\star_{\mathcal{C}} \sim H$
- For each $j \in [n]$ define $\theta_j = \theta^\star_{\mathcal{C}}$, where $\mathcal{C} \in \rho$ and $j \in \mathcal{C}$

The resulting sequence $\theta_1, \theta_2, \dots$ is de Finetti exchangeable. To summarize the procedure of constructing of random sequence from a random partition, we follow the hierarchy

$$\begin{aligned} \rho &\sim \mathrm{CRP}(\alpha) \\ \theta^\star_{\mathcal{C}} &\sim H \quad \text{for each}\, \mathcal{C} \in \rho \\ \theta_j &= \theta^\star_{\mathcal{C}} \quad \text{for each}\, j \in [n],\, \mathcal{C} \in \rho, j \in \mathcal{C} \end{aligned} \tag{2.49}$$

**Theorem 3.** *(Aldous 1985)* Assume the sequence $\theta_1, \theta_2, \ldots$ is generated as in Equation (2.49). This sequence is de Finetti exchangeable and therefore, there is a random probability measure under which the data is conditionally independent. The underlying distribution is a Dirichlet process.

### *2.3.6 The Blackwell-MacQueen Distribution: Pólya Urn Scheme*

The Blackwell-MacQueen is a generalization of the Pólya urn that essentially captures the Chinese restaurant process model discussed in Section 2.3.5. Let $\theta_1, \ldots \theta_n$ be the parameter associated with clusters (not necessarily unique). The predictive distribution is

$$\theta_{n+1} | \theta_1, \ldots \theta_n \sim \frac{1}{\alpha + n} \sum_{j=1}^{n} \delta_{\theta_j} + \frac{\alpha}{\alpha + n} H \tag{2.50}$$

where $\delta_{\theta_j}$ is the point mass at $\theta_j$ and $H$ is the base distribution. The distribution on the sequence of $\theta$ has the Blackwell-MacQueen distribution [50]. Assuming that $\theta^*_1, \ldots, \theta^*_K$ are the unique parameters, the predictive distribution can be re-written as

$$\theta_{n+1} | \theta_1, \ldots \theta_n \sim \sum_{j=1}^{K} \frac{n_j}{\alpha + n} \delta_{\theta^*_j} + \frac{\alpha}{\alpha + n} H \tag{2.51}$$

where $n_j = \left| \{ i : \theta_i = j \} \right|$.

### *2.3.7 Hierarchical Dirichlet Process Modeling*

Often in statistics, we wish to divide the data into groups such that the dependency and statistical strength among the groups are preserved. In classical Bayesian

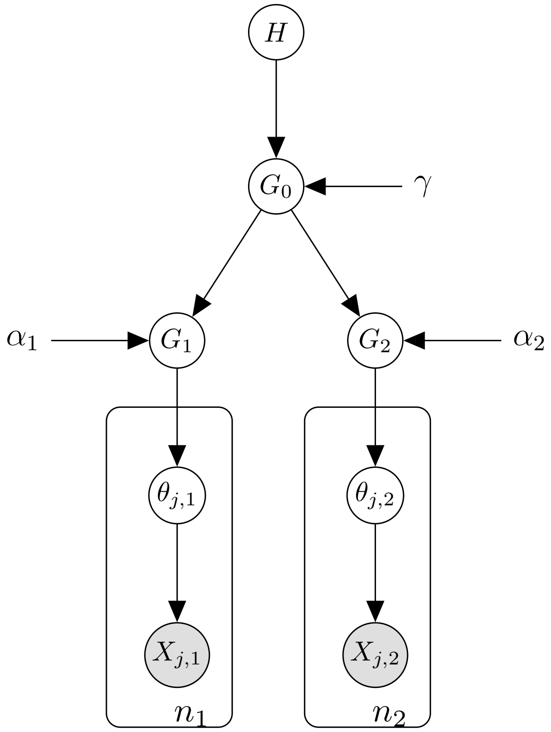


Figure 2.8: Hierarchical Dirichlet Process Mixture Model for 2 Groups.

statistics, a hierarchical modeling is typically employed to allow the groups to remain linked. The hierarchical Dirichlet process (HDP) provides a nonparametric hierarchical framework that captures the dependency among the groups [54]. In particular, assume that the random measure $G_j$ which represents the $j$th group is a conditionally independent draw from a Dirichlet process $\mathrm{DP}(\alpha, G_0)$. To maintain the dependency among all groups, the hierarchical modeling suggests a discrete prior on $G_0$. Assume that $G_0$ is absolutely continuous probability measure with respect to the Lebesgue measure, then there is *almost surely* no shared cluster among the groups. Thus, a nonparametric hierarchical model assumes that $G_0$ is itself drawn from a Dirichlet process as shown in Figure 2.8. Adding one more level of Dirichlet process over the base distribution guarantees that the HDP shares countable infinite cluster parameters among the groups. The hierarchical Dirichlet process is modeled as:

$$\begin{aligned} &G_0 \sim DP(\gamma, H) \\ &G_m|G_0 \sim DP(\alpha, G_0) \\ &\theta_{j,m}|G_m \sim G_m. \end{aligned} \tag{2.52}$$

Note that there are equivalent constructions of the HDP which are analogous to the DP constructions. The Chinese restaurant franchise which is the generalization of Chinese restaurant process is an equivalent method of HDP construction [54]. A stick-breaking definition of HDP is also discussed in [54]. Furthermore, we may generalize the hierarchical Dirichlet process in Equation (2.52) to the hierarchical Dirichlet process mixture to estimate the density using an infinite mixture model. The hierarchical Dirichlet process mixture model is given by

$$\begin{aligned} &G_0 \sim DP(\gamma, H) \\ &G_m|G_0 \sim DP(\alpha, G_0) \\ &\theta_{j,m}|G_m \sim G_m \\ &x_{j,m}|\theta_{j,m} \sim f(\cdot|\theta_{j,m}). \end{aligned} \tag{2.53}$$

## 2.4 Two-Parameter Poisson-Dirichlet Process

As discussed in Section 2.3, the Dirichlet process is a distribution over an infinite dimensional space. Regardless of generating an infinite number of clusters, the rate at which clusters are generated in slow. It is easy to show that for a Dirichlet process, the expected number of clusters after observing $n$ data points is $\alpha \log n$. However, many phenomena can be characterized by the growth of polynomial known as power law [55, 56]. Pitman and Yor introduce a random probability measure that induces marginal distributions characterized by a two-parameter Chinese restaurant process.

A two-parameter Chinese restaurant process, CRP$([n], d, \alpha)$, is a distribution over all partitions with two parameters $\alpha > 0$ and discount parameter $d$ such that $\alpha > -d$

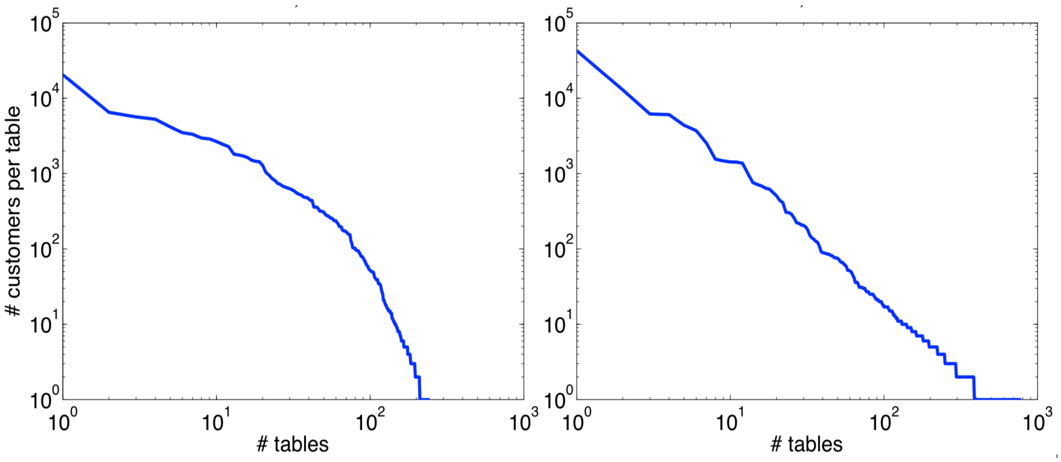


Figure 2.9: Heap's Law for Pitman-Yor Process.

and $0 \leq d < 1$. The probability of choosing a table (cluster) is given by

$$\begin{aligned} \mathbb{P}(\text{Choose table}\,\mathcal{C}) &= \frac{n_{\mathcal{C}} - d}{\alpha + \sum_{\rho} n_{\mathcal{C}}} \\ \mathbb{P}(\text{Choose a new table}) &= \frac{\alpha + d|\rho|}{\alpha + \sum_{\rho} n_{\mathcal{C}}}. \end{aligned} \tag{2.54}$$

The two-parameter Chinese restaurant process is an exchangeable process [56]. Consequently, there exist a probability measure (de Finetti's distribution) such that the data are conditionally independent of one another. The de Finetti measure is known as two-parameter Poisson-Dirichlet process or Pitman-Yor process [56, 57]. It is shown that expected number of generated clusters through Pitman-Yor process follows a power law, and therefore is more suitable for phenomena that follow power law such as text [54, 58].

Equation (2.54) verifies the willingness of Pitman-Yor process to generate more clusters; tables with more occupants are more likely to become even larger and tables with small occupancy numbers tend to have a lower chance of getting new customers. However, bigger values of the discount parameter $d$ tends to have more tables with fewer customers. As shown in Figure 2.9, Pitman-Yor process follows Heap's law where the Dirichlet process tends to have less number of tables.

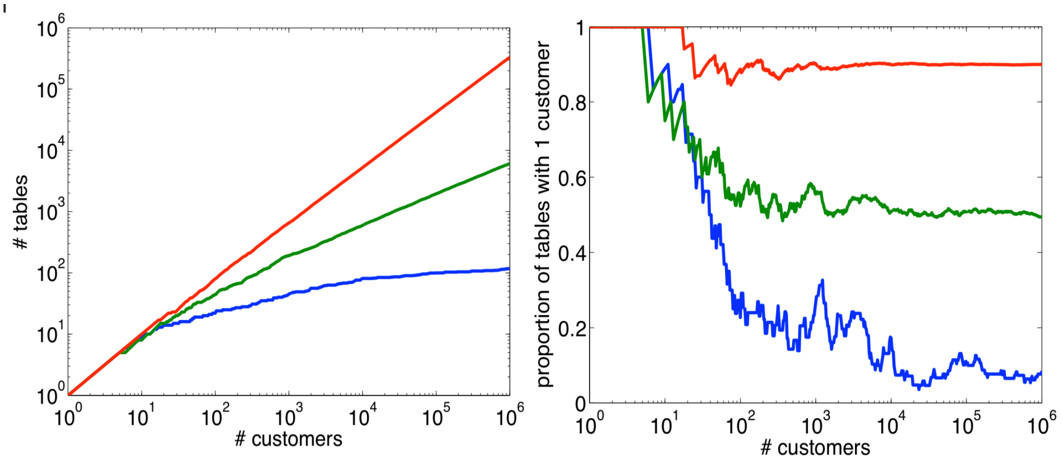


Figure 2.10: Comparison between Pitman-Yor Process and Dirichlet Process for $\alpha = 10$ and $d = 0.9$ (Red), $d = 0.5$ (Green), and $d = 0$ (Blue).

Figure 2.10 compares the Dirichlet process with $\alpha = 10$ to the Pitman-Yor process with $\alpha = 10.$ and $d = 0.9$, $d = 0.5$. We can observe that (a) as the discount parameter $d$ grows, the Pitman-Yor process tends to have more tables with less occupants, (b) larger values of $d$ makes the proportion of tables and customers to follow Zip's law (c) as the number of customers grows, the Dirichlet process tends to have fewer tables where the Pitman-Yor tends to create more tables as required [58]. It is shown that after observing $n$ data points, the expected number of clusters generated though the Pitman-Yor process is $\alpha n^d$.

## 2.5 Inferential Methods

Invention of inferential methods makes the inference for high-dimensional data possible [59–61]. These methods are most useful when it is difficult or impossible to explicitly compute some probability distributions given parameters. In particular, we discuss two main inference methods: Monte Carlo methods and variational Bayes. It is shown that by designing efficient algorithms Monte Carlo methods can produce accurate, exact, and tractable estimates [62]. However, variational Bayes models are an approximation for intractable integrals or posterior distributions [63].

Although, MCMC methods provide a precise estimate to the problem of inference, these methods are expensive for large data. To resolve this issue, we can settle for an approximation rather than the exact solution. Variational Bayes methods offer the approximate solution. These methods may be much faster to sample in high-dimensional data.

### *2.5.1 Monte Carlo Methods*

Markov chain Monte Carlo (MCMC) methods are the most used inferential methods which provide exact samples from the target distribution for any problem with probabilistic interpretation with some parameters, for example, many problems in machine learning, optimization, and statistics. These inferential methods utilize independent samples of distribution to analyze the distribution for which explicit computation of the distribution is difficult.

Many inference tasks such as computing the marginal can be represented as the integral, and therefore as the expected value of some appropriately chosen function [64, 65]. Consequently, due to the law of large number, the expected value can be described as the empirical mean of independent random variables.

There are various Monte Carlo methods that offer different approaches to generate independent samples; most of which are based on the random walk. These methods are based on choosing a proposal distribution, and thus are very sensitive to the step size. The primary idea is to design a first order Markov chain with the target stationary probability distribution where the distribution of the samples converges to the target distribution asymptotically. In addition, the ergodic theorem indicates that the stationary distribution of a Markov chain may be approximated by the empirical measures of the random states of the MCMC sampler [59]. The following theorem is the fundamental idea behind the Monte Carlo methods.

**Theorem 4.** the following statements are equivalent:

(i) $x \sim p(x)$

(ii) $(x, u) \sim \mathcal{U}nif\{(x, u) : 0 \leq u \leq p(x)\}$.

In the following sections, we briefly study the inferential models that are primarily used to develop the methods in this thesis. We discuss MCMC methods which are designed according to a random walk process. In addition to the random walk based MCMC methods, we explore the slice sampling method to solve the issues arising from the random walk modeling.

### *2.5.2 Generalized Importance Sampling*

Importance sampling approach is an MCMC sampling method to estimate the expected value. This method exploits a proposal distribution, and thus relies upon importance functions. Suppose $p(x) = \bar{p}(x)/Z$ can be evaluated up to the normalizing constant $Z$. A proposal distribution $q(x)$ is chosen such that $q(x)$ is absolutely continuous with respect to $p(x)$. In particular, $\operatorname{supp}(p(x)) \subset \operatorname{supp}(q(x))$. Assume $x_1, \ldots, x_N \sim q(x)$, then for any function $h$

$$\frac{1}{N}\sum_{j=1}^{N} \omega_j h(x_j) \xrightarrow{N\to\infty} \mathbb{E}_p[h(x)] = \int h(x)q(x)\frac{p(x)}{q(x)}dx \tag{2.55}$$

where $\omega_j = \frac{\tilde{\omega}_j}{\sum_j \tilde{\omega}_j}$ and $\tilde{\omega}_j = \frac{\bar{p}(x_j)}{q(x_j)}$. This estimation is asymptotically consistent [60]. Moreover, the Equation (2.55) indicates that the expected values can be estimated using the importance functions $\{\tilde{\omega}_j\}_{j=1}^{N}$.

In this section, we provide a general framework for importance sampling based on dependent proposal distributions and adaptive algorithms where it provides an

**Algorithm 1:** Dynamic Importance Sampling.

```
Input: $(x, \tilde{\omega})$, and $\mathbb{K}(x, x')$ where $\tilde{\omega} = \frac{\bar{p}(x)}{q(x)}$
for k = 1,2, ... do
    Draw $x' \sim \mathbb{K}(x_k, x')$
    Compute $\gamma_k = \tilde{\omega}\frac{p(x')\mathbb{K}(x', x_k)}{p(x)\mathbb{K}(x_k, x')}$
    Draw $u \sim \mathcal{U}nif(0, 1)$
            $(x_{k+1}, \tilde{\omega}_{k+1}) \leftarrow (x', \frac{(1+\delta)\gamma_k}{c})$ if $u < c$
            $(x_{k+1}, \tilde{\omega}_{k+1}) \leftarrow (x_k, \frac{(1+\delta)\tilde{\omega}_k}{1-c})$ if $u > c$
    where $c = \frac{\gamma_k}{\gamma_k + \eta(x_k, \omega_k)}$, $\delta > 0$ and $\eta$ are either constant or independent
end for
```

unbiased estimator for the target expected value. It is proven that dependency in the samples still preserves the unbiasedness property [66]. The following lemma shows that the modification of importance weights by a kernel preserves the unbiasedness of the estimator.

**Lemma 1.** if $p$ and $q$ are distributions such that $p \ll q$ and importance weight $\tilde{\omega} = \frac{p(x)}{q(x)}$, then for any kernel $\mathbb{K}(x, x')$ with stationary distribution $p$

$$\int \tilde{\omega}\mathbb{K}(x, x')q(x) = p(x'). \tag{2.56}$$

Since the kernel $\mathbb{K}(x, x')$ corresponds to the target distribution, it can correct the poor choice of proposal distribution. A dynamic approach to importance sampling is introduced in [67]. We summarize this method in Algorithm 1.

### *2.5.3 Metropolis-Hastings Algorithm*

Metropolis-Hastings algorithm is the universal MCMC algorithm where it produces an ergodic Markov chain whose stationary distribution is the target distribution $p(x)$. In particular, to sample from the posterior distribution $p(\theta|z)$, samples $\theta_k$

**Algorithm 2:** Metropolis-Hastings Algorithm.

```
Input: proposal distribution $q(\cdot|\cdot)$
Initialize $x_0$ at random
for k=0,1,2,... do
    Draw $y_{k+1} \sim q(\cdot|x_k = x_k)$
    Draw $u_{k+1} \sim \mathcal{U}nif(0,1)$
    Compute the acceptance probability $\alpha(x_k, y_{k+1})$ in Equation (2.58)
    if $u_{k+1} \leq \alpha(x_k, y_{k+1})$ then
        $x_{k+1} \leftarrow y_{k+1}$
    else
        $x_{k+1} \leftarrow x_k$
    end if
end for
Burn-in Dismiss the first $x_1, \dots x_r$
```

are sequentially drawn from a Markov chain with stationary distribution $p(\theta|z)$; we construct a Markov chain $\mathbb{K}$ such that for sufficiently large $k$, $\theta_k$ is drawn from the desired posterior distribution, i.e., $\mathbb{K}^k \to p(\theta|z)$.

Suppose that Markov chain $\mathbb{K}$ is irreducible[2] and aperiodic[3] whose stationary distribution is $p$. The stationary distribution follows the detailed balance condition,

$$\mathbb{K}(x, y)p(y) = \mathbb{K}(y, x)p(x). \tag{2.57}$$

The Metropolis-Hastings algorithm starts by selecting an easy to implement conditional distribution $q(\cdot|\cdot)$ which is absolutely continuous with respect to the target distribution $p$. Without loss of generality, we can assume $q(x|y)p(x) > q(y|x)p(y)$,

[2] All states can communicate with one another with positive probability in finite time.

[3] To ensure uniqueness of stationary distribution almost surely.

and hence there exist an acceptance probability $0 \leq \alpha(x, y) \leq 1$ such that

$$q(x|y)p(x) = \alpha(x, y)q(y|x)p(y) \implies \alpha(x, y) = \min\Big\{1, \frac{q(x|y)p(x)}{q(y|x)p(y)}\Big\}. \tag{2.58}$$

It is shown in [68], the transition kernel associated with this equation follows

$$\mathbb{K}(x, \Theta) = \int_{\Theta} \alpha(x, y)q(y|x)dy + \mathbb{1}_x(\Theta)\Big(1 - \int_{\Theta} \alpha(x, y)q(y|x)dy\Big). \tag{2.59}$$

The Metropolis-Hastings algorithm associated with the target density $p$ with conditional proposal distribution $q$ produces a Markov chain $\{x_k\}_k$ using Equation (2.59). This method is referred to as Metropolis-Hastings algorithm and summarized in Algorithm 2.

**Theorem 5.** Suppose that the Markov chain produced by Metropolis-Hastings is $p$-irreducible, then

a) For any function $g \in L_1(p)$,

$$\lim_{N\to\infty} \frac{1}{N} \sum_{k=1}^{N} g(x_k) = \int g(x)d\mathbb{P}(x) \tag{2.60}$$

b) If $x_k$ is aperiodic, then for every initial distribution $\nu$

$$\lim_{N\to\infty} || \int \mathbb{K}^N(x, \cdot)\nu(dx) - p||_{TV} = 0. \tag{2.61}$$

Choice of $q$ result in different Metropolis-Hastings algorithms. We study two main choices of $q$ next.

### Independent Metropolis-Hastings

Assume $q(x|y)$ is independent of $y$, that is, $q(y|x) = q(y)$. This leads to an algorithm called Independent Metropolis-Hastings algorithm. In this case, the acceptance probability $\alpha(x, y)$ is simplified to

$$\alpha(x, y) = \min\Big\{1, \frac{q(x)p(x)}{q(y)p(y)}\Big\}. \tag{2.62}$$

Although $y_k$'s are generated independently in Algorithm 2, the resulting samples $x_k$'s are not i.i.d. since, for instance, probability of acceptance of $y_k$ relies directly on $x_k$.

### Random Walk Metropolis-Hastings

Assume $q$ is symmetric, that is, $q(x|y) = q(y|x)$. This leads to a method which is referred to as Metropolis-Hastings random walk algorithm. In this case proposal distribution $q$ depends only on $|x-y|$. In this case, the acceptance probability $\alpha(x, y)$ is given by

$$\alpha(x, y) = \min\left\{1, \frac{p(x)}{p(y)}\right\}. \tag{2.63}$$

Despite simplicity of computing the acceptance probability, this method tends to converge with slower rate. In addition, the random walk Metropolis-Hastings algorithm does not satisfy the uniform ergodicity property [69].

### *2.5.4 Gibbs Sampling*

Gibbs sampling, also known as alternating conditional sampling, is a special case of the Metropolis-Hastings algorithm where we partially update our joint distribution. Gibbs sampler is mostly used when computing the conditional distribution is straightforward. The idea is to use the conditional distribution associated with the target distribution to generate samples from it. In the section, we first study the Gibbs sampler for two variables and then generalize it to a vector of random variables with undemanding conditional distributions. Gibbs sampling can be very slow if the parameters in target distribution are highly correlated. To avoid this issue, one can re-parametrize the parameters of interest.

**Algorithm 3:** Two-Stage Gibbs Sampler to Sample from $p(x, y)$

```
Initialize $(x_0, y_0)$
**for** k=1,2,... **do**
  $x_k \sim p(\cdot|y_{k-1})$
  $y_k \sim p(\cdot|x_k)$
**end for**
Repeat until convergence
```

**Two-Stage Gibbs Sampler**

Consider the joint probability density $p(x, y)$ on the product space $\mathcal{X} \times \mathcal{Y}$. Using Theorem 4, define $\mathcal{E}(p) = \{(x, y, u) : 0 \leq u \leq p(x, y)\}$. We generate

- $x$ uniformly on $\mathcal{E}_x(p) = \{x : u \leq p(x, y)\}$ or equivalently from $\mathcal{E}_x(p) = \{x : \frac{u}{p_y(y)} \leq p(x|y)\}$.
- $y$ uniformly on $\mathcal{E}_y(p) = \{y : u \leq p(x', y)\}$ or equivalently from $\mathcal{E}_y(p) = \{y : \frac{u}{p_x(x)} \leq p(y|x')\}$.
- $u$ uniformly on $\{u : 0 \leq u \leq p(x', y')\}$.

However, if we leave $y$ fixed and repeat this procedure infinite times, we end up with the samples from $p(x|y)$. One can do the same along $y$-axis and end up with the samples from $p(y|x)$. We summarize this procedure in Algorithm 3. To illustrate the two-state Gibbs sampling procedure, we assume $\mathbb{X} = (x, y) \sim \mathcal{N}(\mu, \Sigma)$ for unknown mean $\mu = (\theta_1, \theta_2)$ and known covariance matrix

$$\Sigma = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}.$$

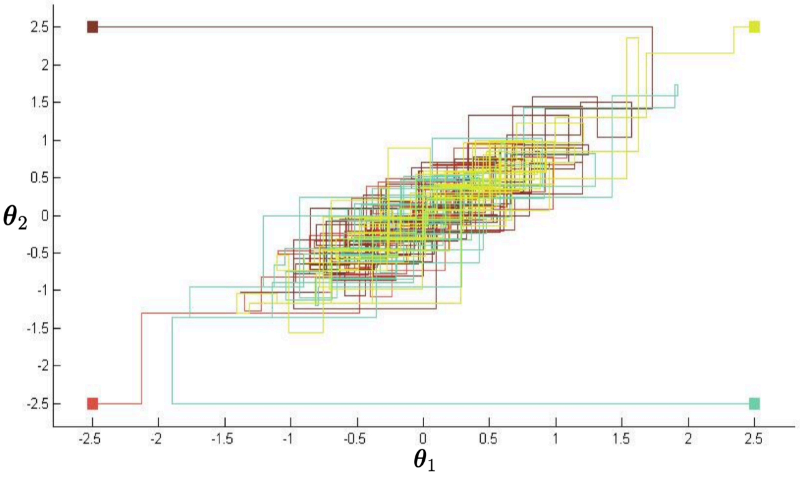


Figure 2.11: Gibbs Sampler for A Bivariate Gaussian Distribution with 10,000 Simulations.

Assuming a uniform distribution as prior on $\mu$, according to Section 2.1.1, the conditional posterior distribution is given by

$$\begin{aligned} \theta_1|\theta_2, \mathbb{X} &\sim \mathcal{N}(x+\rho(\theta_2-y), 1-\rho^2) \\ \theta_2|\theta_1, \mathbb{X} &\sim \mathcal{N}(y+\rho(\theta_1-x), 1-\rho^2). \end{aligned} \tag{2.64}$$

Figure 2.11 demonstrates the Gibbs sampler for this model for 10,000 iterations and $\rho = 0.5$ and burn in $r = 100$

### Multi-Stage Gibbs Sampler

Multi-stage Gibbs sampler is the natural generalization of two variables to $L$ variables. Suppose for any $j$,

$$x_j|x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_L \sim p_j(\cdot|x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_N) \tag{2.65}$$

is simply computed. Then, we can naturally generalize the two-stage Gibbs sampler to a multi-stage Gibbs sampler. The multi-stage Gibbs sampler is summarized in Algorithm 4.

**Algorithm 4:** Multi-Stage Gibbs Sampler to Sample from $p(x_1, \dots, x_N)$.

```
Initialize $(x_{1,0}, \dots, x_{N,0})$
for k=1,2,... do
    $x_{1,k} \sim p_1(\cdot|x_{2,k-1} \dots, x_{N,k-1})$
    $x_{2,k} \sim p_2(\cdot|x_{1,k}, x_{2,k-1} \dots, x_{N,k-1})$
              $\vdots$
    $x_{L,k} \sim p_N(\cdot|x_{1,k}, \dots, x_{N-1,k})$
end for
Repeat until convergence
```

Despite the polynomial bounds for the Gibbs sampler convergence to the stationary distribution, it is difficult to guarantee the convergence in high-dimensional models [43, 70, 71]. One method to resolve this issue is *blocked Gibbs sampling* which rather than sampling the individual variables, it samples from a group of random variables that are assumed to be highly correlated [70, 71]. Another method to improve the convergence rate of Markov chain is *auxiliary variable methods* [60, 62]. The auxiliary variable methods introduce an auxiliary random variable $u$ to sample from the joint distribution $p(x, u)$ rather than the target distribution $p(x)$. Marginalizing the joint distribution leads to sampling from the target distribution $p(x)$.

### The Expectation Minimization Algorithm and Gibbs Sampling

The Expectation Minimization (EM) algorithm first introduced by Dempster to address the problem of maximizing the likelihood of incomplete data [72]. The original method is not considered a stochastic approach for parameter estimation. However, there is a close relationship between the EM and Gibbs sampling algorithms.

Suppose $x_1, \dots, x_N \sim p(x|\theta)$ where $p(x|\theta)$ is the density of incomplete data. The idea is to use the Bayes rule and augment latent variables $z$'s such that $(x, z) \sim p(x, z|\theta)$ is drawn from the complete data density. The EM algorithm aims to estimate the unknown parameters $\theta$ via maximum likelihood (MLE), that is,

$$\hat{\theta}_{\text{MLE}} = \arg\max_{\theta} \log p(x|\theta) = \arg\max_{\theta} \log \int_z p(x, z|\theta) dz \tag{2.66}$$

or estimate the parameters through maximum a posteriori (MAP), that is,

$$\hat{\theta}_{\text{MAP}} = \arg\max_{\theta} \log \int_z p(x, z, \theta) dz = \arg\max_{\theta} \big( \log \int_z p(x, z|\theta) dz + \log p(\theta) \big). \tag{2.67}$$

However, maximizing the likelihood is often troublesome since the integral is taken over a multimodal distribution. Instead, we find a lower bound and maximize the lower bound to be the closest to the log-likelihood. Consider the following log-likelihood function

$$\begin{aligned} \log p(x|\theta) = \log \int_z p(x, z|\theta) dz &= \log \int_z q(z) \frac{p(x, z|\theta)}{q(z)} dz \\ &= \log \mathbb{E}_q \Big[ \frac{p(x, z|\theta)}{q(z)} \Big] \\ &\geq \mathbb{E}_q \log \Big[ \frac{p(x, z|\theta)}{q(z)} \Big] \\ &= \mathbb{E}_q \log p(x, z|\theta) - \mathbb{E}_q \log q(z) = \text{ELBO}_q(\theta) \end{aligned} \tag{2.68}$$

where the inequality is due to the Jensen's inequality. The $\text{ELBO}_q(\theta)$ is called the variational lower bound. Equality in Equation (2.68) holds if and only if

$$q(z) = \log \Big[ \frac{p(x, z|\theta)}{q(z)} \Big] \tag{2.69}$$

is affine. This condition is obtained if $q(z) = p(x|z, \theta)$. The inequality in Equation (2.68) is the fundamental equation in variational methods. The EM algorithm via MLE maximizes the log-likelihood in two steps as a coordinate ascent iteration on the log-likelihood [73]:

$$\begin{aligned} &\textbf{E-Step:} \qquad q^{k+1} = \arg\max_{q} \mathrm{ELBO}_q(\theta^k) \\ &\textbf{M-Step:} \qquad \theta^{k+1} = \arg\max_{\theta} \mathrm{ELBO}_{q^{k+1}}(\theta) \end{aligned} \tag{2.70}$$

It can be shown that the E-Step is equivalent to set $q^{k+1}(z) = p(z|x, \theta^k)$. Intuitively speaking, the EM algorithm alternates between updating $q$ and $\theta$ first by setting $q(z) = p(z|x, \theta)$ to obtain $\log(p(x|\theta)) = \mathrm{ELBO}_q(\theta)$ for a fixed $\theta$ and then by maximizing $\mathrm{ELBO}_q(\theta)$ with respect to $\theta$ for a fixed $q$. This method converges to a local maxima. Maximization in Equation (2.67) is analogous to the aforementioned method. However, the M-Step is a MAP estimate rather than a MLE estimate.

Although the EM algorithm is not a stochastic algorithm, it is linked to the two-stage Gibbs sampling algorithm in the sense that rather than maximizing in the steps of the EM algorithm, we sample from the conditional distribution.

### Application of EM Algorithm in Bayesian Inference

We can exploit the EM algorithm to estimate the mode of the posterior distribution, $p(\theta|x)$. Suppose $p(\theta)$ is the prior on the unknown parameters $\theta$. Bayes' rule suggests

$$\begin{aligned} \log p(\theta|x) &= \log p(x|\theta) + \log p(\theta) - \log p(x) \\ &= \mathrm{ELBO}_q(\theta) + \log p(\theta) - \log p(x). \end{aligned} \tag{2.71}$$

For all distribution $q$, one can rewrite Equation (2.68) as follows:

$$\begin{aligned} \mathrm{ELBO}_q(\theta) &= \int_z q(z) \log \left[\frac{p(x, z|\theta)}{q(z)}\right] dz - \int_z q(z) \log \left[\frac{p(z|x, \theta)}{q(z)}\right] dz \\ &= \mathcal{L}(q, \theta) + \mathrm{KL}(q, p_{z|x,\theta}) \end{aligned} \tag{2.72}$$

where $\mathcal{L}(q, \theta)$ equals to the negative free energy and $\mathrm{KL}(\cdot, \cdot)$ is the Kullback-Leibler divergence [73]. By substituting Equation (2.72) into the Equation (2.71), we can use the EM algorithm to find the posterior mode.

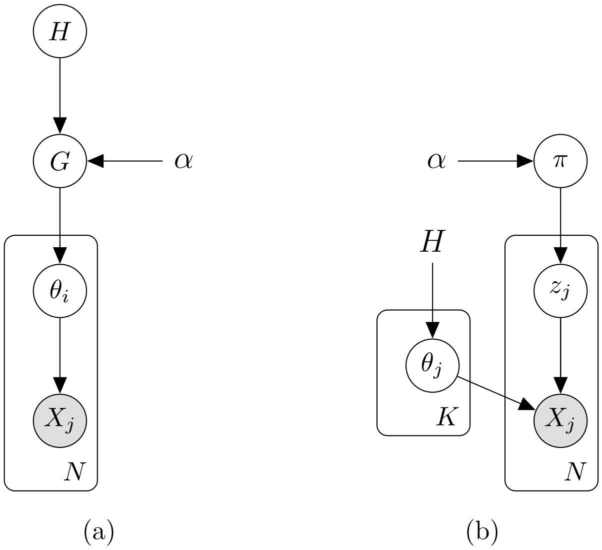


Figure 2.12: (a) Graphical Model Representing Finite Mixture Model (K-clusters) $\pi \sim \mathrm{Dir}(\alpha/K, \ldots, \alpha/K)$ and $\theta_i \sim H$. (b) Graphical Model Representing $K$-finite Mixture Model Where $G$ Is Marginalized.

### Gibbs Sampler for Finite Mixture Models

Finite mixture models propose a clustering method for heterogeneous unknown populations. Let $c_j \in \{1, \ldots, K\}$ be the latent cluster indicator where $K$ unique clusters is assumed. For the observed data $x = \{x_j\}_{j=1}^{N}$, the simplest mixture model given the mixing distribution $\pi$ and cluster distribution $H$ is summarized as

$$\begin{aligned} c_j|\pi &\sim \pi \\ x_j|c_j &\sim p(\theta_{c_j}) \end{aligned} \tag{2.73}$$

where $\pi = (\pi_1, \ldots, \pi_K)|\alpha \sim \mathrm{Dir}(\alpha/K, \ldots, \alpha/K)$ and $\theta \sim H$. Note that $\theta_j$ is the $j$th cluster parameter. Equation (2.73) can be re-written as follows:

$$p(x|\pi, \{\theta_j\}_{j=1}^{K}) = \sum_{j=1}^{K} \pi_j p(x|\theta_j). \tag{2.74}$$

Using Bayes' rule and Equation (2.74)), the Gibbs sampler for the finite mixture model is given by

$$p\big(c_j = k | c_{-j}, x, \pi, \{\theta_j\}_{j=1}^{K}\big) \propto \pi_k p(x_j|\theta_k) \tag{2.75}$$

$$\pi | c_1, \ldots, c_K, \alpha \sim \mathrm{Dir}(\frac{\alpha}{N_1}, \ldots, \frac{\alpha}{N_K}), \qquad N_j = \sum_{i=1}^{N} \mathbb{1}_{c_i}(j) \tag{2.76}$$

$$p(\theta_k = \theta | \pi, c_1, \ldots, c_K, x, \alpha) \propto h(\theta) \prod_{\{j:c_j=k\}} p(x_j|\theta) \tag{2.77}$$

where $c_{-j}$ in Equation (2.75) indicates all cluster indicators with the exception of $c_j$ and $h(\theta)$ is the density associated with the distribution $H$. This method is not as efficient as collapsed Gibbs sampling [52]. We study an efficient implementation of the finite mixture models next.

### *2.5.5 Rao-Blackwellization*

Rao-Blackwellization is a general method to construct a tractable Monte Carlo method to improve the estimator. This method is based on the Rao-Blackwell theorem [41, 74] which reduces the variance of an estimator by conditioning.

**Theorem 6.** ***(Rao-Blackwell Theorem)*** Suppose $x$ and $z$ are independent random variables and $T(x, z)$ is a scaler statistics. Then,

$$\mathrm{Var}_{p_z}(\mathbb{E}_{p_x}[T(x, z)|z]) \leq \mathrm{Var}_{p_{x,z}}(T(x, z)) \tag{2.78}$$

in square error loss.

The variance reduction guaranteed in Theorem 6 can be used in sampling algorithms, e.g., Gibbs sampler, to much more rapidly estimate $\mathbb{E}_p[h(x)]$. In other words, if $T(x)$ is an estimator of $\mathbb{E}_p[h(x)]$ and $x$ can be simulated from distribution $p(x, z)$ such that it satisfies the marginal distribution of $p(x)$, then the estimator $T^*(X) = \mathbb{E}_p[T(x)|z]$ has the smaller variance. One can apply the same results for $z$ [42, 74]. We construct a tractable Monte Carlo method in the next section in detail.

### Rao-Blackwellized Sampling Schemes

Let $p(x)$ be the target distribution. We introduce an auxiliary random variable $z$ with joint distribution $p(x, z)$ such that the marginal distribution satisfies $p(x) = \int_z p(x, z)dz$. The idea is to sample from the joint distribution rather than the marginal distribution. To this end, suppose the conditional density $p(x|z)$ has a tractable analytic form and $N$ samples $\{(x_j, z_j)\}_{j=1}^N$ are drawn from the joint distribution $p(x, z)$. For the statistics $T(x, z)$, the expected values is estimated as

$$\mathbb{E}_{P_{x,z}}[T(x, z)] \approx \frac{1}{N}\sum_{j=1}^{N} T(x_j, z_j) = \mathbb{E}_{P^*_{x,z}}[T(x, z)] \tag{2.79}$$

where $P^*$ is the empirical distribution. Assuming $p(x|z)$ is tractable the alternative estimator is

$$\begin{aligned} \mathbb{E}_{p_{x,z}}[T(x, z)] &= \int_z \Big[\int_x T(x, z)p(x|z)dx\Big]p(z)dz \\ &\approx \frac{1}{N}\sum_{j=1}^{N}\int_x T(x, z_j)p(x|z_j)dx = \mathbb{E}_{P^*_z}\mathbb{E}_{P_{x|z}}[T(x, z)|z]. \end{aligned} \tag{2.80}$$

These estimators are both unbiased estimators; therefore, due to Rao-Blackwell theorem the latter estimator has s lower variance than the former. Intuitively, the second estimator is more concentrated and has less random variables over which to iterate at each step. Generally speaking, Rao-Blackwellization may improve the efficiency of the samplers like Gibbs sampler and may quickly estimate parameters with high posterior probability [42, 75, 76]. While Rao-Blackwellizaion improves the accuracy of samplers, convergence diagnostic is critical [42].

### Rao-Blackwellized Gibbs Sampling for Finite Mixture Model

To show the design of Rao-Blackwellized samplers, we directly compute the predictive distribution of cluster assignments. Assume a $K$-component mixture model as

Equation (2.73). To obtain a collapsed Gibbs sampler, $\pi$ and $\theta$ are marginalized:

$$p(c_j = k | c_{-j}, x, \alpha) \propto \frac{\frac{\alpha}{k} + N_k^{-j}}{\alpha + N - 1} p(x_j | \{x_i : i \neq j, c_i = k\}) \tag{2.81}$$

where $N_k^{-j}$ is the number of data currently assigned to cluster $k$ excluding the $i$th data point, and the predictive likelihood

$$p(x_j | \{x_i : i \neq j, c_i = k\}) \propto \int_{\Theta} h(\theta) p(x_j | \theta) \prod_{\{i \neq j : c_i = k\}} p(x_i | \theta) d\theta. \tag{2.82}$$

Computing the predictive likelihood Equation (2.82) is straightforward when $p$ and $h$ are conjugate priors, for instance, Gaussian cluster distribution result in a Student's t-distribution predictive distribution [42, 52, 77, 78].

### *2.5.6 Slice Sampling*

While MCMC methods presented in previous sections rely on a random walk process, slice sampling introduces an important class of Monte Carlo methods which is more model dependent. These MCMC methods may not adapt the local properties of the density function. On the contrary, slice sampling methods employ the local properties of densities to simulate [79].

Theorem 4 states that sampling from density $p(x)$ is equivalent to sampling uniformly on $\mathcal{E}(p) = \{(x, u) : 0 \leq u \leq p(x)\}$. One way to sample on $\mathcal{E}(p)$ is via random walk on the set. Neal showed that a natural way is to move towards one direction at a time, that is, moving along the $u$-axis which is the conditional distribution

$$u | x = x \sim \mathcal{U}nif(\{u : u \leq p(x)\}) \tag{2.83}$$

and then moving along $x$-axis which corresponds to the conditional distribution

$$x | u = u' \sim \mathcal{U}nif(\{x : u' \leq p(x)\}). \tag{2.84}$$

This procedure is depicted in Figure 2.13. Note that sampling from Equation (2.84)

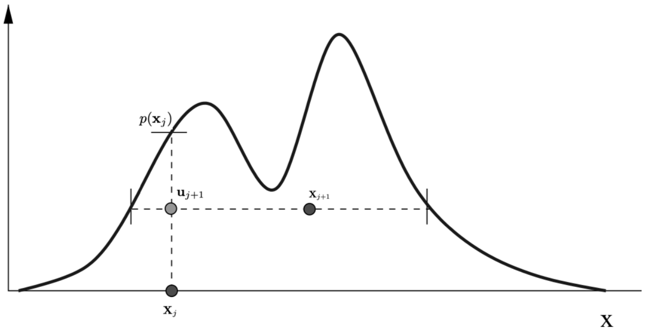


Figure 2.13: Slice Sampler.

can be intractable as $x$-dimension may grow. To address this issue, one can decompose the density $p(x)$ into some positive functions $p_j$, $j = 1, 2, \ldots, L$:

$$p(x) \propto \prod_{j=1}^{L} p_j(x) \tag{2.85}$$

and apply the aforementioned slice sampling method to each $p_j(x)$. This generalized algorithm is summarized in Algorithm 5.

**Algorithm 5:** Slice Sampling.

```
At iteration $k$
for $j = 1, \ldots, L$ do
    $u_j^k \sim \mathcal{U}nif\big([0, p_j(x^{k-1})]\big)$
end for
$x^j \sim \mathcal{U}nif\Big(\{x : u_j^k \leq p_j(x), \quad j = 1, 2, \ldots L\}\Big)$
Repeat till convergence
```

### *2.5.7 Variational Inference Methods*

One of the main problems in statistics, machine learning, and engineering is to approximate posterior probability densities in Bayesian models. These methods are an alternative to MCMC sampling methods which tend to converge faster for high-dimensional data [38].

Suppose $x = \{x_1, \dots x_N\}$ and $z = \{z_1, \dots z_M\}$ are the sets of observed variables and latent variables, respectively. We aim to estimate the conditional density $p(z|x)$ to do inference. The core idea behind the variational method is to select a variational family of distribution $\mathcal{D}$ over latent variables, $q(z|\nu) \in \mathcal{D}$ for some variational parameter $\nu$, and then optimize this distribution to be the closest to the posterior distribution $p(z|x)$ [63]. The best candidate, $q^*(z|\nu)$, is chosen in KL distance, i.e.,

$$q^*(z|\nu) = \underset{q(z|\nu)\in\mathcal{D}}{\arg\min}\, \mathrm{KL}(q(z|\nu), p(z|x)) \tag{2.86}$$

where $\mathrm{KL}(q(z|\nu), p(z|x))$ is written as (see Section 2.5.4)

$$\begin{aligned} \mathrm{KL}(q(z|\nu), p(z|x)) &= -\mathrm{ELBO}_q + \log p(x) \\ &= -\Big(\mathbb{E}_{q(z|\nu)}[\log p(x,z)] - \mathbb{E}_{q(z|\nu)}[q(z|\nu)]\Big) + \log p(x). \end{aligned} \tag{2.87}$$

The $\mathrm{ELBO}_q$ and $\log p(x)$ are the evidence lower bound and the log evidence, respectively. Note that the log evidence is a constant with respect to $q$, and hence optimizing KL distance is the equivalent to maximizing the ELBO. The advantage of maximizing the ELBO rather than minimizing the KL distance is that the ELBO can analytically be computed for a proper choice of $q$ where the evidence (or equivalently log evidence) cannot simply be computed. It is easy to verify that the EM algorithm discussed in Section 2.5.4 is a special case of variational Bayes method where in E-Step the ELBO is maximized.

### The Mean-Field Variational Family

There are various ways to choose the variational family of distribution $\mathcal{D}$ to approximate the posterior distribution. The mean-field variational family is a class of distributions that is useful for high-dimensional data.

One of main the choices for $q$ to make the posterior distribution tractable is to assume that the latent variables are independent, i.e.,

$$q(z|\nu) = \prod_{j=1}^{M} q(z_j|\nu). \tag{2.88}$$

The above variational family assumes a complete factorization of the distribution over each latent variable. Assuming independency, the ELBO can be re-written as

$$\mathrm{ELBO}_q = \sum_{j=1}^{M} \Big( \mathbb{E}_{q_j(z|\nu)}[\log p(z_j|z^{j-1}, x^n)] - \mathbb{E}_{q_j(z|\nu)}[q(z_j|\nu)] \Big). \tag{2.89}$$

Given this family of distributions, we can employ the coordinate ascent optimization method to maximize the ELBO. However, the coordinate ascent optimization may not converge to the local maxima since the convexity of ELBO is not guaranteed. Using the Lagrange multiplier method, the coordinate ascent update of $q(z_j|\nu)$ is given by

$$q^*(z_j|\nu) \propto \exp \mathbb{E}_{q_j}[\log p(z_j, z_{-j}, x)]. \tag{2.90}$$

Chapter 3

# DEPENDENT DIRICHLET PROCESS MODELING AND IDENTITY LEARNING FOR MULTIPLE OBJECT TRACKING

The goal of any multi-object tracking model is to (A) successfully estimate the trajectory of each object given the observation and (B) learn the number of the objects at each time step. Given the state vector configurations at the previous time step and current time observations, we propose nonparametric algorithms to satisfy (A), (B) [80, 81]. In this chapter, we develop probabilistic methods for estimating the trajectory of each object as well as learning the object cardinality using received measurements. We adapt the Bayesian nonparametric models introduced in Chapter 2 to accomplish the aforementioned tasks.

To fully develop a graphical model describing the multiple object tracking, we need to take the following processes into account upon which we model this problem: (a) Survival and transition; (b) Death; and (c) Birth. We develop algorithms that infer object identity and accurately track each object using a measurement set that is collected by sensors. In Section 3.1, we formulate the model constraints and conditions under which the dependency among the objects is captured. In Section 3.2, we construct a class of nonparametric time-dependent prior on the object state parameters. Section 3.3 discusses the inference methods based on the received measurements and constructed prior. A Markov chain Monte Carlo (MAMC) sampling method is proposed and conditions under which the convergence is guaranteed is provided. Section 3.4 discusses the consistency and contraction rate of the proposed methods and show that the contraction rate matches the optimal frequentist rate (minimax rate). We conclude in Section 3.5 with simulations demonstrating the performance of introduced methods and compare their performance in multi-target tracking application. Portions of these results were presented at the Asilomar conference on Bayesian inference [80, 82] and at the IEEE Transaction on Signal Processing [83].

## 3.1 Problem Formulation

We consider the problem of multiple object tracking with time-varying number of objects remaining, entering, and/or leaving the field of view (FOV). Let the time-dependent object and measurement cardinality at time step $k$ be $N_k$ and $M_k$, respectively. We define $\mathbf{X}_k = \{\mathbf{x}_{1,k}, \dots, \mathbf{x}_{N_k,k}\}$ and $\mathbf{Z}_k = \{\mathbf{z}_{1,k}, \dots, \mathbf{z}_{M_k,k}\}$ to be the collection of object state vectors taking values in state space $\mathcal{X}^{N_k}$ and the set of observations taking values in observation space $\mathcal{Z}^{M_k}$ at time $k$, respectively. Assume space $\mathcal{X}$ and $\mathcal{Z}$ are Polish spaces. Note that the time-dependent object cardinality $N_k$ is unknown and we aim to learn this unknown upon receiving the measurements at each time step. Given a state vector at time $(k-1)$, three possible situations may occur at time $k$:

(a) **Survival and Transition**: the object remains in the FOV with probability $\mathrm{P}_{\cdot,k|k-1}$ and its state transitions to the time step $k$ according to the transition probability kernel $\mathbb{Q}_{\boldsymbol{\theta}_{\cdot,k}}(\mathbf{x}_{\cdot,k-1}, \cdot)$ for unknown parameters $\underline{\theta}$.

(b) **Death**: the object leaves the FOV with probability with probability $1-\mathrm{P}_{\cdot,k|k-1}$.

(c) **Birth**: new object enters the scene.

We assume each measurement is generated by only one object and also the measurements are independent of one another. An object with state vector $\mathbf{x}_{\ell,k} \in \mathbf{X}_k$ generates an observations $\mathbf{z}_{l,k} \in \mathbf{Z}_k$ corresponding to the likelihood $p(\mathbf{z}_{l,k}|\mathbf{x}_{\ell,k})$. We employ Bayesian nonparametric methods to model uncertainties in the multiple object tracking. The nonparametric models are versatile tools to model a prior, however it cannot capture evolution over a period of time. Therefore, we need a more powerful tool to capture (a)-(c) over time. To model a collection of random distributions that are related but not identical, we define dependent nonparametric models to not only satisfies (a)-(c) but also captures time dependency. In what follows, we introduce a class of time-dependent nonparametric object-state prior models that conditioned on the process at time $(k-1)$ satisfies the following at time $k$:

A. **Survival**: Given the $\ell$th state at time $k-1$, $\mathbf{x}_{\ell,k-1}$, we define $\mathrm{P}_{\ell,k|k-1} : \Omega \to [0,1]$ to be the survival probability of state $\ell$ at time $k-1$.

B. **Transition**: Let $\nu : \Omega \times \mathcal{B} \to \mathbb{R}^+$ be the transition kernel. For each survived state with parameter $\theta^\star_{\ell,k-1}$ at time $(k-1)$, the parameter is evolved through $\theta_{\ell,k} \sim \nu(\theta^\star_{\ell,k-1}, \cdot)$. We refer to these parameters as cluster parameters.

C. **Trajectory**: Given the measurements, update the marginal (predictive) distribution.

Employing (A) - (C) provides nonparametric frameworks such that an object may perhaps disappear or remain and evolve over time. The evolution of the object throughout the time is recorded and is updated based on observing the measurements, and thus estimating the trajectory.

## 3.2 Evolutionary Time-Varying Prior Construction

In this section, we propose an evolutionary time-dependent model for multiple object tracking using dependent Dirichlet processes (DDP) to capture the full dependency among the objects. The marginal distribution of this dependent process is a Dirichlet process so that inference is simple and can efficiently be implemented. The proposed DDP evolutionary Markov modeling (DDP-EMM) approach can be used to learn multiple object clusters or labels over related information. The DDP-EMM algorithm is different from the random finite set (RFS) based algorithms for characterizing multiple object states and measurements [30, 84]. In particular, our approach directly incorporates learning multiple parameters through related information, including object labeling at the previous time step or labeling of previously considered objects at the same time step. In particular, the choice of the DDP as a prior on the object state distributions is based on the following dynamic dependencies in the state transition formulation: (I) the number of objects present at time step $k$ not only depends on the number of objects that were present at the previous time step $(k-1)$ but it also depends on the popularity of the objects at time $k$ (preferential attachment), (II) the clustering index of the parameter state of the $\ell$th object at time step $k$ depends on the clustering index of the parameter states of the previous $(\ell - 1)$ objects at the same time step $k$, and (III) a new object entering the scene is modeled without requiring any prior knowledge on the expected number of objects. We show that this process is exchangeable. In particular, the exchangeable partition probability function (EPPF) depends only on the size of the clusters. We may thus assume that the $\ell$th object is the last one to consider for clustering. The DDP-EMM algorithm is discussed next in detail and summarized in Algorithm 6. In particular, we provide: (i) the information available at time step $(k-1)$, (ii) how this information transitions from time step $(k-1)$ to time step $k$, and (iii) how the state transition stochastic model is constructed at time step k to form the multiple object state prior.

### Available Parameters at time $(k-1)$

The following set of parameters are assumed to be available at time step $(k-1)$:

- $\mathbf{X}_{k-1} = \{\mathbf{x}_{1,k-1}, \ldots, \mathbf{x}_{N_{k-1},k-1}\}$, where $\mathbf{x}_{\ell,k-1}$, $\ell$th object state vector, $\ell = 1, 2, \ldots, N_{k-1}$

- $\Theta_{k-1} = \{\boldsymbol{\theta}_{1,k-1}, \dots, \boldsymbol{\theta}_{N_{k-1},k-1}\}$, where $\boldsymbol{\theta}_{\ell,k-1}$, $\ell$th object-state cluster parameter vector associated with $\ell$th object

- $\Theta_{k-1} = \{\boldsymbol{\theta}_{1,k-1}, \dots, \boldsymbol{\theta}_{N_{k-1},k-1}\}$, collection of the cluster parameters

- $D_{k-1} = \#$ of unique DP clusters used as state prior

- $\Theta^\star_{k-1} = \{\boldsymbol{\theta}^\star_{1,k-1}, \dots, \boldsymbol{\theta}^\star_{D_{k-1},k-1}\}$, collection of the unique parameters such that $\Theta^\star_{k-1} \subseteq \Theta_{k-1}$

- $V^\star_{k-1}$ = vector of size $D_{k-1}$ where $\left[V^\star_{k-1}\right]_i$ is the number of objects in the $i$th cluster $i = 1, \dots, D_{k-1}$.

The induced *cluster assignment indicator sequence* at time $k-1$ is defined as

$$\mathcal{C}_{k-1} = \{c_{1,k-1}, \dots, c_{D_{k-1},k-1}\}, \tag{3.1}$$

where $c_{i,k-1} \in \{1, \dots, D_{k-1}\}$. Let $\mathcal{CA}_{k-1}$ be the collection of clustering assignment up to time $(k-1)$, i.e., $\mathcal{CA}_{k-1} = \{\mathcal{C}_1, \dots, \mathcal{C}_{k-1}\}$.

### Transitioning from time $(k-1)$ to time $k$

When transitioning from time step $(k-1)$ to time step $k$, it is assumed that the object with the state $\mathbf{x}_{\ell,k-1} \in \mathbf{X}_{k-1}$ may disappear from the FOV with probability $1 - \mathrm{P}_{k|k-1}$ or can stay in the scene with probability $\mathrm{P}_{k|k-1}$ and transition to a new state according to the transition kernel $\mathbb{Q}_{\boldsymbol{\theta}_{\ell,k}}(\mathbf{x}_{\ell,k-1}, \cdot)$. Let $\Theta^\star_{k|k-1}$ be the set of unique transitioned parameters to time step k. We assume if all the objects in a cluster leave the scene the corresponding cluster no longer exists. Therefore, the Bernoulli process associated with appearance/disappearance of the objects during transition from time $(k-1)$ to time $k$ is defined as:

$$\mathcal{B}_{k-1} = \{\mathrm{s}_{1,k|k-1}, \dots, \mathrm{s}_{N_{k-1},k|k-1}\} \tag{3.2}$$

where $\mathrm{s}_{\ell,k|k-1} \sim \mathrm{Ber}(\mathrm{P}_{\ell,k|k-1})$, where $\mathrm{Ber}(p)$ indicates a Bernoulli distribution with mean $p$. Note that $\mathrm{s}_{\ell,k|k-1} = 1$ indicates the survival of the $\ell$th object and transitioning to time $k$. Let the size vector $V^{\star}_{k|k-1}$ be the vector of size $D_{k-1}$ with entries indicating the size of each cluster after transitioning to time $k$. Note that some of the elements of the size vector may be zero. Since a cluster of size zero suggests that the cluster no longer exists, we may eliminate zeros in $V^{\star}_{k|k-1}$. We thus define the *cluster survival indicator* corresponding to nonempty clusters as

$$\mathcal{CS}_{k|k-1} = \{\lambda_{1,k|k-1}, \dots, \lambda_{D_{k-1},k|k-1}\} \tag{3.3}$$

where $\lambda_{j,k|k-1} \in \{0, 1\}$. Note that $\left[V^{\star}_{k|k-1}\right]_j = 0$ implies $\lambda_{j,k|k-1} = 0$ and if there is at least one object in the $j$th cluster, then $\lambda_{\ell,k|k-1} = 1$. Note that the number of non-zero clusters that transitions to time $k$ is $D_{k|k-1} = \sum_j \lambda_{j,k|k-1}$.

### DDP Prior Construction at time $k$

The DDP-EMM algorithm employs the parameters from time $(k-1)$ and the transition step to estimate the state distribution. Each cluster with $\lambda_{j,k|k-1} = 1$, $j \leq D_{k|k-1}$, a non-zero cluster, transitions to time $k$ according to the transition kernel $\nu(\boldsymbol{\theta}^{\star}_{j,k-1}, \cdot)$. Assume $\boldsymbol{\theta}_{j,k}$ is the $j$th cluster parameter at time $k$, we construct a dependent Dirichlet process as follow:

**Case 1:** The $\ell$th object is assigned to one of the survived and transitioned clusters from time $(k-1)$ which is occupied by at least one of the previous $\ell - 1$ previous objects. The survival of each object is determined by the survival indicator $s_{\cdot,k|k-1} \in \mathcal{B}_{k-1}$. Due to partition exchangeability, we may assume the $\ell$th object is the last one to cluster. The object selects one of these clusters with probability:

$$\Pi^1_{j,k}(\text{Select } j\text{th cluster}, j \leq D_{k-1}|\boldsymbol{\theta}^{\ell-1}_{1,k}) = \frac{[V_k]_j + \sum_{i=1}^{D_{k-1}} \left[V^\star_{k|k-1}\right]_i \lambda_{i,k|k-1}\delta_i(c_{j,k})}{g_{\ell-1,k-1}} \tag{3.4}$$

where $\boldsymbol{\theta}^{\ell-1}_{1,k} = \{\boldsymbol{\theta}_{1,k}, \ldots, \boldsymbol{\theta}_{\ell-1,k}\}$, $|\mathcal{A}|$ is the cardinality of set $\mathcal{A}$, and $\delta_i(\cdot)$ is the Dirac delta function, defined as $\delta_i(\mathcal{A})=1$ if $i\in\mathcal{A}$ and $\delta_i(\mathcal{A})=0$ if $i\notin\mathcal{A}$. The normalization term in Equation (3.4) is given by

$$g_{\ell-1,k-1} = (\ell-1) + \sum_{j}^{\ell-1} \sum_{i=1}^{D_{k-1}} \left[V^\star_{k|k-1}\right]_i \lambda_{i,k|k-1}\delta_i(c_{j,k}) + \alpha$$

where $\alpha>0$ is the concentration parameter.

Assume the space of states, $\mathcal{X}$, is Polish, given Equation (3.4) state distribution is drawn from as:

$$p(\mathbf{x}_{\ell,k}|\mathbf{x}_{1,k}, \ldots, \mathbf{x}_{\ell-1,k}, \mathbf{X}_{k|k-1}, \Theta^\star_{k|k-1}, \Theta_k) = \mathbb{Q}_{\boldsymbol{\theta}_{\ell,k}}(\mathbf{x}_{\ell,k-1}, \mathbf{x}_{\ell,k}) f(\mathbf{x}_{\ell,k}|\boldsymbol{\theta}_{\ell,k}) \tag{3.5}$$

for some density $f$ that describes the physical model.

**Case 2:** The $\ell$th object is assigned to one of the survived and transitioned clusters from time $(k-1)$. However, this cluster has not yet been assigned to any of the first $\ell-1$ objects. The object selects such a cluster with probability:

$$\Pi^2_{j,k}(\text{Select } j\text{th cluster that has not been selected yet}, j \leq D_{k-1}|\boldsymbol{\theta}^{\ell-1}_{1,k}) = \tag{3.6}$$
$$\frac{\sum_{i=1}^{D_{k-1}} \left[V^\star_{k|k-1}\right]_i \lambda_{i,k|k-1}\delta_i(c_{j,k})}{g_{\ell-1,k-1}}$$

where $g_{\ell-1,k-1}$ is defined as in Case 1. Note that $\mathbf{x}_{\ell,k-1}$ and $\boldsymbol{\theta}^\star_{\ell,l-1}$ transition to time $k$ using transition kernels $\mathbb{Q}_{\boldsymbol{\theta}_{\ell,k}}(\mathbf{x}_{\ell,k-1}, \cdot)$ and $\nu(\boldsymbol{\theta}^\star_{\ell,k-1}, \cdot)$, respectively.

Assuming the state space $\mathcal{X}$ is Polish and given Equation (3.6), the state distribution is:

$$p(\mathbf{x}_{\ell,k}|\mathbf{x}_{1,k},\ldots,\mathbf{x}_{\ell-1,k},\mathbf{X}_{k|k-1},\Theta^{\star}_{k|k-1},\Theta_k) = \\ \mathbb{Q}_{\boldsymbol{\theta}_{\ell,k}}(\mathbf{x}_{\ell,k-1},\mathbf{x}_{\ell,k})\nu(\boldsymbol{\theta}^{\star}_{\ell,k-1},\boldsymbol{\theta}_{\ell,k})f(\mathbf{x}_{\ell,k}|\boldsymbol{\theta}_{\ell,k}) \tag{3.7}$$

for some density $f$ that best describes the physical model.

**Case 3:** The object does not belong to any of the existing clusters; a new cluster parameter is drawn with probability:

$$\Pi^3_k(\text{Creating new cluster}|\boldsymbol{\theta}^{\ell-1}_{1,k}) = \frac{\alpha}{g_{\ell-1,k-1}} \tag{3.8}$$

The state distribution thus may be drawn as:

$$p(\mathbf{x}_{\ell,k}|\mathbf{x}_{1,k},\ldots,\mathbf{x}_{\ell-1,k},\mathbf{X}_{k|k-1},\Theta^{\star}_{k|k-1},\Theta_k) = \int_{\boldsymbol{\theta}} f(\mathbf{x}_{\ell,k}|\boldsymbol{\theta})dH(\boldsymbol{\theta}) \tag{3.9}$$

for some density $f$ according to the physical model and the base distribution $H$ on parameters. Algorithm 6 summarizes this process.

This model holds the following properties:

(**A**) This model allows for modification of both cluster location and dependent weights,

(**B**) This model ensures that the conditional distribution of DDP at time $k$ given the DDP at time $(k-1)$ is a Dirichlet process,

(**C**) This model records the labels since it is defined in the space of partitions,

(**D**) There exist a simple MCMC inference method to learn the trajectories based on this dependent statistical model.

Properties (**A**)-(**D**) are demonstrated in detail in the following theorems.

**Algorithm 6:** DDP-EMM: Time-Dependent Arrival and Survival Process

```
At time $(k-1)$
• $\mathbf{x}_{\ell,k-1}$: $\ell$th object state parameter vector, $\ell = 1, \ldots, N_{k-1}$
• $D_{k-1}$: # of unique DP clusters used as priors
• $V^{\star}_{k-1}$: vector of size $D_{k-1}$ where $\left[V^{\star}_{k-1}\right]_i$ is # of objects in $i$th cluster
• $\Theta^{\star}_{k-1} = \{\boldsymbol{\theta}^{\star}_{1,k-1}, \ldots, \boldsymbol{\theta}^{\star}_{D_{k-1},k-1}\}$: Cluster sequence of unique cluster parameters
• $\mathcal{B}_{k-1}$ : Bernoulli collection of appearance and disappearance association
• $\mathcal{C}_{k-1}$ : cluster assignment

Transitioning from time $(k-1)$ to $k$
Input: $\mathrm{P}_{\ell,k|k-1}$, transition kernel $\mathbb{Q}_{\boldsymbol{\theta}_{\ell,k}}(\mathbf{x}_{\ell,k-1}, \mathbf{x}_{\ell,k})$
Draw $\ell$th state survival indicator $\mathrm{s}_{\ell,k|k-1} \sim \mathrm{Ber}(\mathrm{P}_{\ell,k|k-1})$
If $\mathrm{s}_{\ell,k|k-1} = 1$, $\ell$th object survives w.p. $\mathrm{P}_{\ell,k|k-1}$ and transitions according to the
  transition kernel $\mathbf{x}_{\ell,k} \sim \mathbb{Q}_{\boldsymbol{\theta}_{\ell,k}}(\mathbf{x}_{\ell,k-1}, \mathbf{x}_{\ell,k})$
Form the object survival indicator set: $\mathcal{CS}_{k|k-1} = \{\mathrm{s}_{1,k|k-1}, \ldots, \mathrm{s}_{N_{k-1},k|k-1}\}$
• Compute the # of survived DP clusters after transitioning: $D_{k|k-1}$
• Form the size vector with entries $\left[V^{\star}_{k|k-1}\right]_j$, $j = 1, \ldots, D_{k|k-1}$
At time $k$
 Set $D_k = D_{k|k-1}$
  for $\ell = 1$ to $D_k$ do
    Set $[V_k]_\ell = \left[V^{\star}_{k|k-1}\right]_\ell$
    if $\ell \leq D_k$ and $\ell$th cluster already selected then
      Draw $\boldsymbol{\theta}_{\ell,k} \sim \nu(\boldsymbol{\theta}_{\ell,k-1}, \cdot)$ for cluster associated to $\ell$th object state w.p. $\Pi^1_{j,k}$
      Draw $\mathbf{x}_{\ell,k}|\boldsymbol{\theta}_{\ell,k}$ for $\ell$th object state from Equation (3.5)
    else if $\ell \leq D_k$ and $\ell$th cluster not yet selected then
      Draw $\boldsymbol{\theta}_{\ell,k} \sim \nu(\boldsymbol{\theta}_{\ell,k-1}, \cdot)$ for cluster associated to $\ell$th object state w.p. $\Pi^2_{\ell,k}$
      Draw $\mathbf{x}_{\ell,k}|\boldsymbol{\theta}_{\ell,k}$ for $\ell$th object state from Equation (3.7)
    else
      Draw $\boldsymbol{\theta}_{\ell,k} \sim H$ for new cluster associated to$\ell$th object state w.p. $\Pi^3_k$
      Draw $\mathbf{x}_{\ell,k}|\boldsymbol{\theta}_{\ell,k}$ for $\ell$th object state from Equation (3.9)
    end if
  end for
  return $\{\mathbf{x}_{1,k}, \mathbf{x}_{2,k}, \ldots, \ldots\}$, $\{\boldsymbol{\theta}_{1,k}, \boldsymbol{\theta}_{2,k}, \ldots, \ldots\}$
```

**Theorem 7.** Suppose that the space of state parameters is Polish. The dependent Dirichlet process in cases (1)-(3) define a Dirichlet process at each time step given the previous time configurations, i.e.,

$$\textit{DDP-EMM}_k|\textit{DDP-EMM}_{k-1} \sim DP\Big(\alpha, \sum_{\Theta_k} \Pi^1_{j,k}\delta_{\boldsymbol{\theta}_{\ell,k}} + \sum_{\Theta^\star_{k|k-1}\setminus\Theta_k} \Pi^2_{j,k}\nu(\boldsymbol{\theta}^\star_{\ell,k-1}, \boldsymbol{\theta}_{\ell,k})\delta_{\boldsymbol{\theta}_{\ell,k}} + \Pi^3_k H\Big). \tag{3.10}$$

*Proof.* To prove this theorem we need to prove (A1) The conditional distribution is a Dirichlet process, (A2) the base distribution is

$$\sum_{\Theta_k} \Pi^1_{j,k}\delta_{\boldsymbol{\theta}_{\ell,k}} + \sum_{\Theta^\star_{k|k-1}\setminus\Theta_k} \Pi^2_{j,k}\nu(\boldsymbol{\theta}^\star_{\ell,k-1}, \boldsymbol{\theta}_{\ell,k})\delta_{\boldsymbol{\theta}_{\ell,k}} + \Pi^3_k H.$$

**Propositions 4.** *(Lemma 3.2 [85])* Let $\mathcal{S}$ be a countable set or an open set in $\mathbb{R}^n$, and $F_\mathcal{S} \sim DDP$. Then, for every $s \in \mathcal{S}$, $F_s \sim DP$.

Proposition 4 guarantees that (A1) holds. Thus, it is sufficient to prove (A2). The base distribution in the Dirichlet process is the mean of the process and therefore is the distribution from which parameters are drawn. Case (1) implies that $\boldsymbol{\theta}_{\cdot,k}$ at time $k$ has degenerate distribution $\delta_{\boldsymbol{\theta}_{\ell,k}}$ for all $\ell$ that are survived and transitioned to time $k$. Case (2) implies that $\boldsymbol{\theta}_{\cdot,k}$ has the same distribution as one of the parameters at previous time step $(k-1)$ that has yet to transition to time $k$; hence, the distribution is proportional to the transition kernel, i.e., $\nu(\boldsymbol{\theta}^\star_{\ell,k-1}, \boldsymbol{\theta}_{\ell,k})\delta_{\boldsymbol{\theta}_{\ell,k}}$ for all $\ell$. Case (3) corresponds to $\boldsymbol{\theta}_{\cdot,k}$ drawn from the base distribution $H$. Probability of selecting each of these cases is given in Equation (3.4), Equation (3.6), and Equation (3.8), respectively. Note that it is straightforward to show these are exactly the features that Sethuraman uses to describe the Dirichlet process [48]. ■

Theorem 7 guarantees the conditional distribution given the configurations at previous time step and identifies the probability of choosing each parameter; thus,

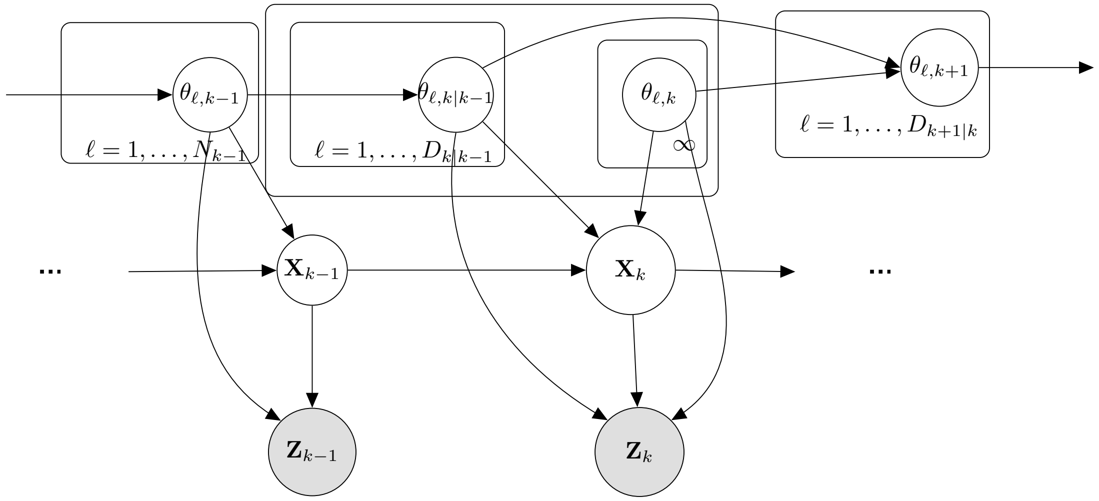


Figure 3.1: Graphical Model Capturing the Temporal Dependence, DDP-EMM Construction.

we can estimate the object density. The following theorem summarizes the density estimation:

**Theorem 8.** Assume that the space of object state parameters is separable and complete. Given the past configurations, the state distribution

$$p(\mathbf{x}_{\ell,k}|\mathbf{x}_{1,k}, \ldots, \mathbf{x}_{\ell-1,k}, \mathbf{X}_{k|k-1}, \Theta^{\star}_{k|k-1}, \Theta_k)$$

is given by

$$\begin{cases} \mathbb{Q}_{\underline{\boldsymbol{\theta}}}(\mathbf{x}_{\ell,k-1}, \mathbf{x}_{\ell,k}) f(\mathbf{x}_{\ell,k}|\boldsymbol{\theta}^{\star}_{\ell,k}) & \text{If case 1 happens} \\ \mathbb{Q}_{\underline{\boldsymbol{\theta}}}(\mathbf{x}_{\ell,k-1}, \mathbf{x}_{\ell,k}) \nu(\boldsymbol{\theta}^{\star}_{\ell,k-1}, \boldsymbol{\theta}_{\ell,k}) f(\mathbf{x}_{\ell,k}|\boldsymbol{\theta}^{\star}_{\ell,k}) & \text{If case 2 happens} \\ \int_{\boldsymbol{\theta}} f(\mathbf{x}_{\ell,k}|\boldsymbol{\theta}) dH(\boldsymbol{\theta}) & \text{If case 3 happens} \end{cases} \tag{3.11}$$

for some density $f$ that is based on the physical model.

*Proof.* The proof follows directly from the problem statement and Theorem 7. If case (1) happens: $\mathbf{x}_{\ell,k-1}$ transitions to time $k$ according to the probability transition kernel $\mathbb{Q}_{\boldsymbol{\theta}_{\ell,k}}(\mathbf{x}_{\ell,k-1},\cdot)$ and then is assigned to one of the existing clusters that is already used by one of the objects, and hence the corresponding density is $f(\mathbf{x}_{\ell,k}|\boldsymbol{\theta}_{\ell,k})$. If case (2): $\mathbf{x}_{\ell,k-1}$ and the cluster parameter $\boldsymbol{\theta}^{\star}_{\ell,k-1}$ transition to time $k$ according to transition kernels $\mathbb{Q}_{\boldsymbol{\theta}_{\ell,k}}(\mathbf{x}_{\ell,k-1},\cdot)$ and $\nu(\boldsymbol{\theta}^{\star}_{\ell,k-1},\cdot)$, respectively, and therefore the object is assigned to the this cluster. If case (3): new object does not belong to any of the previously assigned clusters, i.e., a new object emerges to the scene. In this case, we generate a new parameter from the base distribution $H$ and assign the object to the newly created cluster. ■

The graphical model describing the overall temporal dependence is depicted in Figure 3.1. In the next section, we discuss how we integrate this constructed prior on the states with the received measurements to learn the object cardinality and infer the predictive distribution to estimate the tracks.

## 3.3 Learning Model

The DDP-EMM, as discussed in Algorithm 6, provides a prior on the object state parameter distributions at time step $k$. We update our belief using the available measurement vectors at each time step, e.g., $\mathscr{Z}_k = \{\mathbf{z}_{l,k},\ l=1,\dots,M_k\}$ at time $k$. The posterior distribution is then used to estimate the trajectory of objects and learn the time-dependent object cardinality. It is assumed that each measurement is independent of each other and only generated from one object. Theorem 7 implies that we may exploit an infinite mixture model to estimate the density of the measurements and cluster them. Note that the measurement vectors are unordered meaning the $l$th measurement is not necessarily associated to the $\ell$th object state, $l \neq \ell$. As the DDP is used to label the object states at time step $k$, the infinite mixture model can be used to learn and assign a measurement to its associated object identity. In order

Algorithm 7: Infinite Mixture Model to Cluster and Track Objects

```
Input: Measurements: ${\mathbf{z}_{1,k}, \dots, \mathbf{z}_{M_k,k}}$
Output: $N_k$, cluster configurations, and posterior distributions

From construction of prior distribution
At time k
for $\ell = 1$ to $N_k$ do
    Sample $\{\boldsymbol{\theta}_{1,k}, \dots, \boldsymbol{\theta}_{N_k,k}\}$ and $\{\mathbf{x}_{1,k}, \dots, \mathbf{x}_{N_k,k}\}$ as in Algorithm 6
end for

for $l = 1$ to $L_k$ do
    Draw $\mathbf{z}_{l,k} | \mathbf{x}_{\ell,k}, \boldsymbol{\theta}_{\ell,k}$ from Equation (3.12)
end for

return $\mathcal{C}_k$ : induced cluster assignment indicators
Update: $\mathcal{CA}_k = \mathcal{CA}_{k-1} \cup \mathcal{C}_k$: set of cluster assignments up to time $k$
return $N_k$, $\mathcal{CA}_k$, and posterior of $\mathbf{z}_{l,k} | \mathbf{x}_{\ell,k}, \boldsymbol{\theta}_{\ell,k}$
```

to create the mixtures of distributions, we use the DDP-EMM prior in Algorithm 6. We utilize the generated DDP as a mixing distribution to compute the posterior distribution from the likelihood distribution $p(\mathbf{z}_{l,k} | \boldsymbol{\theta}_{\ell,k}, \mathbf{x}_{\ell,k})$ and update the object state estimates. In particular, $p(\mathbf{z}_{l,k} | \boldsymbol{\theta}_{\ell,k}, \mathbf{x}_{\ell,k})$ is drawn according to the following hierarchy:

$$\begin{aligned} &\boldsymbol{\theta}_{\ell,k} \sim \text{DDP-EMM}(\alpha, H) \\ &\mathbf{x}_{\ell,k} \mid \boldsymbol{\theta}_{\ell,k} \sim F(\boldsymbol{\theta}_{\ell,k}) \\ &\mathbf{z}_{l,k} | \boldsymbol{\theta}_{\ell,k}, \mathbf{x}_{\ell,k} \sim R(\mathbf{z}_{l,k} | \boldsymbol{\theta}_{\ell,k}, \mathbf{x}_{\ell,k}) \end{aligned} \tag{3.12}$$

where $F(\boldsymbol{\theta}_{\ell,k})$ is a distribution whose density follows Equation (3.5), Equation (3.7), Equation (3.9), and $R(\mathbf{z}_{l,k} | \boldsymbol{\theta}_{\ell,k}, \mathbf{x}_{\ell,k})$ is a distribution that depends on the measurement likelihood function. Algorithm 7 summarizes the infinite-dimension mixture model implementation to cluster the measurements and track the objects. Algorithms 6 and Algorithm 7, together with MCMC sampling methods, constitute the overall DDP-EMM multiple object tracking algorithm. In the next section, sampling algorithms are provided in detail.

### *3.3.1 Bayesian Inference: Gibbs Sampler*

Identifying the labels in tracking multiple objects and estimating the density parameters using DDP-EMM is a state-of-the-art method. However, computing the explicit posterior, and therefore the trajectory is impossible. The development of MCMC methods to sample form the posterior distribution has made this issue computationally feasible. The Gibbs sampler is an MCMC method to sample from a density, without directly requiring the density, by using the marginal distributions. The Gibbs sampler provides sample from the posterior distribution from the finite dimensional representation rather than sampling from infinite dimension representations where one can use slice sampling methods.

We outline the Gibbs sampler inference scheme for our model. We use a Gibbs sampling technique to iterate between sampling the state variables and the set of dynamic DDP parameters. We propose a method that can handle conjugate prior. This method can simply be generalized to a non-conjugate prior [52]. A key feature of this modeling is the discreetness of the DDP [85–87]. We outline this scheme next.

**Predictive Distribution:** The Bayesian posterior can be solved through the following:

$$P(\mathbf{x}_{\ell,k}|\mathcal{Z}_k) = \int_{\boldsymbol{\theta}} P(\mathbf{x}_{\ell,k}|\mathcal{Z}_k, \boldsymbol{\theta}) dG(\boldsymbol{\theta}|\mathcal{Z}_k) \tag{3.13}$$

where $G(\boldsymbol{\theta}|\mathcal{Z}_k)$ is the posterior distribution of the parameters given the observations. Note that we have $p(\mathbf{x}_{\ell,k}|\mathcal{Z}_k, \boldsymbol{\theta}) = p(\mathbf{x}_{\ell,k}|\boldsymbol{\theta})$, and hence can be evaluated as follows:

$$p(\mathbf{x}_{\ell,k}|\Theta) = \int p(\mathbf{x}_{\ell,k}|\boldsymbol{\theta}_{\ell,k}) d\pi(\boldsymbol{\theta}_{\ell,k}|\Theta) \tag{3.14}$$

where $\pi(\boldsymbol{\theta}_{\ell,k}|\Theta)$ is posterior distribution of $\boldsymbol{\theta}_{\ell,k}$ given the rest of parameters. The distribution of $\pi(\boldsymbol{\theta}_{\ell,k}|\Theta)$ is given by

$$\pi(\boldsymbol{\theta}_{\ell,k}|\Theta) = \sum_{\boldsymbol{\theta}\in\Theta_k-\{\boldsymbol{\theta}_{\ell,k}\}} \Pi^1_{j,k}\delta_{\boldsymbol{\theta}}(\boldsymbol{\theta}_{\ell,k}) + \sum_{\substack{\boldsymbol{\theta}\in\Theta^\star_{k|k-1}\setminus\Theta \\ \boldsymbol{\theta}\neq\boldsymbol{\theta}_{\ell,k}}} \Pi^2_{j,k}\nu(\boldsymbol{\theta}^\star_{\ell,k-1}, \boldsymbol{\theta}_{\ell,k})\delta_{\boldsymbol{\theta}}(\boldsymbol{\theta}_{\ell,k}) + \Pi^3_k H(\boldsymbol{\theta}_{\ell,k}). \tag{3.15}$$

To compute Equation (3.13), we need to calculate the parameter posterior given observations, $G(\boldsymbol{\theta}|\mathcal{Z}_k)$. However, direct computation of Equation (3.13) is extremely computationally expensive due to the complexity of $G(\boldsymbol{\theta}|\mathcal{Z}_k)$ [88]. Instead, we propose a Gibbs sampling approximation of this distribution. The following distribution is obtained by combining the prior with the likelihood in order to use for Gibbs sampling.

**Theorem 9.** *(Gibbs Sampler)* In the model Equation (3.12) the conditional posterior distribution is given by

$$\boldsymbol{\theta}_{\ell,k} \mid \boldsymbol{\theta}_{-\ell,k}, \mathcal{Z}_k \sim \sum_{j=1}^{|\mathcal{C}_k|} \zeta_{j,k}\; \delta_{\boldsymbol{\theta}_{j,k}}(\boldsymbol{\theta}_{\ell,k}) + \sum_{\substack{j=1 \\ j\notin\mathcal{C}_k}}^{D_{k|k-1}} \beta_{j,k}\; K_{j,k}(\boldsymbol{\theta}_{\ell,k}) + \gamma_{\ell,k}\, H_\ell(\boldsymbol{\theta}_{\ell,k}), \tag{3.16}$$

where $\boldsymbol{\theta}_{-\ell,k}$ by convention is the set $\{\boldsymbol{\theta}_{j,k},\ j \neq \ell\}$, where

$$\begin{aligned}
\zeta_{j,k} &= \frac{\left[V_k\right]_j + \sum_{i=1}^{D_{k|k-1}} \left[V^\star_{k|k-1}\right]_i \lambda_{i,k|k-1}\delta_i(c_{j,k})}{g_{\ell-1,k-1}} R(\mathbf{z}_{\ell,k}|\mathbf{x}_{j,k}, \boldsymbol{\theta}_{j,k}) \\
\beta_{j,k} &= \frac{\sum_{\substack{i=1 \\ i\notin\mathcal{C}_k}}^{D_{k|k-1}} \left[V^\star_{k|k-1}\right]_j \lambda_{j,k|k-1}}{g_{\ell-1,k-1}} \\
&\sum_{j=1}^{|\mathcal{C}_k|} \zeta_{j,k} + \sum_{\substack{j=1 \\ j\notin\mathcal{C}_k}}^{D_{k|k-1}} \beta_{j,k} + \gamma_{\ell,k} = 1
\end{aligned} \tag{3.17}$$

where $g_{\ell-1,k-1} = (\ell-1) + \sum_{i=1}^{D_{k|k-1}} \left[V^\star_{k|k-1}\right]_i \lambda_{i,k|k-1} + \alpha,\ \alpha{>}0$. Moreover, $K_{j,k} = R(\mathbf{z}_{\ell,k}|\mathbf{x}_{j,k}, \boldsymbol{\theta}_{j,k})$ and $dH_\ell(\boldsymbol{\theta}) \propto R(\mathbf{z}_{\ell,k}|\mathbf{x}_{j,k}, \boldsymbol{\theta})dH(\boldsymbol{\theta})$ where $H$ is the base distribution on $\boldsymbol{\theta}$.

*Proof.* The proof of Theorem 9 follows the standard Bayesian nonparametric methods. We know that the base measure in DP($\alpha$, H) is the mean of the Dirichlet prior. The following lemma generalizes this fact.

**Lemma 2.** *(Ferguson 1973, [47])* If $G \sim \mathrm{DP}(\alpha, H)$ and $g$ is any measurable function, then

$$\mathbb{E}\Big[\int g(\boldsymbol{\theta})dG(\theta)\Big] = \int g(\boldsymbol{\theta})dH(\boldsymbol{\theta})$$

Suppose that $A$ and $B$ are measurable sets, then

$$\begin{aligned} P(\boldsymbol{\theta}_{\ell,k} \in A, \mathbf{z}_{\ell,k} \in B|\boldsymbol{\theta}_{-\ell,k}, \mathbf{z}_{-\ell,k}) = \mathbb{E}\big[\mathbb{1}_{\boldsymbol{\theta}_{\ell,k}}(A)\mathbb{1}_{\mathbf{z}_{\ell,k}}(B)|\boldsymbol{\theta}_{-\ell,k}, \mathbf{z}_{-\ell,k}\big] \qquad (3.18)\\ =&\mathbb{E}\Big[\mathbb{E}\big[\mathbb{1}_{\boldsymbol{\theta}_{\ell,k}}(A)\mathbb{1}_{\mathbf{z}_{\ell,k}}(B)|G, \boldsymbol{\theta}_{-\ell,k}, \mathbf{z}_{-\ell,k}\big]|\boldsymbol{\theta}_{-\ell,k}, \mathbf{z}_{-\ell,k}\Big] \qquad (3.19)\\ =&\mathbb{E}\Big[\int \mathbb{1}_{\boldsymbol{\theta}_{\ell,k}}(A)\mathbb{1}_{\mathbf{z}_{\ell,k}}(B)p(\mathbf{z}_{\ell,k}|\boldsymbol{\theta}_{\ell,k}, \mathbf{x}_{\ell,k})d\mathbf{z}_{\ell,k}dG(\boldsymbol{\theta}_{\ell,k}|\boldsymbol{\theta}_{-\ell,k})\Big] \quad (3.20) \end{aligned}$$

where Equation (3.18) follows the definition of expected value, Equation (3.19) is due to the law of iterated expectations, and $G(\theta)$ in Equation (3.20) is the posterior dependent Dirichlet process given in Equation (3.15). Using Lemma 2

$$\begin{aligned} &\mathbb{E}\Big[\int \mathbb{1}_{\boldsymbol{\theta}_{\ell,k}}(A)\mathbb{1}_{\mathbf{z}_{\ell,k}}(B)p(\mathbf{z}_{\ell,k}|\boldsymbol{\theta}_{\ell,k}, \mathbf{x}_{\ell,k})d\mathbf{z}_{\ell,k}dG(\boldsymbol{\theta}_{\ell,k}|\boldsymbol{\theta}_{-\ell,k})\Big] =\\ &\int \mathbb{1}_{\boldsymbol{\theta}_{\ell,k}}(A)\mathbb{1}_{\mathbf{z}_{\ell,k}}(B)p(\mathbf{z}_{\ell,k}|\boldsymbol{\theta}_{\ell,k}, \mathbf{x}_{\ell,k})d\mathbf{z}_{\ell,k}\times \qquad (3.21)\\ &d\Big(\sum_{\Theta_k - \{\boldsymbol{\theta}_{\ell,k}\}} \Pi_1\delta_{\boldsymbol{\theta}}(\boldsymbol{\theta}_{\ell,k}) + \sum_{\substack{\boldsymbol{\theta}\in\Theta^\star_{k|k-1}\setminus\Theta\\ \boldsymbol{\theta}\neq\boldsymbol{\theta}_{\ell,k}}} \Pi_2\nu(\boldsymbol{\theta}^\star_{\ell,k-1}, \boldsymbol{\theta}_{\ell,k})\delta_{\boldsymbol{\theta}}(\boldsymbol{\theta}_{\ell,k}) + \Pi_3 H(\boldsymbol{\theta}_{\ell,k})\Big). \end{aligned}$$

Using the Bayes rule we have:

$$P(\boldsymbol{\theta}_{\ell,k} \in A|\boldsymbol{\theta}_{-\ell,k}, \mathcal{Z}_k) = \frac{\int_B P(\boldsymbol{\theta}_{\ell,k} \in A, \mathbf{z}_{\ell,k}|\boldsymbol{\theta}_{-\ell,k}, \mathbf{z}_{-\ell,k})d\mathbf{z}_{\ell,k}}{\int_\Omega P(\boldsymbol{\theta}_{\ell,k} \in A, \mathbf{z}_{\ell,k}|\boldsymbol{\theta}_{-\ell,k}, \mathbf{z}_{-\ell,k})d\mathbf{z}_{\ell,k}} \qquad (3.22)$$

and this concludes the claim in Theorem 9. ■

### *3.3.2 Convergence of Gibbs Sampler for DDP-EMM Prior*

There are many sets of conditional distributions that can be used as the basis of Gibbs sampler for which they violate the required posterior convergence conditions of the sampler. In this section, we discuss conditions under which the proposed Gibbs sampler in Section 3.3.1 converges to the posterior distribution.

We first prove that the regardless of initial condition the transition kernel converges to the posterior for almost all initial condition and then we provide the set of conditional distributions to guarantee the convergence to the posterior of the introduced Markov chain using Theorem 1 in [89]. To this end, let $\mathrm{K}(\boldsymbol{\theta}_0, \Theta)$ and $P_{\boldsymbol{\theta}}(\cdot|\mathcal{Z}_k)$ be the transition kernel for the Markov chain starting at $\theta_0$ and stopping in the set $\Theta$ after one iteration of the algorithm introduced in Section 3.3.1 and the posterior distribution of parameters given the observations at time $k$, respectively.

**Theorem 10.** At each time step $k$, convergence to the posterior distribution $P_\theta(\cdot|\mathcal{Z}_k)$ does not depend on the starting value, i.e.,

$$||\mathrm{K}_k^n(\boldsymbol{\theta}_0, \cdot) - P_{\boldsymbol{\theta}}(\cdot|\mathcal{Z}_k)||_{TV} \longrightarrow 0 \tag{3.23}$$

as $n \to \infty$, for almost all initial conditions $\boldsymbol{\theta}_0$ in total variation norm.

*Proof.* We first state the following postulate that will be used to prove this theorem.

**Postulate 1** (Theorem 1, Tierney 1994 [89])**.** Assume K is a $\pi$-irreducible and aperiodic Markov transition kernel such that $\pi\mathrm{K} = \pi$. Then K is positive recurrent and $\pi$ is the unique invariant distribution of K and for almost all $x$ we have:

$$||K^n(x, \cdot) - \pi||_{TV} \longrightarrow 0 \tag{3.24}$$

where $|| \cdot ||_{TV}$ is the total variation norm.

Therefore, to prove Theorem 10, we only need to check the conditions in Postulate 1. The proof of invariance of the posterior distribution for the Markov chain defined in Equation (3.16) is similar to the proof of theorem 2 of [90]. We only need to prove the aperiodicity and irreducibility of the Markov transition kernel with respect to the posterior distribution.

*Irreducibility:* Assume that $B^k_{\boldsymbol{\theta}} = \cup B^k_{j,\boldsymbol{\theta}}$ is a partition where the elements of this partition, $B^k_{j,\boldsymbol{\theta}}$, are the parameters configuration vector at time $k$ and $\pi_{j,k}(B^k_{j,\boldsymbol{\theta}})$ is the probability measure associated for a fixed configuration. Note that the distribution $\pi_k$ at time $k$ has a unique distribution $\pi_k = \sum \pi_{j,k}(B^k_{j,\boldsymbol{\theta}})$. Conditioning on a fixed configuration with $\pi_k(B^k_{j,\boldsymbol{\theta}}) > 0$, both posterior and predictive distributions depends on distributions where posterior and $\pi_k$ take to be mutually absolutely continuous with the transition kernel $\mathrm{K}(\boldsymbol{\theta}_0, B^k_{j,\boldsymbol{\theta}}) > 0$. The construction of transition kernel implies that for any $\boldsymbol{\theta}_0$ the transition kernel is positive, $\mathrm{K}(\boldsymbol{\theta}_0, B^k_{j,\boldsymbol{\theta}}) > 0$, therefore, $\mathrm{K}(\boldsymbol{\theta}_0, B^k_{\boldsymbol{\theta}}) > 0$ with respect to $\pi_k$. Note that the posterior and $\pi_k$ are mutually absolutely continuous hence one can conclude that $\mathrm{K}(\boldsymbol{\theta}_0, B^k_{\boldsymbol{\theta}}) > 0$ with respect to the posterior.

*Aperiodicity:* Note that for $B^k_{\boldsymbol{\theta}}$, we have $\pi_k(B^k_{\boldsymbol{\theta}}) > 0$ which directly implies the aperiodicity of the kernel. Therefore, the defined Markov chain sampler is irreducible, aperiodic, and invariant with respect to the posterior, hence, it satisfies the conditions in postulate 1. ■

Theorem 10 guarantees the convergence to the posterior for almost all initial values. This result specifically holds if normal distribution is considered [53, 90].

## 3.4 Properties of DDP-EMM

Given the configurations at time $(k-1)$, the infinite exchangeable random partition induced by $\mathcal{C}_k$ at time $k$ follows the exchangeable partition probability function (EPPF) [49]

$$p([V_k]^\star_1, \ldots, [V_k]^\star_{D_k}) = \frac{\alpha^{D_k}}{\alpha^{[N_k]}} \prod_{j=1}^{D_k} ([V_k]^\star_j - 1)! \tag{3.25}$$

where $D_k$ is the number of unique cluster parameter, $[V_k]^\star_j$, $j = 1, \ldots, D_k$ is the cardinality of the cluster $c_{j,k}$, and $\alpha^{[n]} = \alpha(\alpha+1)\ldots(\alpha+n-1)$. Note that number of the objects at time $k$, $N_k$, plays an important rule in partitioning. Also, due to variability of $N_k$ at time $k$, the relationship between partitions based on $(N_k - 1)$ and $N_k$ is important. The EPPF of the infinite random exchangeable partition based on the partition on $N_k$ and $(N_k - 1)$ objects given the configuration at time $(k-1)$ satisfies

$$\begin{aligned} & p_{N_k-1}([V_k]^\star_1, \ldots, [V_k]^\star_{D_k}) = \\ & \qquad \sum_{j=1}^{D_k} p_{N_k}([V_k]^\star_1, \ldots, [V_k]^\star_j + 1, \ldots [V_k]^\star_{D_k}) + p_{N_k}([V_k]^\star_1, \ldots, [V_k]^\star_{D_k}, 1). \end{aligned} \tag{3.26}$$

Equation (3.26) holds due to the Markov property of the process given the configuration at time $(k-1)$. Equation (3.26) entails a notion of consistency of the partitions in the distribution sense.

### *3.4.1 Consistency*

Suppose $\mathcal{Z}_k = \{\mathbf{z}_{1,k}, \ldots, \mathbf{z}_{M_k,k}\}$ is the collection of $M_k$ measurements at time $k$ with joint conditional distribution $R(\mathcal{Z}_k|\boldsymbol{\theta}, \mathbf{X}_k)$ with respect to the product probability space which is indexed by $\boldsymbol{\theta} \in \Theta$. The probability space $\Theta$ is assumed to be a first countable topological space[1]. Let $r_{\boldsymbol{\theta}}(\mathcal{Z}_k|\mathbf{X}_k)$ be the density corresponding to the probability measure $R(\mathcal{Z}_k|\boldsymbol{\theta}, \mathbf{X}_k)$.

**Definition**: The posterior distribution $P_{\boldsymbol{\theta}}(\cdot|\mathcal{Z}_k)$ is *weakly consistent* at true parameters $\boldsymbol{\theta}_0 \in \Theta$ at each time step $k$ if $P_{\boldsymbol{\theta}}(\mathbf{U}_k|\mathcal{Z}_k) \to 1$ in $r_{\boldsymbol{\theta}_0}(\mathcal{Z}_k|\mathbf{X}_k)$-probability as $n \to \infty$ for every neighborhood $\mathbf{U}_k$ of true parameters $\theta_0$.

**Definition**: The posterior distribution $P_{\boldsymbol{\theta}}(\cdot|\mathcal{Z}_k)$ is *strongly consistent* at true parameters $\boldsymbol{\theta}_0 \in \Theta$, if the convergence is almost sure.

[1] A space $\Theta$ is first-countable if each point has a countable neighborhood basis.

**Posterior Consistency of the Model**

In Section 3.2, we introduce a general model such that the distribution over the parameters at time $k$ conditioned on the configurations at time $(k-1)$ is a Dirichlet process. Schwartz [91] and Ghosal, et.al. [92] discussed the weak and strong consistency of the posterior distribution for a general kernel under a DP prior. In this section, we prove the consistency of the posterior distribution under DDP-EMM prior. The main result on weak consistency is due to Schwartz theorem. Let $r_{\boldsymbol{\theta}_0}$ be the true density of observations with corresponding probability measure $R_{\boldsymbol{\theta}_0}$,

**Propositions 5** (Schwartz 1965)**.** If $r_{\boldsymbol{\theta}_0}$ is in the KL support of the prior distribution $P_k$ on the topological space of all parameters with an appropriate $\sigma$-field, $r_{\theta_0} \in KL(\epsilon, P_k)$, then posterior distribution $P_{\boldsymbol{\theta}}(\cdot|\mathcal{Z}_k)$ is weakly consistent at $r_{\boldsymbol{\theta}_0}$.

The following theorem hence guarantees the consistency of the posterior at time $k$ under the proposed prior distribution introduced in Equation (3.10).

**Theorem 11.** *Let the true density be $r_{\boldsymbol{\theta}_0}$ and $P_k$ be the prior distribution at time $k$ conditioned on the configurations at time $(k-1)$ given by Equation* (3.10)*, if $r_{\boldsymbol{\theta}_0}$ is in the support of $P_k$, then $P_k(KL(\epsilon, r_{\boldsymbol{\theta}_0})) > 0$ and therefore, the posterior is weakly consistent.*

Proof of this theorem is straightforward and aligns with the proof in [92]. Intuitively speaking, one can prove this theorem by drawing an arbitrary measure from the base and show that the condition in the theorem holds for the set $KL(\epsilon, r_{\boldsymbol{\theta}_0})$. It is worth mentioning, $P_k(KL(\epsilon, r_{\boldsymbol{\theta}_0})) > 0$ is not a tight condition and holds true for many nonparametric models. In particular, in the case of Gaussian kernel, this condition is satisfied and hence the posterior is consistent using Gaussian kernels (Theorem 3, [92]).

**Remark:** Note that $r_{\boldsymbol{\theta}_0}$ being in the support of $P_k$ is equivalent to support$(r_{\boldsymbol{\theta}_0}) \subset$ support$\big(\sum_{\Theta_k} \Pi^1_{j,k}\delta_{\boldsymbol{\theta}_{\ell,k}} + \sum_{\Theta^\star_{k|k-1}\setminus\Theta_k} \Pi^2_{j,k}\nu(\boldsymbol{\theta}^\star_{\ell,k-1}, \boldsymbol{\theta}_{\ell,k})\delta_{\boldsymbol{\theta}_{\ell,k}} + \Pi^3_k H\big)$, provided $\Pi^1_{j,k}$, $\Pi^2_{j,k}$, and $\Pi^3_k$ as in Equation (3.10).

**Remark:** The posterior is also strongly consistent due to Theorem 1 of [93].

### *3.4.2 Posterior Contraction Rate of the Model*

Posterior contraction rate discusses how fast the posterior distribution approaches the true parameters from which the observations are generated. The contraction rate is highly related to posterior consistency.

**Definition**: A sequence $\epsilon_n$ is posterior contraction rate at the parameter $\boldsymbol{\theta}_0$ with respect to a metric $d$ if for every sequence $C_n \to \infty$, we have $P_{\boldsymbol{\theta}}(\boldsymbol{\theta} : d(\boldsymbol{\theta}, \boldsymbol{\theta}_0) \geq C_n\epsilon_n | \mathcal{Z}_k) \to 0$ in $P_{\boldsymbol{\theta}_0}$-probability as $n \to \infty$.

The following theorem specifies the contraction rate of the posterior contraction of the DDP based model introduced in Section 3.2. Assume that each $\mathbf{z}_{j,k} \in \mathbb{R}^{n_z}$, $j = 1, \ldots, M_k$. We denote $N_{[]}(\epsilon, \mathcal{H}_\kappa([0,1]^{n_z}), d)$ to be the $\epsilon$-bracketing number of Holder space $\mathcal{H}_\kappa$ with $\kappa$ degree of smoothness on the compact space of $[0,1]^{n_z}$ with respect to the distance $d$.

**Theorem 12.** Suppose $\mathcal{P}$ is the set of all distributions where the square root of the density belongs to the Holder space $\mathcal{H}_\kappa([0,1]^{n_z})$. Let $\epsilon_n$ be a decreasing sequence such that $\log N_{[]}(\epsilon, \mathcal{P}, d_H) \leq n\epsilon_n^2$ and $n\epsilon_n^2/\log n \to 0$, where $d_H$ is Hellinger distance[2]. Then, the posterior distribution at time $k$ of the DDP-EMM prior given $\mathcal{Z}_k$ and the previous time $(k-1)$ configurations converges to the true density at the rate of $\epsilon_n$, where $\epsilon_n$ is the order of $n^{-\frac{\kappa}{2\kappa+n_z}}$.

**Remark:** Note that the rate in Theorem 12 matches the minimax rate for density

[2] $d_H(p,q) = (\int(\sqrt{p} - \sqrt{q})^2 d\mu)^{\frac{1}{2}}$ is the Hellinger distance given the dominating measure $\mu$.

estimators. Hence, the DDP-EMM prior constructed through this model achieves the optimal frequentist rate.

*Proof.* Ghosal et.al prove that $\epsilon_n$ satisfying the conditions in the theorem is indeed the contraction rate [94]. Define $N(\epsilon, \mathcal{H}_\kappa([0,1]^{n_z}), ||\cdot||_\infty)$ to be the $\epsilon$-covering number of $\mathcal{H}_\kappa([0,1]^{n_z})$ with respect to supremum norm. Since one can find the $[l,u]$ bracket from the uniform approximation, the bracketing number with Hellinger distance grows with the same rate as the $\epsilon$-covering number with supremum norm. Therefore, it is enough to find an upper bound for $N(\epsilon, \mathcal{H}_\kappa([0,1]^{n_z}), ||\cdot||_\infty)$.

**Lemma 3** (Kolmogorov, Tihomirov 1961[95, 96])**.** For $[0,1]^{n_z} \subset \mathbb{R}^{n_z}$, there exist Constants $C$ depending on $\kappa$ and $n_z$ such that for every $\epsilon > 0$, we have

$$\log N(\epsilon, \mathcal{H}_\kappa([0,1]^{n_z}), ||\cdot||_\infty) \leq C\big(\frac{1}{\epsilon}\big)^{\frac{n_z}{\kappa}} \tag{3.27}$$

Lemma 3 implies that $\log N_{[]}(\epsilon, \mathcal{P}, d_H) \leq C\big(\frac{1}{\epsilon}\big)^{\frac{n_z}{\kappa}}$ and thus the convergence rate is the order of $n^{-\frac{\kappa}{2\kappa+n_z}}$. ■

## 3.5 Simulations

We now examine the empirical performance of the Bayesian nonparametric DDP-EMM tracker through various examples under different environmental conditions. Section 3.5.1 compares DDP-EMM tracker to labeled multi Bernoulli tracker and displays the error through the consistent OSPA metric [97]. In Section 3.5.2 and Section 3.5.3, we model a real scenario of moving cars and show that our tracker can outperform existing methods. Our results indicate that DDP-EMM can perform well in situations that other methods fail. For example, the DDP-EMM modeling of multiple object tracking improves the tracking and cardinality estimation performance in low signal-to-noise ratio (SNR) scenarios.

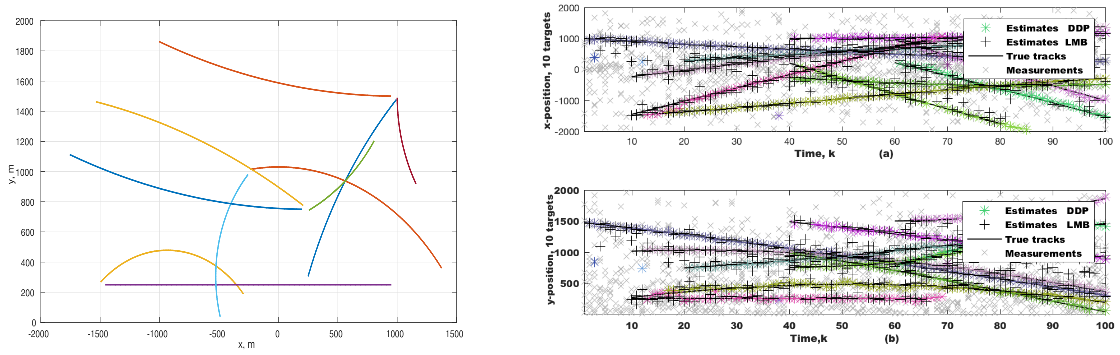
### *3.5.1 Comparison to Multi-Bernoulli Filtering*

The performance of the DDP-EMM model is demonstrated and compared to the labeled multi-Bernoulli filter (LMB) for a radar target tracking simulation example. The time-dependent number of targets are assumed to move according to the coordinated turn motion model. We assume there is a maximum number of ten targets. To perform a fair comparison, we used the same example as used for LMB in [30]. The unknown state parameters of the $\ell$th target at time $k$ are the Cartesian coordinates of the 2-dimensional (2-D) position $[x_{\ell,k}\ y_{\ell,k}]^T$, target velocity $[\dot{x}_{\ell,k}\ \dot{y}_{\ell,k}]^T$ and target turn rate $\omega_{\ell,k}$. The $\ell$th state vector is given by $\mathbf{x}_{\ell,k} = [x_{\ell,k}\ y_{\ell,k}\ \dot{x}_{\ell,k}\ \dot{y}_{\ell,k}\ \omega_{\ell,k}]^T$, $\ell = 1, \ldots, N_k$, where $N_k$ is the time-dependent target cardinality. The actual time-dependent trajectories are shown in Figure 3.2a. The transition probability density $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ for the coordinated turn motion model is assumed to be a Gaussian distribution with mean vector $\boldsymbol{\mu} = [\boldsymbol{\zeta}^T\ \omega_{k-1}]^T$ where $\boldsymbol{\zeta} = A_{\omega_{k-1}}\mathbf{x}_{k-1}$ and covariance matrix $Q = \text{diag}([\sigma_w^2 BB^T, \sigma_u^2])$ where $\sigma_w = 15$ m/s$^2$, $\sigma_u = \pi/180$ radians/s, and

$$A_{\omega_{k-1}} = \begin{bmatrix} 1 & \frac{\sin(\omega_{k-1})}{\omega_{k-1}} & 0 & -\frac{1-\cos(\omega_{k-1})}{\omega_{k-1}} \\ 0 & \cos(\omega_{k-1}) & 0 & -\sin(\omega_{k-1}) \\ 0 & \frac{1-\cos(\omega_{k-1})}{\omega_{k-1}} & 1 & \frac{\sin(\omega_{k-1})}{\omega_{k-1}} \\ 0 & \sin(\omega_{k-1}) & 0 & \cos(\omega_{k-1}) \end{bmatrix}, B = \begin{bmatrix} \frac{1}{2} & 0 \\ 1 & 0 \\ 0 & \frac{1}{2} \\ 0 & 1 \end{bmatrix}. \tag{3.28}$$

We select the probability of a target remaining at a scene during transitioning to be $\mathrm{P}_{\ell,k|k-1} = 0.95$, for all $\ell$. The times each target enters and leaves the scene are summarized in Table 3.1.

The measurement vector $\mathbf{z}_k = [\phi_k\ r_k]^T$ at time $k$ includes bearing $\phi_k$ and range $r_k$, where $r \in [0, 2,000]$ m and $\phi \in [-\frac{\pi}{2}, \frac{\pi}{2}]$. The measurement noise is assumed zero-mean Gaussian with variance $\sigma_r^2 = 25$ and $\sigma_\phi^2 = (\frac{\pi}{180})^2$. For the simulations, 10,000 Monte Carlo runs is used; the overall observed time steps is considered to be $K = 100$ and



(a) (b)

Figure 3.2: (a) Actual Target Trajectories. (b) Actual and Estimated $x$ (Top) and $y$ (Bottom) Position vs. Time $k$ Using DDP-EMM and LMB Methods.

Table 3.1: Target Existence over Time.

| Object | Presence | Object | Presence |
|---|---|---|---|
| Object 1 | $0 \leq k \leq 100$ | Object 6 | $40 \leq k \leq 100$ |
| Object 2 | $10 \leq k \leq 100$ | Object 7 | $40 \leq k \leq 100$ |
| Object 3 | $10 \leq k \leq 100$ | Object 8 | $40 \leq k \leq 80$ |
| Object 4 | $10 \leq k \leq 60$ | Object 9 | $60 \leq k \leq 100$ |
| Object 5 | $20 \leq k \leq 80$ | Object 10 | $60 \leq k \leq 100$ |

SNR = -3 dB. In our proposed model, we used a normal-inverse-Wishart distribution, $\mathcal{NIW}(\mu_0, \lambda, \nu, \Psi)$, with values $\mu_0 = 0.001, \lambda = 0, \nu = 50$, and an identity matrix for $\Psi$ as prior on the space of parameters. We consider a Gamma distribution as prior on the concentration parameter $\alpha$, $\Gamma(\alpha; 1, 0.1)$. Using the proposed DDP-EMM and inferential methods we estimate $x$ and $y$ coordinates. Figure 3.2b displays the actual and estimated target trajectories for the proposed DDP-EMM and the LMB methods in 10,000 Monte Carlo (MC) runs. Figure 3.2b shows that DDP-EMM has a higher estimation accuracy for the $x$ and $y$ coordinates in comparison with the LMB.

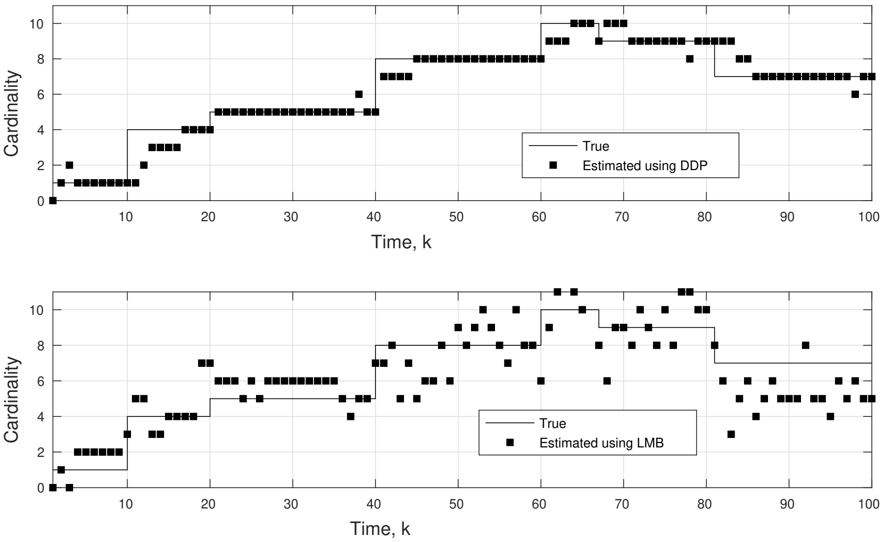


Figure 3.3: Comparison between Cardinality Estimation for DDP (Top) and LMB (Bottom) for Tracking 10 Objects.

in addition, Figure 3.3 shows that the DDP-EMM has higher accuracy than the LMB when estimating the time-dependent target cardinality. The increase in performance is also demonstrated through the optimal sub-pattern assignment (OSPA) metric (of order $p = 1$ and cut-off $c = 100$), for the range and the time-dependent object cardinality as in Figure 3.4. The OSPA location for both methods is compared in Figure 3.4 (top). Note that the lower the OSPA metric is, the higher the corresponding performance is. We observe that the DDP-EMM method often performs better than the LMB; this may be due to the fact that the LMB requires approximations when updating the target state estimates. The DDP-EMM and LMB can both track the targets. However, the DDP-EMM is computationally more efficient and has a higher tracking performance. As shown in Figure 3.3, the LMB drastically overestimates the cardinality of the 10 targets, when compared to the DDP-EMM; showing the elimination of the posterior cardinality bias. This is due to the fact that

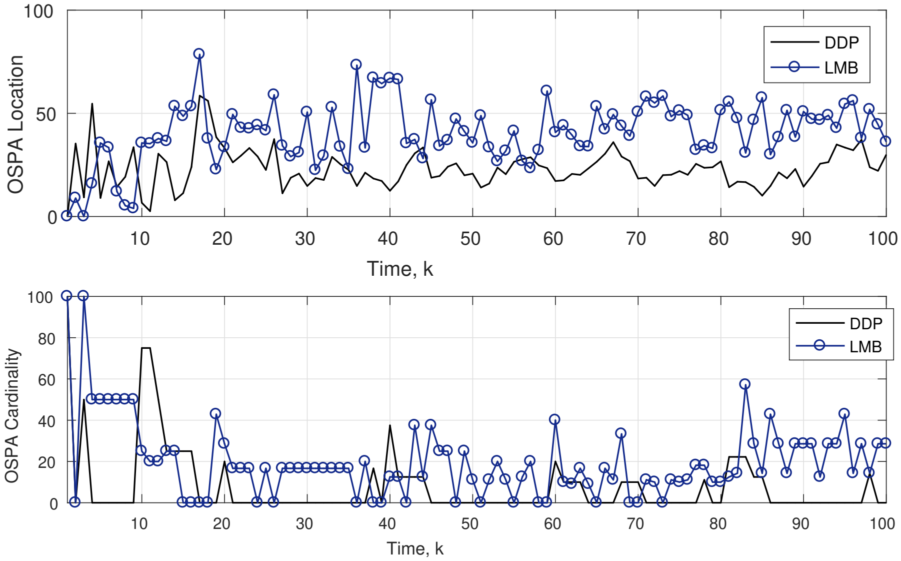


Figure 3.4: OSPA Location (Top) and Cardinality (Bottom) of Order $p = 1$ and Cut-off $c = 100$.

the LMB is highly sensitive to the presence of noise/clutter.

### *3.5.2 DDP-EMM and Low SNR: Moving Cars with Turn*

In this section, we show through simulations that DDP-EMM algorithm may accurately track objects in the presence of high noise and objects that are located very close to one another. We consider five moving cars where it is assumed that each car may enter, leave, or turn at any time. Each car comes to the scene at a different time and must follow the cars in front of it. The goal is to estimate the location/range of each car as well as the number of cars in the scene at each time step based on the noisy measurements received from the sensor.

The unknown state of each car is considered to be $[x, y, \dot{x}, \dot{y}, \omega]^T$ where $(x, y)$, $(\dot{x}, \dot{y})$, and $\omega$ are the location, velocity, and turning rate, respectively. The sensor only collects information about the range and angle at each time step. An additive

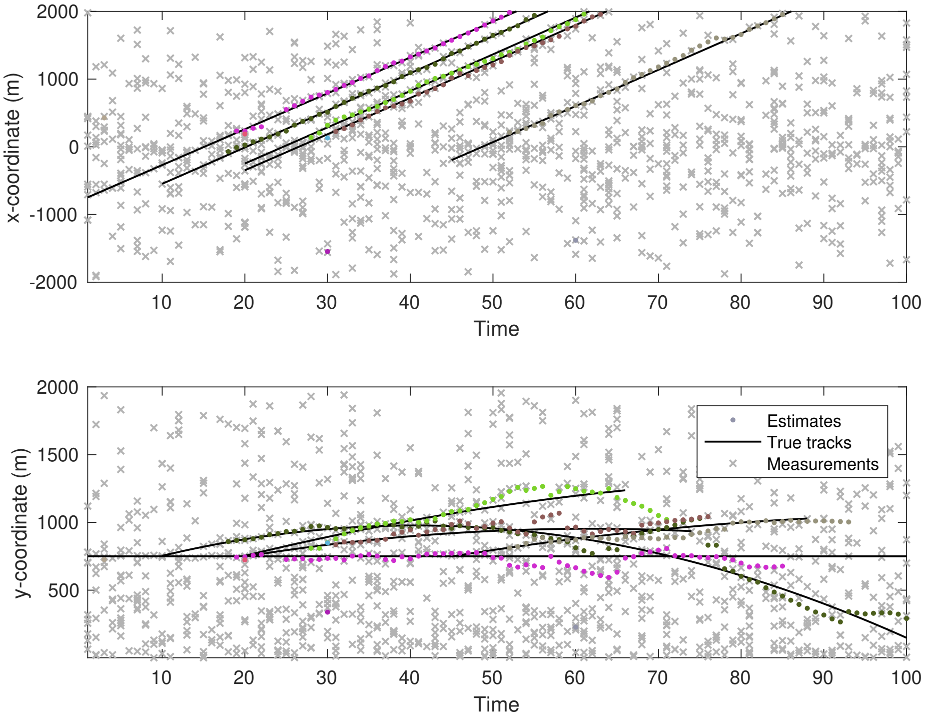


Figure 3.5: $x$-coordinate and $y$-coordinate Estimation Using DDP-EMM Model.

Gaussian noise is assumed throughout simulations. The SNR for this model is $-3$ dB.

In this scenario, the objects are assumed to be located near to one another which makes the model complicated to analyze. We compare the tracker introduced in this paper to the LMB tracker. We illustrate through simulations that DDP-EMM algorithm produces an accurate estimate of the location and cardinality despite high noise level and adjacency of objects. We assume a normal-inverse-Wishart distribution, $\mathcal{NIW}(\mu_0, \lambda, \nu, \Psi)$, with values $\mu_0 = 0.01, \lambda = 0, \nu = 100$, and an identity matrix for $\Psi$ as prior on the space of parameters. We consider a Gamma distribution as prior on the concentration parameter $\alpha$, $\Gamma(\alpha; 1, 0.3)$. Figure 3.5 and Figure 3.6 display the $x$-coordinate and $y$-coordinate estimation and the location of the objects using the DDP-EMM tracker, respectively. Running 10,000 Monte Carlo (MC) simulations,

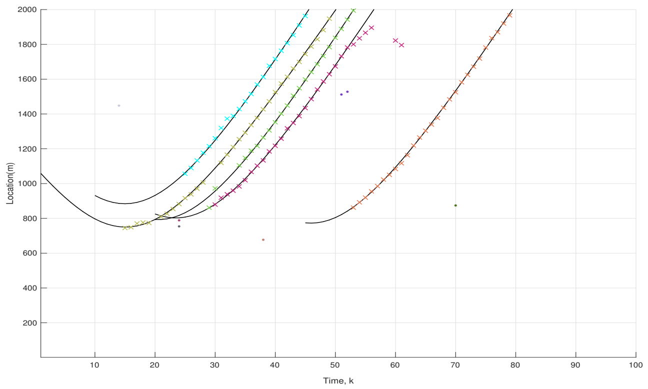


Figure 3.6: Location Estimation through DDP-EMM.

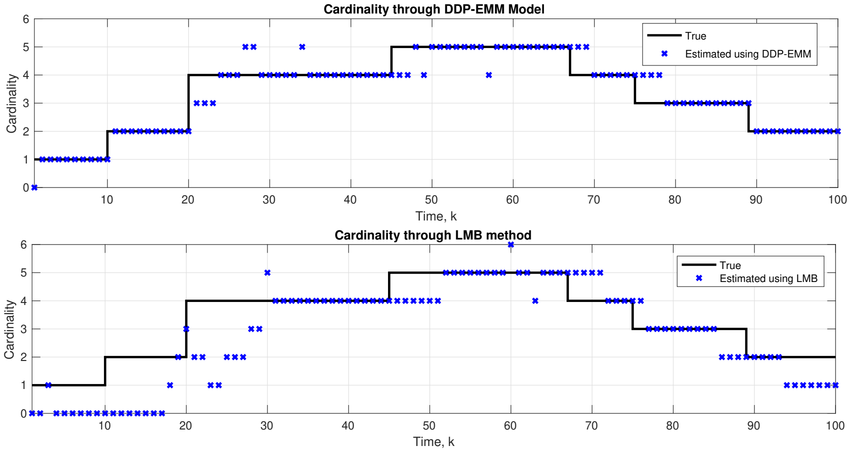


Figure 3.7: Cardinality Estimation via DDP-EMM and LMB.

the estimated cardinality and the OSPA metric for the location estimation error is depicted in Figure 3.7 and Figure 3.8, respectively. For OSPA metric, we set the order $p = 1$ and the cut-off $c = 100$. As shown in Figure 3.7 and Figure 3.8, under the same conditions, if the objects are located close to each other, the proposed DDP-EMM algorithm outperforms the LMB method and estimates the trajectory of each object more accurately.

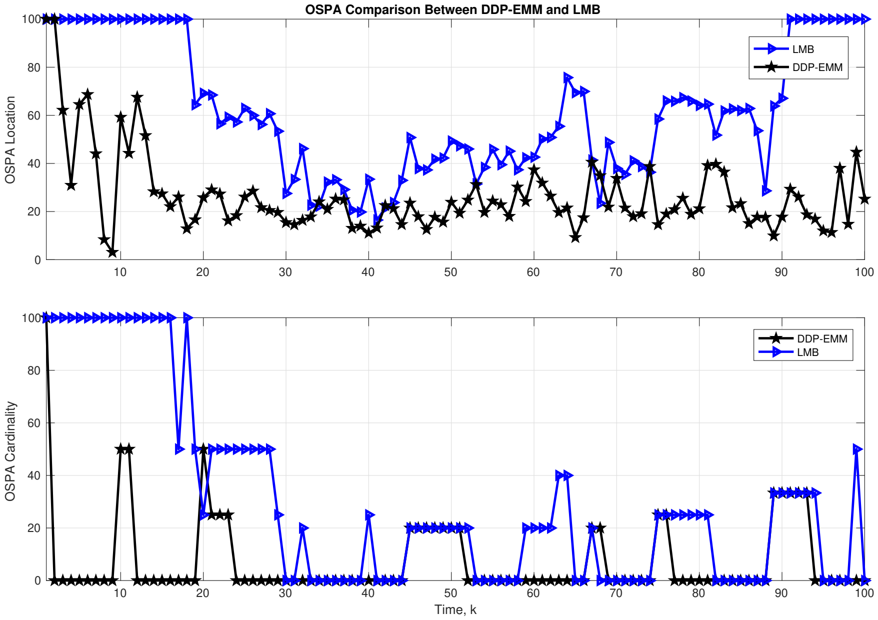


Figure 3.8: OSPA Comparison between DDP-EMM and LMB for Cut-off $c = 100$ and Order $p = 1$.

### *3.5.3 DDP-EMM under Different SNR Values*

We assume the same scenario as discussed in Section 3.5.2. However, in this example, we assume the turning rate is zero, i.e., $\omega = 0$. Thus, the unknown state vector is $[x, y, \dot{x}, \dot{y}]^T$. We put our proposed DDP-EMM method to the test under different SNR values. With the DDP-EMM prior, we model the state parameters as a realization of the proposed process. We assume Gaussian distributions throughout this simulation. Note that if we learn the states with zero mean, our model reduces to that of constant acceleration model and by assuming a non-zero mean we may consider faster changes. We simulate the algorithms for SNR $= -3$ dB, $-5$ dB, and $-10$ dB by place a normal-inverse-Wishart distribution, $\mathcal{NIW}(\mu_0, \lambda, \nu, \Psi)$, with values $\mu_0 = 0, \lambda = 0, \nu = 100$, and an identity matrix for $\Psi$ as prior on the space

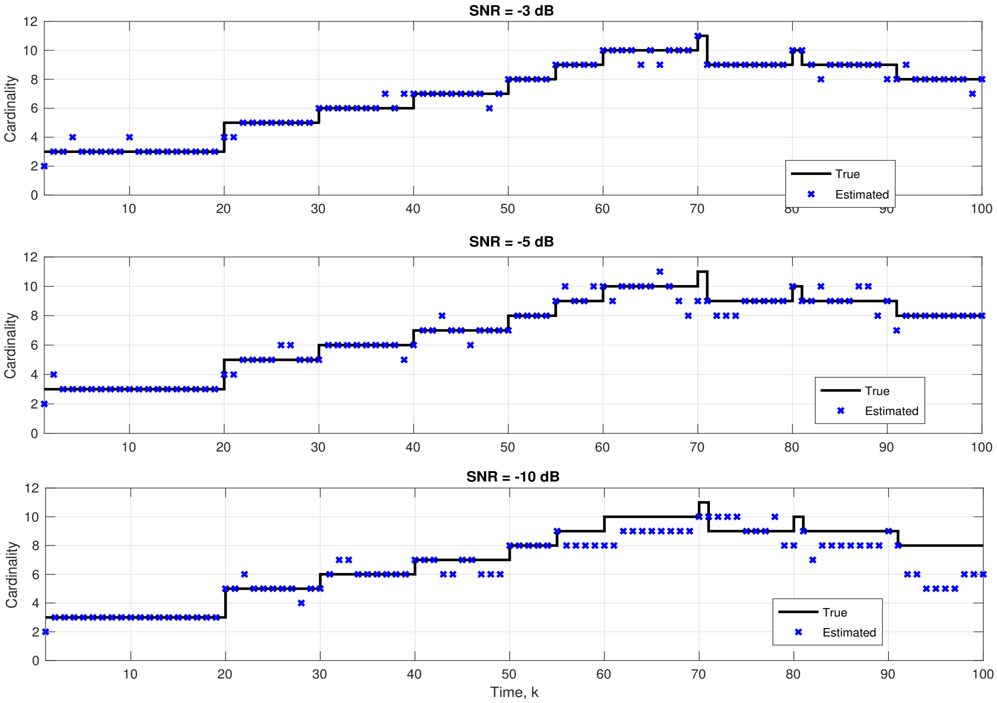


Figure 3.9: Cardinality Estimation for Different SNR Values.

of parameters. we also put the Gamma distribution $\Gamma(\alpha; 1, 0.2)$ as prior over the concentration parameter $\alpha$. Figure 3.9 presents the cardinality of the model under various SNR values for 10,000 MC simulation runs. As shown in Figure 3.9, the DDP-EMM method enables us to obtain the correct cardinality of the states most of the times even under high level of noise.

Figure 3.10 depicts the performance of this method under different SNR values. Note that for higher SNRs the OSPA metric is still fairly low which verifies the excellent performance of this method.

## 3.6 Discussion

Motivated by the success of Bayesian nonparametric methods in estimation and clustering, this chapter developed a class of nonparametric, sampling–based depen-

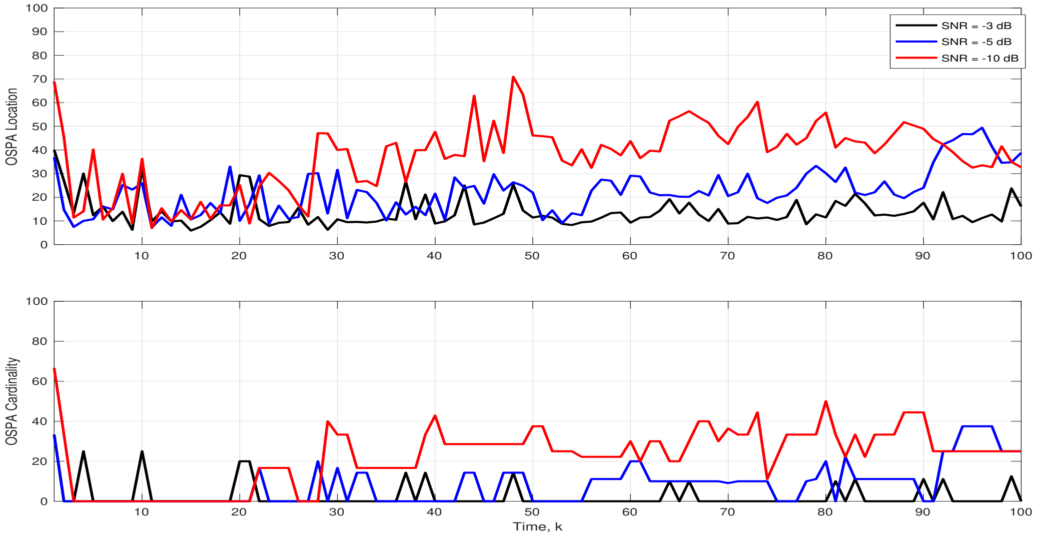


Figure 3.10: DDP-EMM Performance for SNR = −3 dB, SNR = −5 dB, and SNR = −10 dB.

dent Dirichlet process as a prior on the evolving state distributions in a multiple object tracking problem with time-dependent number of objects. Interestingly, we have shown that the proposed prior is consistent and the contraction rate matches the optimal frequentist minimax rate. We introduced a simple multi-scale sampling method to efficiently and accurately do inference using the DDP-EMM tracker. Chapter 4 revisits the problem of time-dependent multiple object tracking and develops models that directly incorporates learning multiple parameters from correlated information. We show that these models better suit the multiple object tracking with the time-varying number of objects due to their flexibility.

Chapter 4

# DEPENDENT PITMAN-YOR PROCESS FOR MODELING EVOLUTION IN MULTIPLE STATE PRIORS

In Chapter 3, we introduced the dependent Dirichlet process model to incorporate a learning algorithm as a prior over the time evolving object state distribution based on the measurements. When using the Dirichlet process to model the transitioning of objects into clusters, the expected number of unique clusters varies exponentially according to $\alpha log(N)$, where $\alpha$ is the concentration parameter and $N$ is the total number of objects to be clustered. A more flexible model is offered by the two-parameter Poisson-Dirichlet process, Pitman-Yor process, as, in this case, an additional discount parameter, $0 \leq d < 1$, with $\alpha > -d$, is used to control the number of clusters in the model [56, 98]. Specifically, as stated in Chapter 2, with the Pitman-Yor process model, the expected number of unique clusters varies according to the power-law $\alpha N^d$. Following the power-law, the higher the number of unique (non-empty) clusters, the higher the probability of having even more unique clusters. Also, clusters with only a small number of objects have a lower probability of having new objects. This more flexible model offered by the Pitman-Yor process is a better match for the tracking problem with a time-varying number of objects. With a maximum number of $N_k$ objects at time step $k$, an object may stay in the scene from the previous time step, leave the scene, or enter the scene for the first time. Thus, the object state would benefit from a larger number of available clusters to ensure all dependencies are captured.

In order to also capture time evolution, we introduce a family of dependent Pitman-Yor (DPY) processes that can be used to model a collection of random distributions that are related but not identical. As a result, we utilize the DPY to model the multiple object state prior distributions by directly incorporating learning multiple parameters from correlated information. The resulting DPY state transitioning prior (DPY-STP) method formulates the state transition such that the object cardinality at time step $k$ is dependent on its value at the previous time step $(k-1)$. Also, the index assigned to the cluster that contains an object state is dependent on the cluster indexing of the previously clustered object states at the same time step $k$. If a new object enters the scene, its state must be modeled without knowledge on the expected number of objects. We begin to address the problem of time-varying multi-object tracking in Section 4.1 by introducing a class of flexible consistent models through the dependent Pitman-Yor process as prior on the object state parameters. Section 4.2 describes an inference method to utilize the prior. We construct a flexible, robust, and accurate tracker by incorporating a learning method with the prior. In Section 4.3, we study the properties of the introduced methods. We also discuss conditions under which our model is consistent. Later results in Section 4.4 confirm that this method can significantly improve the model introduced in Chapter 3 and outperforms previously introduced methods [13, 30, 99]. The results are presented at the the 2019 22$^{\text{nd}}$ Information Fusion conference [81], and at the IEEE Transaction on Signal Processing [83].

## 4.1 DPY-STP Algorithm Construction for State Transitioning

In this section, we introduce an evolutionary time-dependent model for multiple object tracking based on our proposed dependent Pitman-Yor (DPY) process to learn object labels. The advantage of this model over the DDP-EMM method introduced in Chapter 3 is that this approach proposes a dependent Pitman-Yor (DPY) process that marginally preserves the Pitman-Yor process, and therefore it allocates higher probability to unique clusters. This observation makes DPY a better fit for multiple object tracking. In particular, our approach directly incorporates learning multiple parameters through related information, including object labeling at the previous time step or labeling of previously considered objects at the same time step. The choice of the DPY as a prior on the object state distributions is based on the following dynamic dependencies in the state transition formulation: (A) the number of objects present at time step $k$ relies on the number of objects that were present at the previous time step $(k-1)$, (B) the clustering index of the parameter state of the $\ell$th object at time step $k$ depends on the clustering index of the state parameters of the previous $(\ell-1)$ objects at the same time step $k$, and (C) new object entering the scene is modeled without requiring any prior knowledge on the expected number of objects. We propose the DPY-STP method to model the state transition process, accounting for multiple dependencies next in detail. This method is summarized in Algorithm 8. In particular, we provide: (a) the information available at time step $(k-1)$, (b) how this information transitions from time step $(k-1)$ to time step $k$, and (c) how the DPY-STP model is constructed at time step $k$ to estimate the object state density.

### Available Parameters at Time $(k-1)$

The DPY-STP algorithm assumes that the following parameters are available from previous time steps at time $(k-1)$:

- Let $\mathbf{X}_{k-1} = \{\mathbf{x}_{\ell,k-1} : \ell = 1, \ldots, N_{k-1}\}$ be the object states at time $(k-1)$.
- Let $\mathcal{CA}_{k-1} = \{\mathcal{C}_1, \ldots, \mathcal{C}_{k-1}\}$ be the cluster assignment up to time $(k-1)$, where $\mathcal{C}_J = \{c_{1,J}, \ldots, c_{N_J,J}\}$ is the cluster assignments at time step $J$.
- Define $\Theta_{k-1} = \{\boldsymbol{\theta}_{\ell,k-1} : \ell = 1, \ldots, N_{k-1}\}$ to be the set of object state parameters available at time $(k-1)$ associated with $\mathcal{C}_{N_{k-1}}$(note that $\boldsymbol{\theta}_\ell$'s are not necessarily

unique).

- Let $\Theta^{\star}_{k-1} = \{\boldsymbol{\theta}^{\star}_{\ell,k-1} : \ell = 1, \ldots, D_{k-1}\} \subset \Theta_{k-1}$ be the set of unique parameters, and $D_{k-1}$ be the number of uniques parameters.
- Define $\mathbf{V}^{\star}_{k-1}$ to be a vector of size $D_{k-1}$ containing the size of non empty clusters associated with $\mathcal{C}_{k-1}$. One can include empty clusters and define the size of this vector to be $N_{k-1}$. However, it is computationally more efficient to exclude size zero clusters.

### Parameters Transitioning from Time $(k-1)$ to Time $k$

Assume $s_{\ell,k|k-1}$ associate with the $\ell$th object at time $(k-1)$ has a Bernoulli distribution with parameter $P_{\ell,k|k-1}$, $s_{\ell,k|k-1} \sim \mathrm{Ber}(P_{\ell,k|k-1})$. Given $s_{\ell,k|k-1}$, the object $\mathbf{x}_{\ell,k-1}$ leaves the scene with probability $1 - P_{\ell,k|k-1}$ or remains in the field of view (FOV) with probability $P_{\ell,k|k-1}$ and transitions to a new state using the Markov transition kernel $\mathbb{Q}_{\boldsymbol{\theta}_{\ell,k}}(\mathbf{x}_{\ell}(k-1), \cdot)$. We assume if all the objects in a cluster (all object with the same parameter) leave the scene the cluster no longer exists. Let $\Theta^{\star}_{k|k-1}$ be the set of unique parameters at time $(k-1)$ that are transitioned to time step $k$. We define ${\mathbf{V}^{\star}}_{k|k-1}$ to be the vector of size of $D_{k-1}$ containing the size of each cluster after transitioning to time $k$. it is worth mentioning that a cluster with size zero implies that the cluster no longer exists. To keep track of the survived objects, let $\mathcal{CS}_{k|k-1}$ be the cluster survival indicator defined as

$$\mathcal{CS}_{k|k-1} = \{\eta_{1,k|k-1}, \ldots, \eta_{D_{k-1},k|k-1}\}$$

where $\eta_{j,k|k-1} = 0$ corresponds to disappearance of the $j$th cluster and $\eta_{j,k|k-1} = 1$ implies that there is at least one element in the $j$th cluster.

### DPY Prior Construction at Time $k$

Each survived cluster (a cluster with non-zero size after transitioning) is updated through a transition kernel. Assume that the cardinality of $\ell$th cluster at time $(k-1)$ is still non-zero after transitioning, then the $\ell$th object parameter will evolve according to the following transition kernel:

$$\boldsymbol{\theta}_{\ell,k} \sim \zeta(\boldsymbol{\theta}^{\star}_{\ell,k-1}, \cdot). \tag{4.1}$$

Let $\boldsymbol{\theta}_{\ell,k}$ be the transitioned $\ell$th state object parameter at time $k$, we construct the dependent Pitman-Yor prior as follows:

**Case1**: The $\ell$th object belongs to one of the survived and transitioned clusters from time $(k-1)$ and occupied at least by one of the previous $\ell-1$ objects. The object selects one of these clusters with probability:

$$\Gamma^{1}_{j,k}(\text{select } j\text{th cluster}|\boldsymbol{\theta}^{\ell-1}_{1,k}, \Theta_{k|k-1}) = \frac{\sum\limits_{i=1}^{D_{k-1}} \left[\mathbf{V}^{\star}_{k|k-1}\right]_i \eta_{i,k|k-1}\delta_i(c_{j,k}) + [\mathbf{V}_k]_j - d}{\sum\limits_{j=1}^{\ell-1}\sum\limits_{i=1}^{D_{k-1}} \left[\mathbf{V}^{\star}_{k|k-1}\right]_i \eta_{i,k|k-1}\delta_i(c_{j,k}) + \sum\limits_{j=1}^{\ell-1} [\mathbf{V}_k]_j + \alpha} \tag{4.2}$$

where $[\mathbf{V}_k]_j$ indicates the $j$th element of vector $\mathbf{V}_k$ at time $k$, $0 \leq d < 1$ and $\alpha > -d$ are the discount and strength parameters in the Pitman-Yor process, respectively.

**Case2**: The $\ell$th object belongs to one of the survived and transitioned clusters from time $(k-1)$ but this cluster has not yet been occupied by any one the first $\ell-1$objects. The object selects such a cluster with probability:

$$\Gamma^{2}_{j,k}(\text{Select } j\text{th cluster that has not been selected yet}|\boldsymbol{\theta}^{\ell-1}_{1,k}, \Theta_{k|k-1}) = \frac{\sum\limits_{i=1}^{D_{k-1}} \left[\mathbf{V}^{\star}_{k|k-1}\right]_i \eta_{i,k|k-1}\delta_i(c_{j,k}) - d}{\sum\limits_{j=1}^{\ell-1}\sum\limits_{i=1}^{D_{k-1}} \left[\mathbf{V}^{\star}_{k|k-1}\right]_i \eta_{i,k|k-1}\delta_i(c_{j,k}) + \sum\limits_{j=1}^{\ell-1} [\mathbf{V}_k]_j + \alpha} \tag{4.3}$$

**Case3**: The object does not belong to any of the existing clusters, thus a new cluster parameter is drawn from some base distribution $H$, corresponding to the base distribution in Pitman-Yor process, with probability:

$$\Gamma_k^3(\text{Create a new cluster}|\boldsymbol{\theta}_{1,k}^{\ell-1}, \Theta_{k|k-1}) = \frac{|D_k|_{\ell-1}d + \alpha}{\sum_{j=1}^{\ell-1}\sum_{i=1}^{D_{k-1}} \left[\mathbf{V}_{k|k-1}^{\star}\right]_i \eta_{i,k|k-1}\delta_i(c_{j,k}) + \sum_{j=1}^{\ell-1} [\mathbf{V}_k]_j + \alpha} \tag{4.4}$$

where $|D_k|_{\ell-1}$ is the total number of the clusters at time $k$ created by the first $(\ell-1)$ objects.

In above construction, $\Gamma_{j,k}^1, \Gamma_{j,k}^2,$ and $\Gamma_k^3$ are the probability of selecting an object cluster or creating a new object cluster. The temporal dependency among the objects follows a dependent Pitman-Yor process where the marginal distribution is a Pitman-Yor process. This property makes this process easy to implement since the marginal distribution becomes a Pitman-Yor process. The following theorem summarizes this property:

**Theorem 13.** Suppose that the space of state parameters is separable and complete metrizable space. The process defined by probabilities Equation (4.2), Equation (4.3), and Equation (4.4) defines a Pitman-Yor process at each time step given the previous time configurations, i.e.,

$$\textit{DPY-STP}_k|\textit{DPY-STP}_{k-1} \sim \mathcal{PY}\Big(d, \alpha, \sum_{\Theta_k} \Gamma_{j,k}^1 \delta_{\boldsymbol{\theta}_{\ell,k}} + \sum_{\Theta_{k|k-1}^{\star}\setminus\Theta_k} \Gamma_{j,k}^2 \zeta(\boldsymbol{\theta}_{\ell_{k-1}}^{\star}, \boldsymbol{\theta}_{\ell,k})\delta_{\boldsymbol{\theta}_{\ell,k}} + \Gamma_k^3 H\Big). \tag{4.5}$$

where $\delta_{\boldsymbol{\theta}}(\Theta) = 1$ if $\boldsymbol{\theta} \in \Theta$ and $\delta_{\boldsymbol{\theta}}(\Theta) = 0$, if $\boldsymbol{\theta} \notin \Theta$.

*Proof.* The proof of Theorem 13 is the direct result of cases (1)-(3). We eliminate the proof since it is analogous to the proof of Theorem 7. ■

Given the conditional distribution Equation (4.5), Theorem 14 provides an object density estimator.

**Theorem 14.** Assume the space of states, $\mathcal{X}$, is separable and complete metrizable topological space, given (Equation (4.2))-(Equation (4.4)) state distribution

$$p(\mathbf{x}_{\ell,k}|\mathbf{x}_{1,k},\ldots,\mathbf{x}_{\ell-1,k},\mathbf{X}_{k|k-1},\Theta^{\star}_{k|k-1},\Theta_k)$$

is estimated as follows:

$$\begin{cases} \mathbb{Q}_{\boldsymbol{\theta}_{\ell,k}}(\mathbf{x}_{\ell,k-1},\mathbf{x}_{\ell,k})f(\mathbf{x}_{\ell,k}|\boldsymbol{\theta}_{\ell,k}) & \text{If case 1 happens} \\ \mathbb{Q}_{\boldsymbol{\theta}_{\ell,k}}(\mathbf{x}_{\ell,k-1},\mathbf{x}_{\ell,k})\zeta(\boldsymbol{\theta}^{\star}_{\ell,k-1},\boldsymbol{\theta}^{\star}_{\ell,k})f(\mathbf{x}_{\ell,k}|\boldsymbol{\theta}_{\ell,k}) & \text{If case 2 happens} \\ \int_{\boldsymbol{\theta}} f(\mathbf{x}_{\ell,k}|\boldsymbol{\theta})dH(\boldsymbol{\theta}) & \text{If case 3 happens} \end{cases} \tag{4.6}$$

for some density $f(\cdot|\boldsymbol{\theta})$ that describes the physical model, base distribution $H$ on parameters, and $\mathbf{X}_{k|k-1}$ the set of survived state objects. Note that elements of $\Theta_k$ are chosen with probability $\Gamma^i$, $i = 1, 2, 3$ as in Equation (4.2), Equation (4.3), and Equation (4.4).

*Proof.* (Sketch of proof) The proof is immediately resulted from the problem statement. We provide an intuitive proof for this theorem. From case (1): $\mathbf{x}_{\ell,k-1}$ transitions to time $k$ according to the Markov transition kernel $\mathbb{Q}_{\boldsymbol{\theta}_{\ell,k}}(\mathbf{x}_{\ell,k-1},\cdot)$ and then is assigned to one of the existing clusters that is already used by one of the objects. From case (2): $\mathbf{x}_{\ell,k-1}$ and the cluster parameter $\boldsymbol{\theta}^{\star}_{\ell,k-1}$ transition to time $k$ according to Markov transition kernels $\mathbb{Q}_{\boldsymbol{\theta}_{\ell,k}}(\mathbf{x}_{\ell,k-1},\cdot)$ and $\zeta(\boldsymbol{\theta}^{\star}_{\ell,k-1},\cdot)$, respectively, and therefore the object is assigned to the this new cluster. From case (3): new object does not belong to any of the previously assigned clusters, i.e., a new object comes into the FOV. In this case, we generate a new parameter from the base distribution $H$ and assign the object to the newly created cluster. ■

**Algorithm 8:** DPY-STP Model for State Transition Process

```
At time $(k-1)$:
  • $\mathbf{X}_{k-1} = \{\mathbf{x}_{\ell,k-1} \ \ldots \ \mathbf{x}_{N_{k-1},k-1}\}$: collection of object states vectors
  • $\mathcal{C}_{N_{k-1}} = [c_1, \ c_2, \ \ldots, \ c_{N_{k-1}}]$, cluster assignment
  • $\Theta_{k-1} = \{\boldsymbol{\theta}_{\ell,k-1} : \ell = 1, \ldots, N_{k-1}\}$, cluster parameters
  • $D_{k-1}$, number of uniques cluster parameters
  • $\Theta^*_{k-1} = \{\boldsymbol{\theta}^*_{\ell,k-1} : \ell = 1, \ldots, D_{k-1}\}$, for unique clusters
Transitioning from time $(k-1)$ to $k$:
  Input: $\mathbf{X}_{k-1}$, $\Theta^*_{k-1}$, transition kernel $\mathbb{Q}_{\boldsymbol{\theta}_{\ell,k}}(\mathbf{x}_{\ell,k-1}, \mathbf{x}_{\ell,k})$ and probability of
         object staying in the scene $P_{k|k-1}$
if $\mathbf{x}_{\ell,k-1} \in \mathbf{X}_{k-1}$ leaves with probability $(1 - P_{k|k-1})$ then
    return null
end if
if $\mathbf{x}_{\ell,k-1} \in \mathbf{X}_{k-1}$ transitions with probability $P_{k|k-1}$ then
    $\mathbf{x}_{\ell,k-1} \sim \mathbb{Q}_{\boldsymbol{\theta}_{\ell,k}}(\mathbf{x}_{\ell,k-1}, \mathbf{x}_{\ell,k})$
    return $D_{k|k-1}$: number of unique cluster, $\mathbf{V}^*_{k|k-1} \in \mathbb{R}^{D_{k|k-1}}$: size vector,
           and $\Theta_{k|k-1}$: collection of survived parameters
end if

At time $k$:
for $\ell = 1$ to $|\mathbf{V}^*_{k|k-1}|$ do
    Draw $\boldsymbol{\theta}_{\ell,k}$ from $\zeta(\boldsymbol{\theta}^*_{\ell,k-1}, \boldsymbol{\theta}_{\ell,k})$ according to Equation (4.5)
    Draw $\mathbf{x}_{\ell,k}|\boldsymbol{\theta}_{\ell,k}$ from (Equation (4.6))
end for
return $\{\mathbf{x}_{1,k}, \mathbf{x}_{2,k}, \ldots\}$ and $\{\boldsymbol{\theta}_{1,k}, \boldsymbol{\theta}_{2,k}, \ldots\}$
```

The graphical model representing the entire process is depicted in Figure 4.1. In the following section, we discuss how this constructed prior on the states can be exploited to estimate the trajectory of objects, and then learn the hyperparameters based on the received measurements.

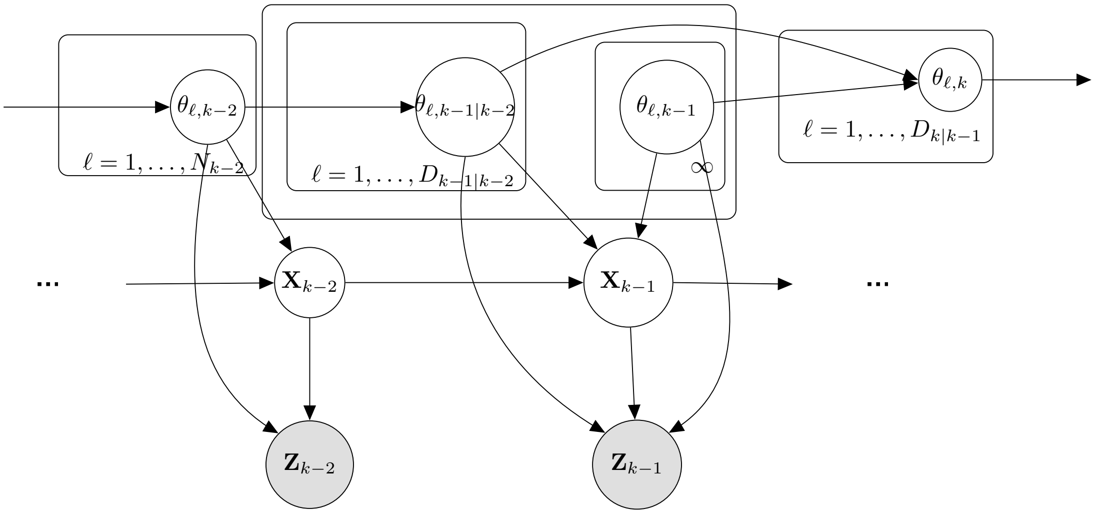


Figure 4.1: Graphical Model Capturing the Temporal Dependence, DPY-STP Construction.

## 4.2 Learning Model

The DPY-STP algorithm, summarized in Algorithm 8, provides the density estimation of objects at time step $k$ as in Equation (4.6). Upon receiving the set of measurements $\mathbf{Z}_k = \{\mathbf{z}_{1,k}, \ldots, \mathbf{z}_{M_k,k}\}$ at time step $k$, we updates the estimated density, and thus the trajectory of objects. Using Theorem 13, we introduce an infinite mixture model to update our estimates as discussed in Algorithm 8. The learning model is summarized in Algorithm 9.

To use Algorithm 9, we assume that each measurement is associated only with one object and also the measurements are independent of one another. We thus exploit Dirichlet process mixture (DPM) model as an infinite mixture model with the base distribution drawn from Algorithm 8 to update our belief. Note that he identity of the object that corresponds to a particular measurement is not known. However, the DPM model can learn the association between each measurement and the corresponding object as objects are already labeled from the DPY clustering. The

**Algorithm 9:** Infinite Mixture Model Used to Associate Measurements with Objects.

```
Input: $\{\mathbf{z}_{1,k},\ldots,\mathbf{z}_{M_k,k}\}$, $\{\mathbf{x}_{1,k},\mathbf{x}_{2,k},\ldots\}$, $\{\boldsymbol{\theta}_{1,k},\boldsymbol{\theta}_{2,k},\ldots\}$
At time $k$:
for $m = 1 : M_k$ do
    Draw $\mathbf{z}_{m,k}|\mathbf{x}_{\ell,k},\boldsymbol{\theta}_{\ell,k}$ from (Equation (4.7))
    return $\mathcal{C}_{N_k}$, cluster assignment at time $k$
end for
Update: $\mathcal{CA}_k = \mathcal{CA}_{k-1} \cup \mathcal{C}_{N_k}$
return Number of clusters $N_k$, $\mathcal{CA}_k$ and posterior distribution $\mathbf{x}_{\ell,k}|\boldsymbol{\theta}_{\ell,k},\mathbf{z}_{m,k}$
```

clustering of the measurements exploit DPY model results for the state distribution from Theorem 14,

$$\mathbf{x}_{\ell,k}|\mathbf{x}_{1,k},\ldots,\mathbf{x}_{\ell-1,k},\mathbf{X}_{k|k-1},\Theta_k \sim p(\mathbf{x}_{1,k}|\mathbf{x}_{1,k},\ldots,\mathbf{x}_{\ell-1,k},\mathbf{X}_{k|k-1},\Theta^{\star}_{k|k-1},\Theta_k), \quad (4.7)$$

and then

$$\mathbf{z}_{l,k}|\mathbf{x}_{\ell,k},\boldsymbol{\theta}_{\ell,k} \sim R(\mathbf{z}_{l,k}|\mathbf{x}_{\ell,k},\boldsymbol{\theta}_{\ell,k}) \quad (4.8)$$

for some distribution $R$ that depends on the measurement likelihood function.

Note that the DPY-STP algorithm is closely related to DDP-EEM algorithm introduced in Chapter 3, and thus both algorithms are well-defined. One can derive DDP-EMM model from the DPY-STP model by setting $d = 0$. The discount parameter $d$ is used to control the number of clusters in the model. Intuitively speaking, on account of power-law property of Pitman-Yor modeling, the higher the number of unique (non-empty) clusters is, the higher the probability of having even more unique clusters is. Furthermore, we aim to have a lower probability of having new objects for clusters with a small number of objects. Consequently, the DPY-STP is more flexible and a better match for the tracking problems with a time-varying number of objects. With a maximum number of $N_k$ objects at time step $k$, an object may stay in the scene from the previous time step, leave the scene, or enter the scene for the first time. Thus, the object state would benefit from a larger number of available clusters to ensure all dependencies are captured.

### *4.2.1 Bayesian Inference: Gibbs Sampler*

Exact posterior computation for DPY-STP algorithm is difficult when the number of parameters and observations are large. Nevertheless, we can make use of Gibbs sampling for inference in the DPY-STP where the conjugate priors are used. To provide an efficient sampling method, we introduce an auxiliary random variables to identify the cluster associations for the measurements. The resulting sampler allows model and measurement parallelization. Note that inference in DPY-STP model depends directly on the number of the clusters and number of measurements at each time step. Under the cluster assignments $\mathcal{CA}_k$, we introduce a cluster indicator $\mathcal{C}_k = \{c_{1,k}, \ldots, c_{N_k,k}\}$ at time k such that $c_{i,k} = c_{j,k}$ if and only if $\boldsymbol{\theta}_{i,k} = \boldsymbol{\theta}_{j,k}$ and $c_{i,k} = \ell$ if and only if $\boldsymbol{\theta}_{i,k} = \boldsymbol{\theta}^{\star}_{\ell,k}$ ( Note that $\boldsymbol{\theta}^{\star}_{\cdot,k}$'s indicate the unique parameters at time $k$). The cluster indicator $\mathcal{C}_k$ provides a partition the set of $\{1, \ldots, N_k\}$. Since realization of the Pitman-Yor process is almost surely a discrete random measure, we can marginalize this process and derive the successive conditional Blackwell-MacQueen distribution:

$$\boldsymbol{\theta}_{\ell,k}|\Theta \sim \sum_{\Theta_k - \{\boldsymbol{\theta}_{\ell,k}\}} \Gamma^1_{j,k} \delta_{\boldsymbol{\theta}}(\theta_{\ell,k}) + \sum_{\substack{\boldsymbol{\theta} \in \Theta^{\star}_{k|k-1} \backslash \Theta \\ \boldsymbol{\theta} \neq \boldsymbol{\theta}_{\ell,k}}} \Gamma^2_{j,k} \nu(\boldsymbol{\theta}^{\star}_{\ell,k-1}, \boldsymbol{\theta}_{\ell,k}) \delta_{\boldsymbol{\theta}}(\boldsymbol{\theta}_{\ell,k}) + \Gamma^3_k H(\boldsymbol{\theta}_{\ell,k}). \quad (4.9)$$

Assuming the base measure $H$ is nonatomic, the required conditional distribution to do local inference is derived by marginalizing over the mixing measures:

$$p(c_{i,k} = \ell | \mathcal{C}_k \setminus \{c_{i,k}\}, \mathbf{Z}_k, \text{rest}) \propto \tag{4.10}$$
$$\begin{cases} \Gamma_{\ell,k}^{1,-i} R(\mathbf{z}_{l,k} | \mathbf{x}_{\ell,k}, \boldsymbol{\theta}^\star_{\ell,k}) & \text{for cluster } \ell \text{ that has been selected} \\ \Gamma_{\ell,k}^{2,-i} R(\mathbf{z}_{l,k} | \mathbf{x}_{\ell,k}, \boldsymbol{\theta}^\star_{\ell,k}) & \text{for cluster } \ell \text{ that has not yet been selected} \\ \Gamma_{k}^{3,-i} \int R(\mathbf{z}_{l,k} | \mathbf{x}_{\ell,k}, \boldsymbol{\theta}) dH(\boldsymbol{\theta}) & \text{new cluster is created} \end{cases}$$

where $\Gamma_{\ell,k}^{j,-i}$ is the probability of choosing $c_{t,k} = \ell$ where $t \neq i$ and follows

$$\Gamma_{\ell,k}^{1,-i} = \frac{\left[ \sum_{j=1}^{D_{k-1}} \left[ \mathbf{V}^\star_{k|k-1} \right]_j \eta_{j,k|k-1} \delta_j(c_{\ell,k}) + [\mathbf{V}_k]_\ell \right]_{-i} - d}{\left[ \sum_{t=1}^{\ell-1} \sum_{j=1}^{D_{k-1}} \left[ \mathbf{V}^\star_{k|k-1} \right]_j \eta_{j,k|k-1} \delta_j(c_{t,k}) + \sum_{t=1}^{\ell-1} [\mathbf{V}_k]_t \right]_{-i} + \alpha} \tag{4.11}$$

$$\Gamma_{\ell,k}^{2,-i} = \frac{\left[ \sum_{j=1}^{D_{k-1}} \left[ \mathbf{V}^\star_{k|k-1} \right]_j \eta_{j,k|k-1} \delta_j(c_{\ell,k}) \right]_{-i} - d}{\left[ \sum_{t=1}^{\ell-1} \sum_{j=1}^{D_{k-1}} \left[ \mathbf{V}^\star_{k|k-1} \right]_j \eta_{j,k|k-1} \delta_j(c_{t,k}) + \sum_{t=1}^{\ell-1} [\mathbf{V}_k]_t \right]_{-i} + \alpha} \tag{4.12}$$

$$\Gamma_{k}^{3,-i} = \frac{|D_k|_{-i} d + \alpha}{\left[ \sum_{t=1}^{\ell-1} \sum_{j=1}^{D_{k-1}} \left[ \mathbf{V}^\star_{k|k-1} \right]_j \eta_{j,k|k-1} \delta_j(c_{t,k}) + \sum_{t=1}^{\ell-1} [\mathbf{V}_k]_t \right]_{-i} + \alpha} \tag{4.13}$$

where $[\cdot]_{-i}$ indicates the total number of object parameters observed excluding the $i$th object, $|D_k|_{-i}$ is the total number of unique clusters created at time $k$ before $i$th object is observed, and $R$ is the likelihood function. Equation (4.10) is derived by multiplying the likelihood function by the conditional prior derived in Equation (4.9).

To fully specify the sampling procedure, we also need to update the parameters, $\Theta^\star_k = \{\boldsymbol{\theta}^\star_{1,k}, \ldots, \boldsymbol{\theta}^\star_{D_k,k}\}$. To do so, we only need draw $\boldsymbol{\theta}^\star_{\ell,k}$ from a distribution proportional to

$$\prod_{\{\mathbf{z}_{l,k} : \boldsymbol{\theta}_{l,k} = \boldsymbol{\theta}^\star_{\ell,k}\}} R(\mathbf{z}_{l,k} | \mathbf{x}_{\ell,k}, \boldsymbol{\theta}^\star_{\ell,k}) dH(\boldsymbol{\theta}^\star_{\ell,k}). \tag{4.14}$$

## 4.3 Properties of DPY-STP Model

In this section, we verify properties of proposed DPY-STP model. The DPY-STP provides an exchangeable partition; thus, it is useful to provide the exchangeable partition probability function associated with it. However, this model, unlike DDP-EMM, is not always consistent. In this section. we discuss conditions under which this model is consistent in detail.

### *4.3.1 Posterior Distribution*

As mentioned in Section 4.2.1, the DPY-STP method induces a partition [1] over $\{1, 2, \ldots, N_k\}$ which is shown to be exchangeable. Let $\mathcal{C}_k = \{c_{1,k}, \ldots, c_{D_k,k}\}$ and $|\mathcal{C}_k| = \{[\mathbf{V}_k]_1, \ldots, [\mathbf{V}_k]_{D_k}\}$ be the unordered collection of clusters assignment (partition) and its cardinality. In particular, we have $|c_{j,k}| = [\mathbf{V}_k]_j$ and $\sum_{j=1}^{D_k} [\mathbf{V}_k]_j = N_k$, at time $k$. Define $\left( [\mathbf{V}_k]_1^*, \ldots, [\mathbf{V}_k]_{D_k}^* \right)$ to be the size of ordered clusters (partition) such that $[\mathbf{V}_k]_1^* \leq \ldots, \leq [\mathbf{V}_k]_{D_k}^*$. Due to exchangeability of the sequence associated with the cluster assignments (partitions), it is shown that the EPPF is given in [56] by

$$p([\mathbf{V}_k]_1^*, \ldots, [\mathbf{V}_k]_{D_k}^*) = \frac{\prod_{j=1}^{D_k} (\alpha + jd)}{\alpha^{[N_k]}} \prod_{i=1}^{D_k} (1-d)^{[\mathbf{V}_k]_i^*} \tag{4.15}$$

where $\alpha^{[n]} = \alpha(\alpha+1)\ldots(\alpha+n-1)$. Note that if we set $d = 0$ the Equation (4.15) reduces to the EPPF for the Dirichlet process with concentration parameter $\alpha$ in Equation (3.25). The induced random partition by $\mathcal{C}_k$ at each time $k$ is distributed according to the Equation (4.15).

Furthermore, if the distribution on the cluster parameters is drawn from the conditional distribution DPY-STP$_k$|DPY-STP$_{k-1}$ as in Equation (4.5) with $d > 0$, then

[1] A partition of set $\mathcal{A}$ is an unordered collection of nonempty subsets of $\mathcal{A}$ such that $\mathcal{A}$ is the disjoint union of its subsets and each element of $\mathcal{A}$ belongs to only one subset.

posterior distribution given $\boldsymbol{\theta}^{\star}_{1,k}, \ldots, \boldsymbol{\theta}^{\star}_{D_k,k}$ is the distribution of the random measure [56]

$$B_n \sum_{i=1}^{D_k} \pi_i \delta_{\boldsymbol{\theta}^{\star}_{i,k}} + (1 - B_n)\tilde{H} \tag{4.16}$$

where $B_n \sim Beta(N_k - D_k d, \alpha + D_k d)$, $(\pi_1, \ldots, \pi_{D_k}) \sim \text{Dirichlet}([\mathbf{V}_k]_1 - d, \ldots, [\mathbf{V}_k]_{D_k} - d)$, and $\tilde{H} \sim \mathcal{PY}(d, \alpha + D_k d, G^*)$ for

$$G^* = \sum_{\Theta_k} \Gamma^1_{j,k} \delta_{\boldsymbol{\theta}_{\ell,k}} + \sum_{\Theta^{\star}_{k|k-1} \backslash \Theta_k} \Gamma^2_{j,k} \zeta(\boldsymbol{\theta}^{\star}_{\ell_k - 1}, \boldsymbol{\theta}_{\ell,k}) \delta_{\boldsymbol{\theta}_{\ell,k}} + \Gamma^3_k H.$$

Note that $B_n$ and $(\pi_1, \ldots, \pi_{D_k})$, and $\tilde{H}$ are mutually independent.

### *4.3.2 Posterior Consistency of DPY-STP model*

The DDP-EMM statistical model introduced in Chapter 3 along with the introduced dependent Pitman-Yor model may be used to estimate the densities, and consequently to accurately and efficiently track the objects. As discussed in Section 3.4, DDP-based priors result in consistent posteriors. However, the Pitman-Yor process priors assume the inconsistency of the Gibbs process priors to estimate distributions. The conditions under which Gibbs processes are consistent is thoroughly studied in Section 3, Theorem 1 in [100]. Consistency of Pitman-Yor processes is the direct result of Gibbs prior consistency. The following proposition summarizes these conditions:

**Propositions 6.** Let $G_k \sim \mathcal{PY}(d, \alpha, H)$ be the prior distribution drawn from a Pitman-Yor Process. The posterior distribution of $G_k|\mathbf{Z}_k$ is consistent at probability measure $G_0$ if and only if one the following conditions holds:

A. $G$ is the mixture of at most $\lceil|\frac{\alpha}{d}|\rceil$ degenerated measures, i.e., $G_0$ is discrete

B. $H$ is proportional to $G_{0,c}$ where $G_{0,c}$ is continuous part of the probability measure $G_0$

C. $d = 0$, which is equivalent to the consistency of the Dirichlet process.

*Proof.* This Proposition immediately results from the Gibbs prior consistency theorem.

**Lemma 4.** *(Gibbs prior consistency [100])* If $G_k$ is equipped with a Gibbs process prior with non-negative coefficients $W_{N_k,D_k^\star}$ which satisfy the backward recurrence $W_{N_k,D_k^\star} = (N_k - D_k^\star d)W_{N_k+1,D_k^\star} + W_{N_k+1,D_k^\star+1}$ for $W_{1,1} = 1$ and $\sigma \in (-\infty, 1)$ such that $\frac{W_{N_k+1,D_k^\star+1}}{W_{N_k,D_k^\star}} \xrightarrow{\text{a.s.}} \eta$ and $\frac{W_{N_k+2,D_k^\star+2}}{W_{N_k+1,D_k^\star+1}} \xrightarrow{\text{a.s.}} \eta$ for $0 < \eta \leq 1$ or $\frac{W_{N_k+1,D_k^\star+1}}{W_{N_k,D_k^\star}} \xrightarrow{\text{a.s.}} 0$ almost surely, then the posterior distribution $G_k|\mathbf{Z}_k$ convergence almost surely under $G_0$ relative to weak topology to $\kappa G_{0,d} + \gamma G_{0,c} + \eta G$ for some $\alpha$ and $\gamma$ ($N_k =$ number of states at time $k$, $D_k^\star =$ number of unique clusters (partitions) at time $k$). In particular, the posterior is consistent if and only if $\eta = 0$, and one of the following holds:

A. $\sigma = 0$

B. $G_0$ is discrete

C. $G_0$ is atomless

unless, $G_{0,c}$ is proportional to $G$.

A Pitman-Yor process is a special case of a Gibbs prior, where $\frac{W_{N_k+1,D_k^\star+1}}{W_{N_k,D_k^\star}} \xrightarrow{\text{a.s.}} \frac{\alpha+D_k^\star d}{N_k+D_k^\star}$ and $\frac{W_{N_k+2,D_k^\star+2}}{W_{N_k+1,D_k^\star+1}} \xrightarrow{\text{a.s.}} \frac{\alpha+D_k^\star d+d}{N_k+D_k^\star+1}$ where both converges to $\gamma = \sigma\xi$ where $D_k^\star/N_k \to \xi$. Note that $D_k^\star$ depends directly on $N_k$. Using Lemma 4, the proof is complete. ■

Most of the discrete nonparametric priors, except for the Dirichlet process, are inconsistent when it is used to directly model continuous measurements; however, when these priors are used towards hierarchical mixture models, they generally lead to a consistent density estimator [101, 102]. On that account, the density estimators introduced in this work are all consistent and may be used to robustly and efficiently track multiple objects with a time-varying number of objects.

## 4.4 Simulations

In this section, we examine the empirical performance of the Bayesian nonparametric DPY-STP tracker. Due to the flexibility of DPY-STP method, we expect the DPY-STP tracker to refine other multiple object tracking trackers. To this end, we first compare this method to the labeled multi Bernoulli (LMB) [30] and then compared it to DDP-EMM tracker introduced in Chapter 3. We show through simulations that this model can successfully estimate the trajectory of objects and learn the number of time-varying objects. It is also shown that this method outperforms existing methods. In Section 4.4.1, we compare the performance of the DPY-STP method to the LMB using the optimal sub-pattern pattern assignment (OSPA) measure [97]; we demonstrate that our method has a lower error. Section 4.4.2 studies the comparison between DPY-STP and DDP-EMM. Our results indicate that despite the outstanding performance of DDP-EMM, DPY-STP is usually superior.

### *4.4.1 Comparison to Multi-Bernoulli Filtering*

The DPY-STP multiple object tracking method is implemented using MCMC sampling methods, together with Algorithms 8 and 9. To demonstrate the performance of this method, we simulated a dynamic linear tracking example using five objects that enter, leave, and/or stay in the scene at different times, as summarized in Table 4.1. The performance is compared to that of the LMB tracker.

Assume that the $\ell$th state vector is $\mathbf{x}_{\ell,k} = [x_{\ell,k}, y_{\ell,k}, \dot{x}_{\ell,k}, \dot{y}_{\ell,k}, \omega_{\ell,k}]^T, \ell = 1, \dots, N_k$, where $N_k$ is the time-dependent target cardinality. This vectors consists of $[x_{\ell,k}, y_{\ell,k}]^T$, $[\dot{x}_{\ell,k}, \dot{y}_{\ell,k}]^T$, and $\omega_{\ell,k}$ that are the 2-dimensional (2-D) position, velocity, and target turn rate, respectively. The actual time-dependent trajectories are shown in Figure 4.2. The transition probability density $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ for the coordinated turn mo-

Table 4.1: Time Intervals that Objects Enter/Leave the Scene

| Object | Time step entering scene | Time step leaving the scene |
|---|---|---|
| Object 1 | $k=0$ | $k=70$ |
| Object 2 | $k=5$ | $k=100$ |
| Object 3 | $k=10$ | $k=100$ |
| Object 4 | $k=20$ | $k=45$ |
| Object 5 | $k=30$ | $k=80$ |

tion model is assumed to be a Gaussian distribution with mean vector $\boldsymbol{\mu}=[\boldsymbol{\zeta}^T\ \omega_{k-1}]^T$ where $\boldsymbol{\zeta}=A_{\omega_{k-1}}\mathbf{x}_{k-1}$ and covariance matrix $Q=\text{diag}([\sigma_w^2 BB^T, \sigma_u^2])$ where $\sigma_w=15$ m/s$^2$, $\sigma_u=\pi/180$ radians/s, and

$$A_{\omega_{k-1}}=\begin{bmatrix} 1 & \frac{\sin(\omega_{k-1})}{\omega_{k-1}} & 0 & -\frac{1-\cos(\omega_{k-1})}{\omega_{k-1}} \\ 0 & \cos(\omega_{k-1}) & 0 & -\sin(\omega_{k-1}) \\ 0 & \frac{1-\cos(\omega_{k-1})}{\omega_{k-1}} & 1 & \frac{\sin(\omega_{k-1})}{\omega_{k-1}} \\ 0 & \sin(\omega_{k-1}) & 0 & \cos(\omega_{k-1}) \end{bmatrix}, B=\begin{bmatrix} \frac{1}{2} & 0 \\ 1 & 0 \\ 0 & \frac{1}{2} \\ 0 & 1 \end{bmatrix}. \tag{4.17}$$

The measurement vector $\mathbf{z}_k=[\phi_k\ r_k]^T$ at time $k$ includes bearing $\phi_k$ and range $r_k$, where $r\in[0,2,000]$ m and $\phi\in[-\frac{\pi}{2},\frac{\pi}{2}]$. The measurement noise is assumed zero-mean Gaussian with variance $\sigma_r^2=25$ and $\sigma_\phi^2=(\frac{\pi}{180})^2$.

For the simulations, we used 10,000 Monte Carlo runs, $K=100$ overall observed time steps, and $-3$ dB signal-to-noise-ratio (SNR). For the parameters of the DPY-STP method, the prior used on the parameters is a normal-inverse-Wishart distribution, $\mathcal{NIW}(\mu_0,\lambda,\nu,\boldsymbol{\Psi})$, with values $\mu_0=0$, $\lambda=0.001$, $\nu=50$, and an identity matrix for $\boldsymbol{\Psi}$. The discount parameter is selected as $d\in(0,1)$, and a Gamma distribution prior, $\Gamma(\alpha;1,0.2)$, is used over the concentration parameter $\alpha$. Using the DPY-STP, the estimated $x$ and $y$ coordinates are shown to match the true coordinates in Fig-

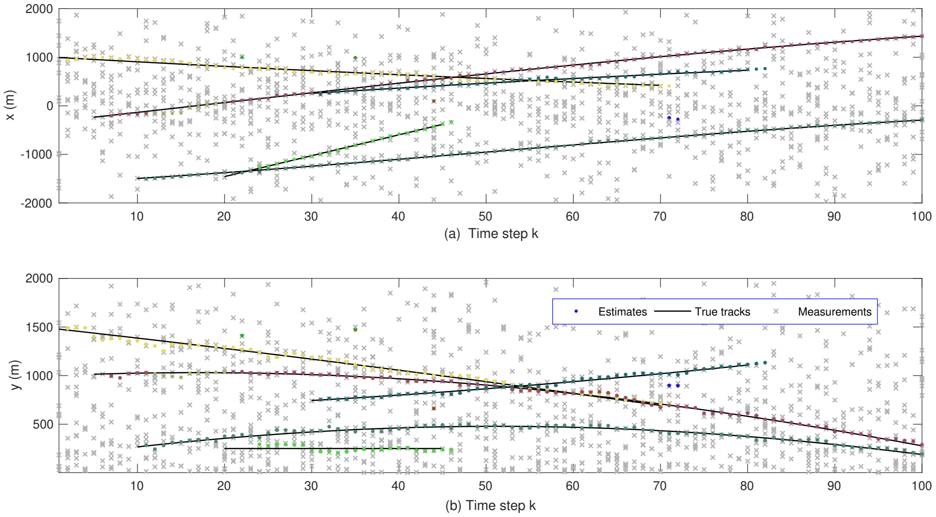


Figure 4.2: True and Estimated (a) $x$-coordinate and (b) $y$-coordinate as A Function of the Time Step $k$ for Five Objects.

ure 4.2(a) and Figure 4.2(b), respectively. Figure 4.3 demonstrate that DPY-STP can more accurately estimate the number of object at each time step. In comparison with the LMB, the DPY-STP shows a higher estimation accuracy for the $x$ and $y$ coordinates. The OSPA measure to compare the performances of the DPY-STP to the LMB, as in Figure 4.4, indicates a higher accuracy and consistency for both the range and the time-varying object cardinality estimate of multiple targets using the DPY-STP tracker.

### *4.4.2 Comparison between DPY-STP and DDP-EMM*

In this section, we compare the DPY-STP tracker to the DDP-EMM tracker to demonstrate that object states may benefit from a larger number of available clusters, given the conditions in Equation (4.5). Therefore, we compare both proposed methods in Chapter 3 and Chapter 4 to verify that the algorithm based on the dependent

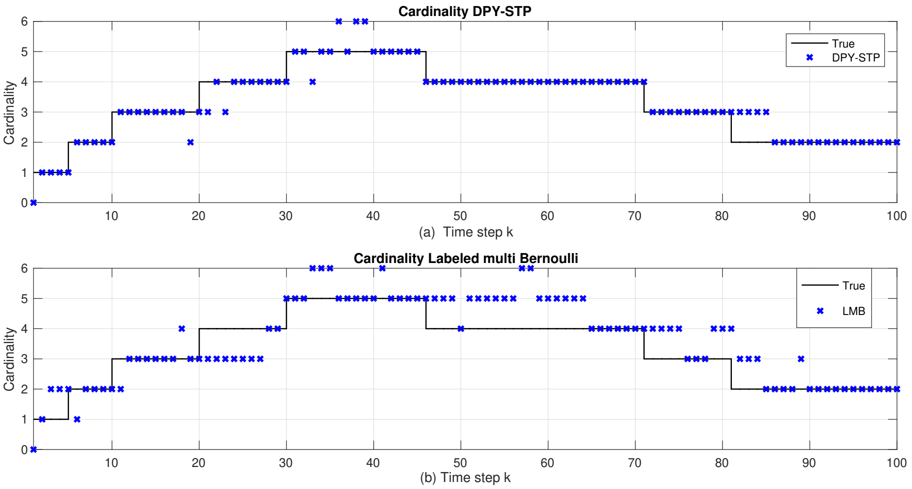


Figure 4.3: (a) True and Estimated $x$-coordinate (Top) and $y$-coordinate (Bottom) as A Function of Time Step $k$ for Five Objects. (b) OSPA (Order $p = 1$ and Cut-off $c = 100$ for Range (Top).

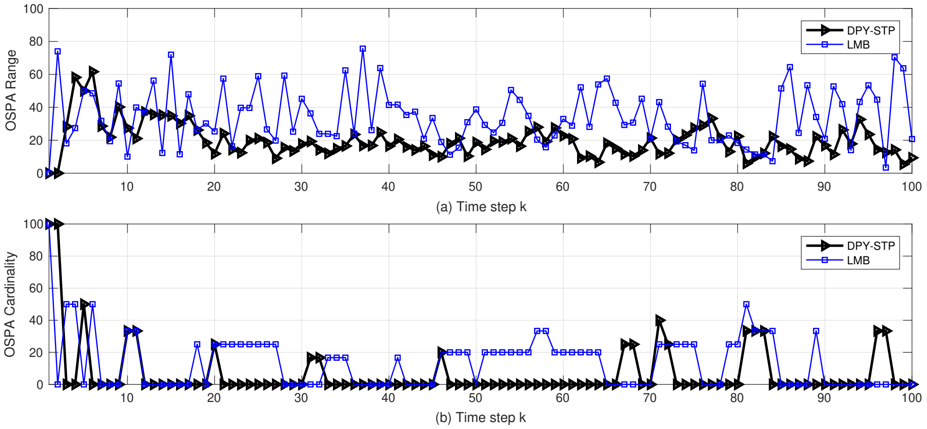
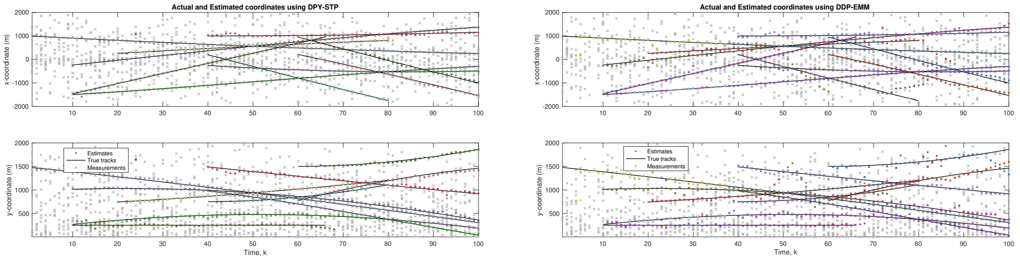


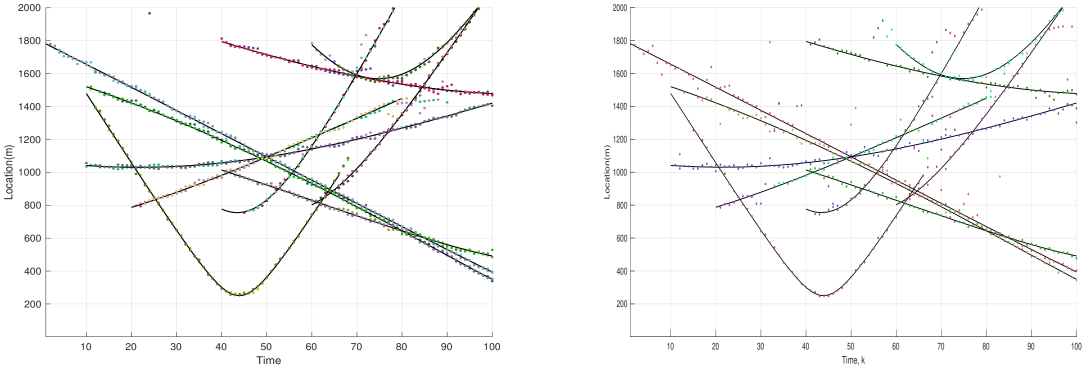
Figure 4.4: Cardinality (Bottom) Averaged over 10,000 MC Simulations for the DPY-STP and the Labeled Multi-Bernoulli (LMB) Based Tracking Approaches.

Pitman-Yor process may have better results than DDP-EMM tracker. To do this end, we consider the problem of tracking 10 objects using both methods. We assume the base distribution to have a normal-inverse-Wishart distribution, $\mathcal{NIW}(\mu_0, \lambda, \nu, \mathbf{\Psi})$ where $m_0 = 0, \lambda = 0, \nu = 100,$ and $\mathbf{\Psi} = I$. We select $\alpha$ and $d$ the same way as discussed in Section 4.4.1.



(a) (b)

Figure 4.5: (a) Actual and Estimated $x$ and $y$-coordinates through DPY-STP (b) Actual and Estimated $x$ and $y$-coordinates through DDP-EMM.



(a) (b)

Figure 4.6: (a) Actual and Estimated Location through DPY-STP (b) Actual and Estimated Location through DDP-EMM.

Figure 4.5a and Figure 4.5b display the actual and estimated coordinates through DPY-STP and DDM-EMM, respectively. We show the location estimation of objects through the DPY-STP and the DDP-EMM in Figure 4.6a and Figure 4.6b,

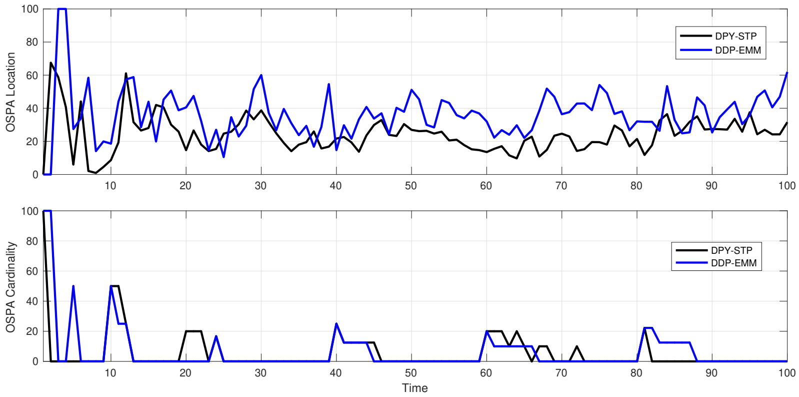


Figure 4.7: OSPA Comparison between DPY-STP (Black) and DDP-EMM (Blue) for Cut-off $c = 100$ and Order $p = 1$.

respectively. The Figure 4.6a shows that DPY-STP has higher accuracy compared to DDP-EMM model. We also demonstrate the comparison between the DPY-STP and the DDP-EMM performances using the OSPA metric with cut-off $c = 100$ and order $p = 1$. We observe that DPY-STP has a better performance compared to DDP-EMM as depicted in Figure 4.7.

## 4.5 Discussion

The preceding results demonstrated the substantial benefits of proposed dependent Pitman-Yor multi-object tracker over the dependent Dirichlet process multi-object tracker as having small clusters is now more probable. Our result further manifested that the proposed dependent nonparametric model leads to a learning algorithm which can successfully provide object identity and cardinality. We studied conditions under which our model is consistent. This model is also empirically compared to the famous labeled multi-Bernoulli filter and outperformed it.

Chapter 5

# BAYESIAN NONPARAMETRICS ON RANDOM INFINITE TREES AND ITS APPLICATION ON MODELING IN MULTIPLE OBJECT TRACKING

Recent methods for tracking multiple objects have addressed important issues such as time-varying cardinality, unordered sets of measurements, and object labeling. However, some of these methods may be computationally expensive. The main challenge is how to robustly associate objects on a new scene with previously estimated objects efficiently. In this chapter, We propose a new method based on infinite random trees to track a dynamically varying number of objects using information from previously tracked ones. We propose a new approach where links graph theory, Bayesian nonparametric modeling, and multi-object tracking. Our model exploits Bayesian nonparametric modeling and introduces a diffusion-based process to construct infinite random trees. This method can robustly track objects and compute the trajectory only by tracing each leaf. In Section 5.1, we discuss the problem at hand and introduce the tracking model in Section 5.2. A Bayesian nonparametric prior on infinite random trees are constructed in Section 5.3, and its properties are discussed. Section 5.4 discusses the Bayesian nonparametric inference model to infer the trajectory and update the object cardinality. We conclude this chapter by simulations to demonstrate the performance of the proposed algorithm and compare it to a labeled multi-Bernoulli (LMB) filter based tracker and DDP-EMM introduced in Chapter 3. A portion of the results was presented at the 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) [103].

## 5.1 Introduction

The multiple object tracking problem could include estimating the objects' time-varying cardinality, label and state parameters, among other information, depending on the application [14, 30, 104–109]. Among various approaches, random finite set methods were used to solve this problem, together with probability hypothesis density filtering and multi-Bernoulli or labeled multi-Bernoulli filtering [14, 30, 104, 106]. Nonparametric Bayesian methods were recently used for modeling evolving object state priors. In [34], the hierarchical Dirichlet process was used as a prior on the number of unobserved input modes to track maneuvering objects. We recently used the dependent Dirichlet process to model the object prior and adaptively estimate both the object label and cardinality at each time step [110].

The Dirichlet diffusion trees (DDT), and its Pitman-Yor diffusion tree generalization, nonparametric Bayesian priors over tree structures, are thus useful for estimating latent parameters with a hierarchical structure [111–113]. They were used, for example, in [114], as structure priors to infer different possible scenarios based on trees of different depth and path lengths. It was demonstrated in [115] that the high computational cost of Markov chain Monte Carlo (MCMC) inference can be avoided using efficient approximate inference DDT models. In this chapter, we propose a dependent Poisson diffusion tree (D-PoDT) that extends the capability of DDTs to model hierarchies to also capture dependencies among the object states for the multiple object tracking problem. The dependent Poisson diffusion process (D-PoDP) introduces a prior on the space of the object state parameters using an infinite random tree. It is used as a state prior to capture the time-dependency among the states and estimate the state parameters. A time-dependent process is introduced to infer from the measurements and update the object state parameters, label the objects and estimate the object cardinality at each time step. An MCMC sampling method integrates the distribution over the infinite random tree and the time-dependent process for updating the states.

## 5.2 Tracking Model

We consider a multiple object tracking model with time-varying numbers of objects entering, leaving or remaining in the scene at each time step $k$. The object cardinality $N_k$ and the number of measurements $L_k$ are both assumed unknown [106, 110]. This tracking model is used to jointly estimate the object state information and the cardinality at each time step. We assume that the sample spaces of the $\ell$th object state vector $\mathbf{x}_{\ell,k}$, $\ell = 1, \ldots, N_k$ and $l$th measurement vector $\mathbf{z}_{l,k}$, $l = 1, \ldots, L_k$, at time step $k$, are $\mathcal{X} \subseteq \mathbb{R}^{n_x}$ and $\mathcal{Z} \subseteq \mathbb{R}^{n_z}$, respectively.

We also assume that the sequence $X_k = \{\mathbf{x}_{k,1}, \ldots, \mathbf{x}_{k,N_k}\}$ corresponds to the configuration of the multiple object state vectors at time step $k$.

Given the state vector configuration at time $(k-1)$, we consider three possible scenarios for the $\ell$th object and its state vector at time step $k$: (a) the object leaves the scene with probability $(1 - \mathrm{P}_{\ell,k|k-1})$; (b) the object remains in the scene with probability $\mathrm{P}_{\ell,k|k-1}$ and its state $\mathbf{x}_{\ell,k-1}$ transitions with probability $\mathrm{P}_{\boldsymbol{\theta}}(\mathbf{x}_k|\mathbf{x}_{k-1})$ and unknown parameter vector $\boldsymbol{\theta}$; and (c) a new object, with state $\mathbf{x}_{\ell,k} \in X_k$, enters the scene generating a measurement. We also assume that each measurement is generated by only one object and that measurements are independent of one another.

## 5.3 Dependent Diffusion Prior Modeling

We propose a new method for multiple object tracking based on a D-PoDP. These are similar to the Dirichlet diffusion trees in [111] and Pitman-Yor diffusion trees in [112] in that, they can be used as priors to latent parameters to capture hierarchical structure. The D-PoDTs are different, however, in that the prior can directly incorporate time-dependent learned information. For multiple object tracking, the state prior can include the number of objects at the current time and the object label at the previous time. Thus, the proposed method can be used to make inference on the object labels over related information by tracing random tree paths. Following outlines the proposed D-PoDP model.

### *5.3.1 Poisson Diffusion Process*

We consider a class of priors on trees whose terminal nodes (leaves) are the object state parameters, and whose non-terminal nodes (branch nodes) represent the clustering of the state parameters in a hierarchy. We assume that a tree may have an infinite number of vertices, and every edge can occur with some probability. The probability of an edge occurring that violates the tree conditions is assumed zero. We assume that the first vertex (at time step $k=0$) is drawn from $P_{\boldsymbol{\theta}_0}$ with probability 1. To generate this infinite random tree, the branch nodes and leaves must be specified. We describe the generative process in terms of a diffusion process on a unit interval; that is, the leaves correspond to the location of the diffusion process at time step $k=1$. Each point starts at time $k=0$ and follows a diffusion process, i.e., a Brownian motion, until time $k=1$, where it is observed. For example, we assume that the first object state at time step $k=1$, $\boldsymbol{\theta}_{1,1}$, is drawn from a diffusion process and fixed. The second object state, $\boldsymbol{\theta}_{1,2}$, starts at time $k=0$ and follows the same path as $\boldsymbol{\theta}_{1,1}$ up to time $\delta t$ (time between steps) before it diverges from the first path and takes an independent path. The generative process for the $i$th object parameter at time step $k=1$ is as follows. At a branch point, if $\boldsymbol{\theta}_{1,i}$ does not diverge off the branch before reaching to the previous divergence point, then the previous branches are selected with probability

$$\Pr(\text{select } j\text{th path}) = \frac{n_j - \beta}{m+\eta}, \quad \Pr(\text{diverge}) = \frac{\eta - \beta K}{m+\eta}\,. \tag{5.1}$$

Here, $n_j$ is the number of objects previously in the $j$th branch, $K$ is the total number of branches originating from this branch point, $m = \sum_{l=1}^{K} n_l$, and $\beta$ and $\eta$ are discount and concentration hyperparameters. It was shown in [112] that, since the specific diffusion path taken between nodes can be ignored, the probability of generating a specific tree structure with associated divergence times can be determined by the accumulative divergence function $H(\cdot)$; this can analytically determine the locations at each leaf node. Therefore, $\boldsymbol{\theta}_{1,i}$ follows the path of the previous points and diverges in the interval $\delta t$, assuming it has not diverged up to time $t \in [0,1]$, with probability

$$\frac{\Gamma(m-\beta)}{\Gamma(m+1-\eta)} \int_{\delta t} dH(s)\,.$$

Here, $\Gamma$ is the gamma function, $m$ is the number of points that have previously traversed this path and $\beta$ and $\eta$ are discount and concentration hyperparameters, respectively. For large $m$, the probability of diverging from this path is small. As a result, an infinite random tree can be generated from which there is a probability measure on each vertex that is dependent on its parent vertex. Note that the random tree generated is exchangeable [112]. Therefore, the probability of generating a specific tree, divergence times, and divergence locations are invariant to the ordering of the object state parameters.

The proposed algorithm is initialized by drawing $N_1$ from a Poisson distribution, $N_1 \sim \text{Po}(\alpha)$ for some hyperparameter $\alpha$. Subsequently, we select $N_1$ state parameters (leaves), which, without loss of generality, can be assumed to be the first $N_1$ leaves due to exchangeability. We then set $\{\boldsymbol{\theta}_{1,1}, \ldots, \boldsymbol{\theta}_{1,N_1}\}$ to be the first $N_1$ parameters generated through this process that are associated with the state configuration $\{\mathbf{x}_{1,1}, \ldots, \mathbf{x}_{1,N_1}\}$.

### *5.3.2 Transition from Time $k-1$ to Time $k$*

We define $V_{k-1}$ and $V_{B,k-1}$ to be the set of generated state parameters (leaves) nodes and branch nodes, respectively, that are connected to the state parameter (leaf) node at time $k-1$. Each point $\boldsymbol{\theta}_{\ell,k-1} \in V_{k-1}$ that is generated in the tree has two options: (i) it can remain in the tree with probability $\mathrm{P}_{\ell,k|k-1}$ and transition to $\boldsymbol{\theta}_{\ell,k}$ according to the transition kernel probability $\nu(\boldsymbol{\theta}_{\ell,k-1}, \boldsymbol{\theta}_{\ell,k})$ with the corresponding state transition is proportional to the transition probability $\mathbb{Q}_{\boldsymbol{\theta}_{\ell,k}}(\mathbf{x}_{\ell,k-1}, \mathbf{x}_{\ell,k})$; (ii) it can leave the tree with probability $(1 - \mathrm{P}_{\ell,k|k-1})$. We assume that the following parameters are available at time $k-1$:

- $N_{k-1}$, number of objects
- $V_{k-1} = \{\boldsymbol{\theta}_{k-1,1}, \ldots, \boldsymbol{\theta}_{k-1,N_{k-1}}\}$, generated parameters
- $V_{B,k-1}$, branch nodes connected to a leaf node
- $V_{k|k-1} \subseteq V_{k-1}$, survived parameters
- $V_{B,k|k-1} \subseteq V_{B,k-1}$, survived branch nodes
- $S_{a,k-1}$, siblings with common parent branch node $a$
- $S_{a,k|k-1} \subseteq S_{a,k-1}$, survived siblings with common parent branch node $a$

Note that if all the leaves connected to a branch node disappear, the branch node is removed from the set of branch nodes. A probability vector $\mathbf{p}_{\text{branch}} = [p_a]_{a \in V_{B,k|k-1} \cup \delta}$ is then assigned to the survived branch nodes as

$$p_a = \begin{cases} \frac{|S_{a,k-1}|+|S_{a,k|k-1}|-\gamma}{N_{B,k|k-1}-1+\sum_{a \in V_{B,k|k-1}} |V_{a,k-1}|+\zeta}, & a \in V_{B,k|k-1} \\ \\ \frac{\zeta-|V_{B,k|k-1}|\gamma}{N_{B,k|k-1}-1+\sum_{a \in V_{B,k|k-1}} |V_{a,k-1}|+\zeta}, & a = \delta \end{cases}$$

where $|S_{a,k-1}|$ is the cardinality of the set $S_{a,k-1}$, $N_{B,k|k-1}$ is the number of points that survives after transition, $\delta$ denotes a new branch, $p_\delta$ is the probability of generating a new branch, and $\zeta$ and $\gamma$ are hyperparameters.

### *5.3.3 Evolution and Parameter Estimation at Time $k$*

At time $k$, we utilize the distribution on set $V_{k|k-1}$ to find $V_k$. To this end, we can assume that $\boldsymbol{\theta}_{i,k|k-1} \in S_{a,k|k-1}$, $i=1,\ldots,|V_{k|k-1}|$ are transitioned from time $k-1$ to $k$. We draw $\tilde{N}_{i,k|k-1} {\sim} \text{Po}(\frac{p_a \times \alpha}{2\,|S_{a,k|k-1}|})$ and draw $\tilde{N}_{i,k|k-1}$ points given $\boldsymbol{\theta}_{i,k|k-1}$ based on a diffusion process described in Section 5.3.1. At time $k$, we also draw $\tilde{N}_{\delta,k|k-1} \sim \text{Po}(\frac{p_\delta \times \alpha}{2})$ and draw $\tilde{N}_{\delta,k|k-1}$ new points from the infinite random graph from $P_{\boldsymbol{\theta}_0}$. We set $\tilde{N}_k = \Sigma_i \tilde{N}_{i,k|k-1}$ and $\tilde{V}_k = \{\theta_1,\ldots\theta_{\tilde{N}_k}\}$. The overall algorithm is summarized in Algorithm 10.

## 5.4 Inference Model

The D-PoDP in Algorithm 10 provides a joint estimation of the object state parameters and number of objects, at time step $k$. At time $k$, the measurement vector, $\mathbf{z}_{l,k}$, $l=1,\ldots,L_k$, becomes available to update the time-dependent cardinality and infer the posterior distribution. Note that the probability of selecting some of the generated parameters may be zero; also, some new parameters may also need to be generated. We introduce an algorithm to dependently cluster these measurements as follows.

We use the state parameter vector distribution from the output of Algorithm 10 as the mixing distribution to infer measurement distributions to update the object cardinality. The probability of choosing a parameter $\boldsymbol{\theta}_{i,k}$ is proportional to the popularity of the parameter at time $k$, in addition to the cardinality of the set of siblings with the common parent branch node at time $k-1$. Specifically, if we assume that $\boldsymbol{\theta}_{\ell,k}$ is transitioned from $\boldsymbol{\theta}_{\ell,k-1}$ (for which it shares the common parent $a$), then $\pi_\ell = \text{Pr}(\text{select } \boldsymbol{\theta}_{\ell,k}) \propto (n_{\ell,k} + |S_{a,k-1}|)$, where $n_{\ell,k}$ is the number of measurements that have already selected $\boldsymbol{\theta}_{\ell,k}$ at time $k$. The probability of selecting a parameter that has

**Algorithm 10:** D-PoDP Algorithm

```
Initialization:
• Draw $\boldsymbol{\theta}_0^0 \sim P_{\boldsymbol{\theta}_0}$
• Draw $N_1 \sim \mathrm{Po}(\alpha)$
• Generate $\{\boldsymbol{\theta}_{1,1}, \ldots, \boldsymbol{\theta}_{1,N_1}\}$ based on a diffusion process with branching
probability of convergence in (Section 5.3.1)

Transitioning from time $k-1$ to $k$
for $\boldsymbol{\theta}_{i,k|k-1} \in V_{k|k-1}$ do
    Draw $\tilde{N}_{i,k|k-1} \sim \mathrm{Po}(\frac{p_a \times \alpha}{2\,|S_{a,k|k-1}|})$
    Generate $\tilde{N}_{i,k|k-1}$ parameter points given $\boldsymbol{\theta}_{i,k|k-1}$ using a diffusion process
end for
• Draw $\tilde{N}_{\delta,k|k-1} \sim \mathrm{Po}(\frac{p_\delta \times \alpha}{2})$
• Draw $\tilde{N}_{\delta,k|k-1}$ new parameter points from the base distribution $P_{\boldsymbol{\theta}_0}$
following a diffusion process

At time $k$
Set $\tilde{N}_k = \sum_i \tilde{N}_{i,k|k-1}$
Set $\tilde{V}_k = \{\boldsymbol{\theta}_{k,1}, \ldots, \boldsymbol{\theta}_{k,\tilde{N}_k}\}$.
Set $X_k = \{\mathbf{x}_{k,1}, \ldots \mathbf{x}_{k,\tilde{N}_k}\}$.
```

not been used up to time $k$ is proportional to some hyperparameter $\lambda$. In particular, $p(\mathbf{z}_{l,k} \mid \mathbf{x}_{\ell,k}, \boldsymbol{\theta}_{\ell,k}, \pi_\ell)$ can be inferred as

$$\pi_\ell \propto \begin{cases} n_{\ell,k} + |S_{a,k-1}|, & \theta_{\ell,k-1} \in S_{a,k-1}, \theta_{\ell,k} \in \tilde{V}_k \\ \\ \lambda, & \quad\quad\quad\; \mathit{New}\ \theta_{\ell,k} \end{cases} \tag{2}$$

$$\mathbf{x}_{\ell,k} \mid \boldsymbol{\theta}_{\ell,k}, \mathbf{X}_{k|k-1} \sim G(\theta_{\ell,k}) \tag{3}$$

$$\mathbf{z}_{\ell,k} \mid \mathbf{x}_{\ell,k}, \boldsymbol{\theta}_{\ell,k}, \pi_\ell \sim F(\mathbf{x}_{\ell,k}, \boldsymbol{\theta}_{\ell,k}) \tag{4}$$

where $\mathbf{X}_{k|k-1}$ is the set of states whose objects survive from time step $(k-1)$ to $k$ and $G$ and $F$ are two appropriately selected distributions that come from the physical model. Algorithm 11 summarizes the implementation of the dependent mixtures to cluster the measurements and track the objects. Note that since the D-PoDP is used to find the object trajectories, one needs to trace the random tree. Algorithms 10 and 11, together with MCMC sampling methods, constitute the proposed D-PoDP multiple object tracking algorithm. Sampling in both algorithms is performed using MCMC methods; in particular, we use Gibbs sampling for models based on conjugate prior distributions.

**Algorithm 11:** Dependent Mixture Model to Cluster Measurements and Track Objects

```
Input: Measurements: $\{\mathbf{z}_{1,k}, \ldots, \mathbf{z}_{k,L_k}\}$
Output: $N_k$, cluster configurations, and posterior
At time k
Sample $\{\boldsymbol{\theta}_{1,k}, \ldots, \boldsymbol{\theta}_{k,\tilde{N}_k}\}$ and $\{\mathbf{x}_{k,1}, \ldots, \mathbf{x}_{k,\tilde{N}_k}\}$ according to Algorithm 10
Draw $\{\pi_i\}$ according to (Equation (2))
for $l = 1$ to $L_k$ do
    Sample $\mathbf{z}_{l,k}|\mathbf{x}_{\ell,k}, \boldsymbol{\theta}_{\ell,k}, \pi_\ell$ using (Equation (4))
end for
$N_k \leftarrow \tilde{N}_k$
$V_k = \{\boldsymbol{\theta}_{1,k}, \ldots, \boldsymbol{\theta}_{N_k,k}\}$
return  $N_k$ and posterior of $\mathbf{z}_{l,k}|\mathbf{x}_{\ell,k}, \boldsymbol{\theta}_{\ell,k}, \pi_\ell$
```

## 5.5 Simulations

In this section, we demonstrate through simulations the performance of the D-PoPD algorithms. We first compare this tracker to that of labeled multi-Bernoulli (LMB). We show this method is efficient and can outperform the LMB tracker. A comparison between the D-PoDP tracker and DPY-STP is manifested. This comparison shows that the performance of the D-PoDP tracking method is approximately the same as the DPY-STP. However, the D-PoDP algorithm is easier to implement and can more efficiently track the objects.

### *5.5.1 Comparison to Labeled Multi-Bernoulli Tracker*

In order to demonstrate the performance of our proposed D-PoPD method, we simulated a dynamic nonlinear tracking example using five objects that enter and leave a scene at different times. The overall observed time is $K = 100$ times steps and the signal-to-noise ratio (SNR) was -3 dB. The time steps over which each object is present in the scene is summarized in Table 5.1. The time steps are also depicted in Figure 5.1(a) and (b) that show that $x$ and $y$-coordinates of the true trajectory of each object. The D-PoPD estimated $x$ and $y$-coordinates of the trajectory of each object are also shown in Figure 5.1(a) and (b).   The D-PoPD algorithm was compared with the labeled multi-Bernoulli (LMB) based tracker; both algorithms used 10,000 Monte Carlo (MC) simulations. As shown in Figure 5.2a and Figure 5.2b, the proposed tracker is more accurate in estimating the time-dependent object cardinality than the LMB. This is also demonstrated using the optimal sub-pattern assignment (OSPA) metric (of order $\mathrm{p} = 1$ and cut-off $\mathrm{c} = 100$) for range an cardinality in Figure 5.3(a) and Figure 5.3(b), respectively. As it can be seen for the D-PoPD, for example, for the cardinality OSPA measure, the highest error is observed at time step $k = 0$, when

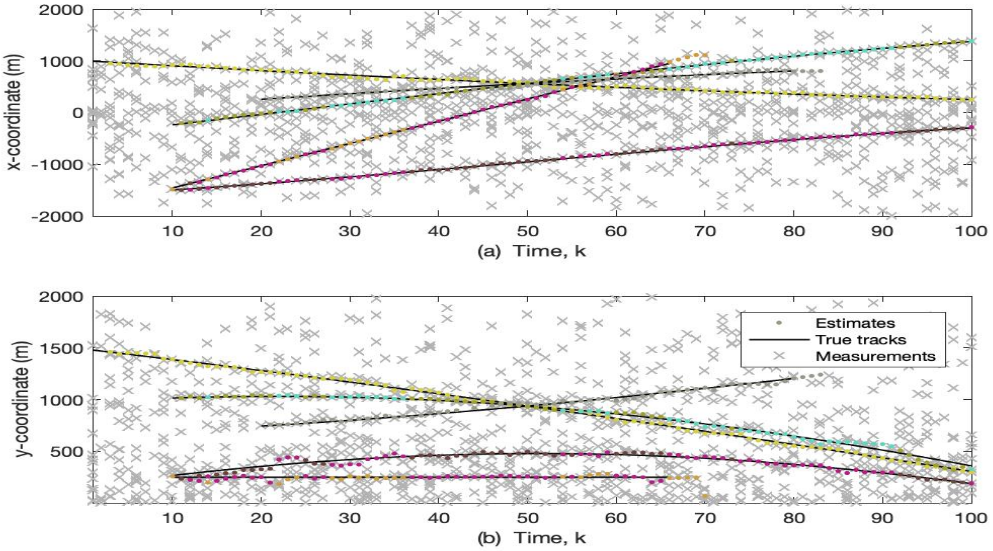


Figure 5.1: True and Estimated (a) $x$- and (b) $y$-coordinates as A Function of the Time Step $k$ of Five Objects.

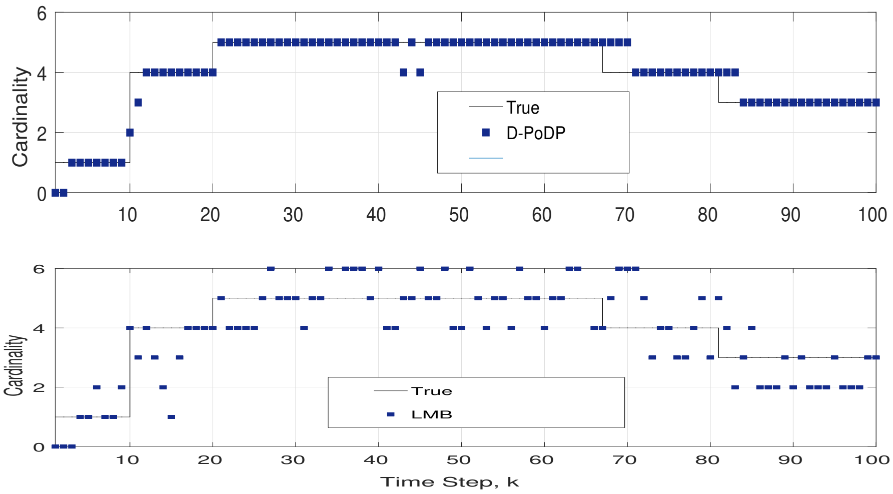


Figure 5.2: Comparison of estimated cardinality using proposed D-PoDP method (top) and LMB (bottom) when tracking 5 objects.

Table 5.1: Time Step at Which Object Enters and Leaves the Scene.

| Object | Time Enters | Time Leaves |
|---|---|---|
| Object 1 | $k = 0$ | $k = 100$ |
| Object 2 | $k = 10$ | $k = 100$ |
| Object 3 | $k = 10$ | $k = 100$ |
| Object 4 | $k = 10$ | $k = 60$ |
| Object 5 | $k = 20$ | $k = 80$ |

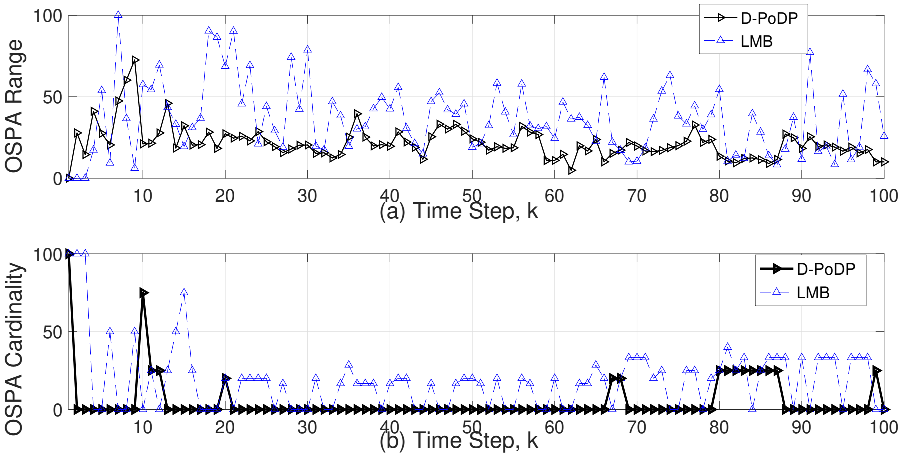


Figure 5.3: OSPA of Order $p = 1$ and Cut-off $c = 100$ for (a) Range and (b) Cardinality Averaged over 10,000 MC Simulations for the Proposed D-PoPD and the Labeled Multi-Bernoulli (LMB) Based Tracker.

the first object enters the scene and then at time step $k = 10$, when three new objects enter the scene. The method performs very well for a long time, tracking all four objects in the scene, with only a small error that soon decreases object 5 enters the scene. It continues to track the correct number of objects even when object 4 leaves the scene.

### *5.5.2 Comparison to DPY-STP Tracking Model*

In this section, we compare the D-PoDP tracker to DPY-STP tracker for ten objects at SNR = -3 dB. Figure 5.4 depicts this comparison. The OSPA comparison is the order of $p = 1$ and performed at cut-off $c = 100$. Figure 5.4 is obtained by averaging over 10,000 Monte Carlo runs. The performance for both trackers seem to approximately be the same on average. However, the D-PoDP tracking method on the infinite random tree is simpler to implement. The order of complexity for search on this tree it the worst case is order of $O(N_k)$. Hence, the D-PoDP is shown to be much faster algorithm.

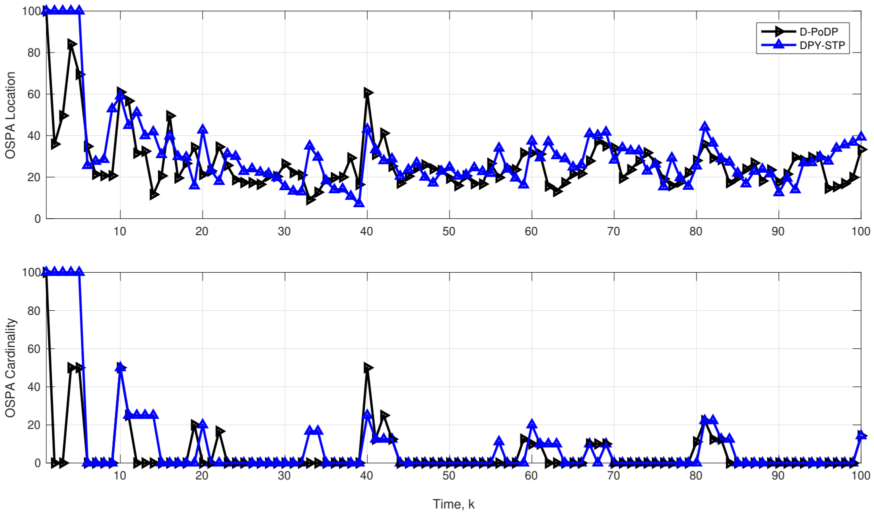


Figure 5.4: OSPA of Order $p = 1$ and Cut-off $c = 100$ for (a) Location and (b) Cardinality Averaged over 10,000 MC Simulations for the Proposed D-PoPD and the DPY-STP.

## 5.6 Discussion

In this chapter, we presented a novel class of infinite random trees to address the multiple object tracking via diffusion processes. We generated infinite random trees where tracing each path on the tree allows for tracking object trajectories. We demonstrated that integrating the proposed dependent Bayesian nonparametric modeling through Poisson diffusion process with multiple object tracking can efficiently obtain object tracks, labels, and time-varying cardinality. Moreover, the Markov chain Monte Carlo implementation of the proposed tracking framework verifies the accuracy and simplicity of this algorithm.

Chapter 6

# DEPENDENT OBSERVATIONS FROM MULTIPLE SENSORS FOR MULTIPLE OBJECT TRACKING

A multimodal sensing system can facilitate the development of algorithms by incorporating and learning new information using observations collected from multiple but disparate sensors. In particular, the integration of multiple modalities can lead to significant performance improvement for tracking objects in diverse operational and environmental conditions [104, 116]. However, incorporating the dependent measurements can be troublesome since pooling all measurements can cause loss of the information collected by the sensors [117, 118]. In this chapter, we consider the problem of state estimation for a dynamic system with dependent measurements where multiple sensors measure the dependent observations. Since sensors observe the same scene, the received measurements from the sensors are correlated. The goal is not only to distinguish whether these observed measurements are from the object but also to estimate the object trajectory using measurement models that match the observations. We first address the problem of tracking a single object with multiple correlated measurements from multiple sensors and then extend this problem to multiple object tracking. The Bayesian nonparametric paradigm is an elegant and flexible approach for modeling complex and dependent observations with unknown latent dimensionality. Hence, we exploit a Bayesian nonparametric approach to address the problem of tracking with multiple correlated sensors measurements. In Section 6.1, we study how to incorporate dependent measurements to achieve the best results through grouping the multimodal dependent measurements via a Bayesian hierarchical model and then estimate the tracks using the grouped measurements. We then extend this problem to the multimodal multi-object tracking in Section 6.2. We illustrate this nonparametric model in Section 6.3. Our results were presented at the 2019 53$^{\text{rd}}$ Asilomar Conference on Signals, Systems, and Computers [82] and 2019 22$^{\text{nd}}$ Information Fusion conference [119].

## 6.1 Multi Sensor Dependent Observations: Single Object Tracking

In this section, we introduce a hierarchical modeling to utilize the dependency among the collected measurements. We propose a prior that can robustly model the dependent measurements. To accurately estimate the object trajectory, we form the optimal hypothesis test to discard the noise measurements. We then employ a Bayesian tracker to track a single object.

### *6.1.1 Measurement Model for Dependent Observations*

We consider a single object tracking problem where the measurements are assumed to receive from multiple sensors scanning the same scene without having any knowledge of observation to sensor associations. It is worth mention that the number of observations collected by each sensor may vary with time. Hence, the multimodal system observations are statistically dependent. As the dependent observations in the object tracking model are collected from $M$ disparate sensors, they can correspond to different measurement models. The state-space model from the system dynamics and measurements for estimating the object state parameter vector $\mathbf{x}_k$ is thus given by

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}) + \mathbf{u}_k \tag{6.1}$$

$$\mathbf{Z}_{m,k} = h_m(\mathbf{x}_k) + \mathbf{w}_{m,k}\,. \tag{6.2}$$

where $\mathbf{u}_k$ is a transition error random process, $\mathbf{w}_{m,k}$ is the additive measurement noise process from the $m$th sensor, and $\mathbf{Z}_{m,k} = \{\mathbf{z}_{m,k}^{(1)}, \ldots, \mathbf{z}_{m,k}^{(L_m)}\}$, is the collection of $L_m$ measurements received by the $m$th sensor. The function $h_m(\mathbf{x}_k)$ is a time-varying and possibly nonlinear function that describes the relation between the object state and the measurement set $\mathbf{Z}_{m,k}$ from the $m$th sensor.

We assume that the $m$th sensor generates the measurement set $\mathbf{Z}_{m,k}$ according to the likelihood function $p(\mathbf{Z}_{m,k}|\mathbf{x}_k)$. In Equation (6.1), the object state $\mathbf{x}_k$ is assumed to evolve from time $(k-1)$ following the possibly nonlinear transition function $f(\mathbf{x}_{k-1})$, and thus according to a transition probability kernel $\mathbb{Q}_{\boldsymbol{\theta}_k}(\mathbf{x}_{k-1}, \mathbf{x}_k)$. We assume the observations are dependent; meaning it is assumed that both $\mathbf{Z}_{m,k}$ and $\mathbf{Z}_{n,k}$, $m \neq n$ as well as $\mathbf{z}_{m,k}^{(i)}$ and $\mathbf{z}_{m,k}^{(j)}$, $i \neq j$, for a fixed sensor $m$, are both correlated.

### 6.1.2 Measurement Associations and Prior

As discussed in Chapter 2, hierarchical Bayesian models can be utilized to capture the dependency among measurements that may have originated from different sensors [54]. In particular, the hierarchical Dirichlet process (HDP) framework can be exploited to model dependency among measurements that are related to clusters which are shared among all groups (sensors). The object trajectory can be more accurately estimated once the sensor measurement association is determined while accounting for statistical dependency. To this end, we place a prior on the collection of measurements to capture the dependency among them and provide a prior measurement distribution. Following the HDP model, each sensor parameter is drawn from a discrete random probability measure with probability one to ensure the dependency among measurements. A graphical representation of the HDP mixture model, as described next, is depicted in Figure 6.1.

The measurement parameters are drawn from a shared Dirichlet process $\mathrm{DP}(\gamma, G_0)$

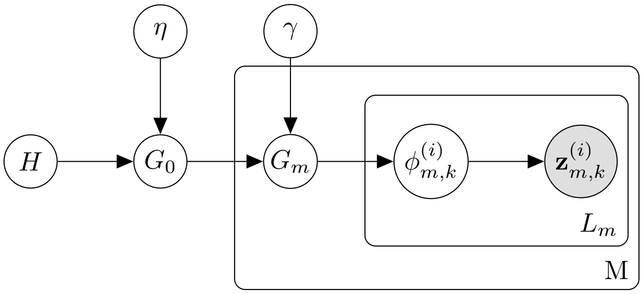


Figure 6.1: Graphical Representation of the HDP Mixture Model.

with concentration hyperparameter $\gamma$ and base distribution $G_0$; this base distribution is drawn from another Dirichlet process with concentration parameter $\eta$ and base distribution $H$. We assign a random probability measure $G_m$, drawn from a discrete random probability measure $G_0$, for the measurements of the $m$th sensor, $m=1,\dots,M$. We assume that the parameters $\phi_{m,k}^{(i)}$ of the $i$th measurements from the $m$th sensor at time $k$ are drawn from $G_m$, that is, $\phi_{m,k}^{(i)}|G_m \sim G_m$, $i=1,\dots,L_m$. This is needed in order to place a prior on the dependent measurements that originated from the same sensor such that the same structure is inherited within sensor measurements. The resulting model needs to both capture the dependency among the measurement sets and the identity of the sensor measurement model as in Equation (6.2). This would not have been achieved if a Dirichlet process prior was placed on all the measurements or if independent random probability measures $G_m$ were drawn for each measurement set. As shown in [54], if a measure $G_m$, given the distribution $G_0$, is drawn independently from $G_0$, then dependency within each measurement set and among sensor measurements are captured as they share the same parameters. We assume that the distribution $G_0$ is a global random probability measure that cannot be continuous and that $G_m$, $m = 1,\dots,M$ are conditionally independent given $G_0$; $G_m$, $m = 1,\dots,M$ are drawn from a Dirichlet process with base measure $G_0$ and concentration parameter $\eta$. Therefore, the parameters associated with each sensor,

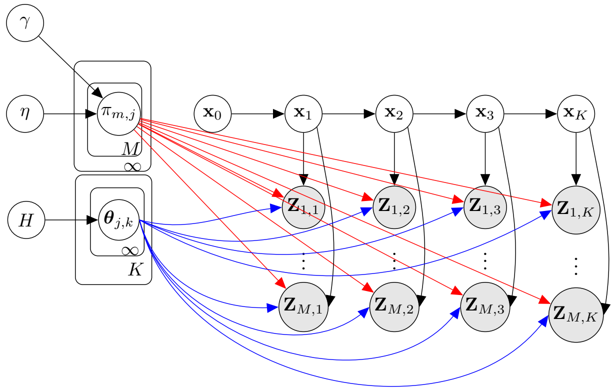


Figure 6.2: Graphical Model Capturing the Temporal Dependence among the Measurements. Note That $\boldsymbol{\theta}_{j,k}$ Correspond to Parameters at Time $k$ Which Are Shared among All The Groups of Measurements Received from The Sensors.

measurement are drawn from a Dirichlet process.

By placing the HDP prior on the measurement parameters collected from the $m$th sensor, the distribution of the measurements can be modeled as

$$\begin{aligned}
&G_0 \sim \mathrm{DP}(\eta, H) \\
&G_m \mid G_0 \sim \mathrm{DP}(\gamma, G_0),\ \ m = 1, \dots M_k \\
&\phi^{(i)}_{m,k} \mid G_m \sim G_m,\ \ i = 1, \dots, L_{m,k} \\
&\mathbf{z}^{(i)}_{m,k} \mid \phi^{(i)}_{m,k} \sim F\left(\phi^{(i)}_{m,k}\right)
\end{aligned} \tag{6.3}$$

for some distribution $F(\cdot)$ that captures the physical model. This method clusters the measurements that are collected by each sensor and estimates the joint density of the dependent measurements. As shown next, this density is used to infer the object trajectory. Note that although we assume that the total number of sensors is fixed, our approach can be generalized to a time-varying number of sensors.

### *6.1.3 Bayesian Inference*

Once the HDP mixture model provides an estimate of the measurement density and clusters the measurements, we use Bayesian tracking methods to infer the object trajectory. We refer to the overall approach as HDP-DM (HDP for dependent measurements). The graphical representation of the HDP-DM is provided in Figure 6.2; the approach is summarized in Algorithm 12.

**Hypothesis Testing for Object Detection**

We assume that $\mathbf{Z}_k = \{\mathbf{Z}_{1,k},\ \ldots,\ \mathbf{Z}_{M,k}\}$ is the set of measurements from all $M$ sensors at time step $k$. It is also assumed that the measurements at time step $k$ depend only on the object state at the same time step. Specifically,

$$P\Big(\mathbf{Z}_1, \ldots, \mathbf{Z}_k | \mathbf{x}_1,\ \ldots\ \mathbf{x}_k\Big) = \prod_{j=1}^{k} P(\mathbf{Z}_j | \mathbf{x}_j). \tag{6.4}$$

The received observations may not always include object information. As a result, before estimating the object state parameter vector $\mathbf{x}_k$ using the $m$th sensor observations $\mathbf{Z}_{m,k}$, $m = 1, \ldots, M$, a detection test statistic must be formed based on the binary hypothesis

$$\mathcal{H}_0 : \mathbf{Z}_{m,k} = \mathbf{w}_{m,k}$$
$$\mathcal{H}_1 : \mathbf{Z}_{m,k} = h_m(\mathbf{x}_k) + \mathbf{w}_{m,k}\ .$$

where, $\mathbf{w}_{m,k}$ is the noise vector at time $k$ and $h_m(\cdot)$ is the measurement model corresponding to the $m$th sensor in Equation (6.2). The Neyman-Pearson detection test statistic $\mathcal{T}_m(\cdot)$ is selected to maximize the probability of detection for a given probability of false alarm. Thus, we decide that the object is detected using the measurements from the $m$th sensor if the test statistic exceeds a threshold value obtained from the given probability of false alarm. The test statistic is given by

$$\mathcal{T}_m\Big(\mathbf{Z}_{m,k}, \boldsymbol{\phi}_{m,k}; \mathbf{x}_k\Big) = \frac{p\Big(\mathbf{Z}_{m,k} \mid \mathbf{x}_k; \mathcal{H}_1\Big)}{p\Big(\mathbf{Z}_{m,k}; \mathcal{H}_0\Big)}\,, \tag{6.5}$$

where $\boldsymbol{\phi}_{m,k} = \big\{\phi^{(i)}_{m,k}, \ldots, \phi^{(L_m)}_{m,k}\big\}$. Note that we assume that all sensor measurements are dependent, including measurements from the same sensor. As a special case, if the measurements from the same sensor were to be assumed independent, the likelihood ratio in Section 6.1.3 would simplify to

$$\mathcal{T}_m\Big(\mathbf{Z}_{m,k}, \boldsymbol{\phi}_{m,k}; \mathbf{x}_k\Big) = \frac{\prod_{i=1}^{L_m} p\Big(\mathbf{z}^{(i)}_{m,k} \mid \mathbf{x}_k; \mathcal{H}_1\Big)}{\prod_{i=1}^{L_m} p\Big(\mathbf{z}^{(i)}_{m,k}; \mathcal{H}_0\Big)}\,. \tag{6.6}$$

Note that the formulation in Section 6.1.3 for this special case does not contradict the dependency among measurements from different sensor as $\mathbf{Z}_{m,k}$ and $\mathbf{Z}_{n,k}$, $m \neq n$ are still correlated.

### Bayesian Object Tracking Method

We assume that $\mathcal{Z}_{m,k} \subset \mathbf{Z}_k$ is the set of measurements from the $m$th sensor that originated from the object, and that $\mathcal{Z}_k = \{\mathcal{Z}_{1,k}, \ldots, \mathcal{Z}_{M,k}\}$ is the set of all measurements that originated from the object. Then, the object state density $p(\mathbf{x}_k|\mathcal{Z}_k)$ summarizes all information about the history of the object up to time $k$. The estimated state is obtained as the posterior mean given by

$$\hat{\mathbf{x}}_k = \mathbb{E}\big[p(\mathbf{x}_k|\mathcal{Z}_k)\big]. \tag{6.7}$$

The posterior density can be computed recursively for all $k \geq 1$. Assuming an initial probability, the state probability at time $k$ must be predicted using all the sensor measurements up to time $(k-1)$. The tail recursive function for the prediction is given by

$$p\Big(\mathbf{x}_k|\mathcal{Z}_1, \ldots, \mathcal{Z}_{k-1}\Big) = \int \mathbb{Q}_{\boldsymbol{\theta}_k}\Big(\mathbf{x}_{k-1}, \mathbf{x}_k\Big)\, p\big(\mathbf{x}_{k-1}|\mathcal{Z}_1, \ldots, \mathcal{Z}_{k-1}\big) d\mathbf{x}_{k-1}, \tag{6.8}$$

where $\mathbb{Q}_{\boldsymbol{\theta}_k}\Big(\mathbf{x}_{k-1}, \mathbf{x}_k\Big)$ is the transition probability kernel and $\boldsymbol{\theta}_k = \{\phi_{1,k}, \phi_{2,k}, \dots\}$. We use forwards recursion to obtain the filtering distribution, which is the distribution of the state at time $k$ conditioned on the measurements history up to time $k$. Specifically, at time step $k$, the Bayesian recursion is given by

$$p(\mathbf{x}_k|\mathcal{Z}_1,\ \dots,\ \mathcal{Z}_k) \propto p(\mathcal{Z}_k|\mathbf{x}_k)\, p(\mathbf{x}_k|\mathcal{Z}_1, \dots, \mathcal{Z}_{k-1}). \tag{6.9}$$

To compute this probability, we use the tail recursive Equation (Equation (6.8)) and the density of $\mathcal{Z}_k$ estimated using the HDP mixture in Section 6.1.2. That is, the distribution of $\mathcal{Z}_{m,k}$, for any $m$, conditioned on $\mathbf{x}_k$, is obtained as

$$P(\mathcal{Z}_{m,k}|\mathbf{x}_k) = \sum_{j=1}^{\infty} \pi_{m,j}\, F(\boldsymbol{\theta}_{j,k}), \tag{6.10}$$

for some distribution $F$ that is chosen to describe the physical model. Here, $\boldsymbol{\theta}_{j,k} \sim H$ for a base distribution $H$ and for hyperparameters $\eta$ and $\gamma$. The parameters $\pi_{m,j}$ follow from $\pi_m = (\pi_{m,1}, \pi_{m,2}, \ldots)$, where $\pi_m \sim \mathrm{DP}(\eta, \mathrm{GEM}(\gamma))$, $\mathrm{GEM}(\gamma)$ is defined as

$$\begin{aligned} \pi'_{m,j} &\sim \mathrm{Beta}(1, \gamma) \\ \pi_{m,j} &= \pi'_{m,j} \prod_{\ell=1}^{j-1} (1 - \pi'_{m,\ell}) \end{aligned} \tag{6.11}$$

and $\mathrm{Beta}(1, \gamma)$ is the Beta distribution. Note that the dependency among sensor measurements comes from the fact that $\boldsymbol{\phi}_{m,\ell}|G_m \sim G_m$ are shared among all the sensors (groups) [120, 121].

## 6.2 Multi Sensor Dependent Observations: Multi-Object Tracking

In this section, we generalize the problem discussed in Section 6.1 to track multiple objects with unknown cardinality from measurements collected by multiple sensors. We develop robust algorithms that fully capture the dependency among the measurements as well as being capable of dealing with unknown time-dependent objects and

**Algorithm 12:** HDP-DM Approach for Computing the HDP Prior via Dependent Measurements and Estimating the Object State.

```
**Input:** $\eta$, $\gamma$, $H$ and $\mathbf{Z}_{1,k},\ \ldots,\ \mathbf{Z}_{M,k}$
Draw a $G_0$ from a DP$(\eta, H)$
for $m = 1$ to $M$ do
    Draw $G_m \mid G_0 \sim$ DP$(\gamma, G_0)$
end for
for $m=1$ to $M$ do
    for $i=1$ to $L_m$ do
        Draw $\phi_{m,k}^{(i)} \mid G_m \sim G_m$
    end for
end for
Draw each measurement $\mathbf{z}_{m,k}^{(i)}$ from the probability distribution $F\big(\phi_{m,k}^{(i)}\big)$
for  $m=1$ to $M$ do
    Compute the likelihood $\mathcal{T}(\mathbf{Z}_{m,k}, \boldsymbol{\phi}_{m,k}; \mathbf{x}_k)$ as in Equation (6.4)
end for
Return: Object generated measurements
    $\mathcal{Z}_k = \{\mathcal{Z}_{1,k}, \ldots, \mathcal{Z}_{M,k}\}$
Sample from $p(\mathcal{Z}_{m,k}|\mathbf{x}_k)$ using an MCMC method
Prediction: Compute $p(\mathbf{x}_k|\mathcal{Z}_1, \ldots, \mathcal{Z}_{k-1})$ from Equation (6.8)
Update: Draw $\mathbf{x}_k$ from $p(\mathbf{x}_k|\mathcal{Z}_1, \ldots, \mathcal{Z}_k)$ from Equation (6.9)
Return: $\hat{\mathbf{x}}_k$ using Equation (6.7)
```

their identity. Additionally, given the dependent observations received from multiple sensors, our model takes advantage of the additional information provided by dependency among the measurements to improve the tracking performance. We integrate a dependent Dirichlet process as a prior on the time-varying object state distributions with a hierarchical Dirichlet process mixture as a model to capture the dependency among the measurements to accurately and robustly estimate the evolving object cardinality and their trajectory. We demonstrate through simulations that providing multimodal dependent measurements the proposed method can improve the accuracy of the object trajectory estimation and can robustly determine the time-dependent cardinality.

### 6.2.1 Problem Formulation

We consider multiple object tracking where a time-varying number of sensors collect measurements. Assume that $\mathbf{X}_k = \{\mathbf{x}_{1,k}, \ldots, \mathbf{x}_{N_k,k}\}$ is the collection of the object states at time $k$ for an unknown variable $N_k$. Each object at time $(k-1)$, $\mathbf{x}_{\ell,k-1}$, may leave the scene with probability $1 - \mathrm{P}_{k|k-1}$ or may stay in the field of view with probability $\mathrm{P}_{k|k-1}$ and transition to state $\mathbf{x}_{\ell,k}$ at time $k$ according to the transition kernel probability $\mathbb{Q}_{\boldsymbol{\theta}_{\ell,k}}(\mathbf{x}_{\ell,k-1}, \mathbf{x}_{\ell,k})$, given unknown parameters $\boldsymbol{\theta}_{\ell,k}$. At each time step, a time-dependent number of new objects may also enter scene. We aim to jointly estimate the number of objects as well as the trajectory of each object using measurements. Suppose $L_k$ sensors collect information of the scene at time $k$. Each sensor collects an unordered measurement set $\mathbf{Z}_{m,k} = \{\mathbf{z}^1_{m,k}, \ldots, \mathbf{z}^{M_k}_{m,k}\}$, $m = 1, \ldots, L_k$. We define $\mathbf{Z}_k = \{\mathbf{Z}_{1,k}, \ldots, \mathbf{Z}_{L_k,k}\}$ to be the set of all measurements collected by $L_k$ sensors such that $\mathbf{Z}_{m,k}$ and $\mathbf{Z}_{n,k}$ are highly correlated for $n \neq m$. We employ Bayesian nonparametric modeling to use the dependency among measurements to improve the tracking a time-varying number of objects. To this end, we place a DDP-based prior on the object state parameters and a hierarchical Dirichlet process prior on the measurements. This modeling not only takes the time-dependency among the objects but also takes advantage of dependency among the measurements received by multiple sensors. In the next section, we provide the prior model as well as the inference model in detail.

### *6.2.2 Prior Construction*

Assume that $\Theta_k$ is the collection of all parameters associated with tracking at time $k$ and $\theta_{\ell,k} \in \Theta_k$ is the $\ell$th parameter at time $k$. Let $D_{k|k-1}$ and $D_k$ be the number of parameters transitioning from time $(k-1)$ to $k$ using transition kernel $\nu(\theta_{\ell,k-1}, \cdot)$ and the number of parameters at time $k$, respectively. We define $V_k$ to be $(1 \times D_k)$-vector where the $\ell$th element, $[V_k]_\ell$, represents the number of states associated with the $\ell$th parameter. One can similarly define $V_{k|k-1}$ to be the vector consisting of the number of survived and then transitioned objects from time $(k-1)$ to $k$. Note that entires of these vectors may be zero because some objects may leave, and thus no state is associated with the corresponding parameter. We similarly define vector $V^*_{k|k-1} \in \mathbb{R}^{D^*_{k|k-1}}$ from $V_{k|k-1}$ by eliminating zero entires. We construct the DDP prior on the object state parameters as follows:

**Case 1:** The $\ell$th object belongs to one of the survived and already transitioned clusters where has not yet been assigned to any object at time $(k-1)$ The object selects such a cluster with probability:

$$\Pi_1(\text{Select } j\text{th unassigned cluster}|\boldsymbol{\theta}^{\ell-1}_{1,k}, \Theta_{k|k-1}) = \frac{\left[V^*_{k|k-1}\right]_j}{\sum_j \left[V^*_{k|k-1}\right]_j + \sum_j [V_k]_j + \alpha} \quad (6.12)$$

for hyperparameter $\alpha$.

**Case 2:** The $\ell$th object selects one the survived clusters which has already been occupied by the previous objects. The object belongs to one of these clusters with probability:

$$\Pi_2(\text{Select } j\text{th assigned cluster}|\boldsymbol{\theta}_{1,k}^{\ell-1}, \Theta_{k|k-1}) = \frac{\left[V^*_{k|k-1}\right]_j + [V_k]_j}{\sum\limits_j \left[V^*_{k|k-1}\right]_j + \sum\limits_j [V_k]_j + \alpha} \tag{6.13}$$

**Case 3:** The $\ell$th object does not belong to any of the transitioned clusters. We initiate such a cluster with probability:

$$\Pi_3(\text{Create new cluster}|\boldsymbol{\theta}_{1,k}^{\ell-1}, \Theta_{k|k-1}) = \frac{\alpha}{\sum\limits_j \left[V^*_{k|k-1}\right]_j + \sum\limits_j [V_k]_j + \alpha} \tag{6.14}$$

Given the cases (1)-(3), the state distribution $p(\mathbf{x}_{\ell,k}|\mathbb{X}_k^{\ell-1}, \mathbb{X}^\star_{k|k-1}, \Theta^\star_{k|k-1}, \Theta_k)$ is given by:

$$\begin{cases} \mathbb{Q}_{\boldsymbol{\theta}_k}(\mathbf{x}_{\ell,k}|\mathbf{x}_{\ell,k-1}) f(\mathbf{x}_{\ell,k}|\boldsymbol{\theta}_{\ell,k}) & \text{If case 1 happens} \\ \mathbb{Q}_{\boldsymbol{\theta}_k}(\mathbf{x}_{\ell,k}|\mathbf{x}_{\ell,k-1}) \nu(\boldsymbol{\theta}^*_{\ell,k-1}, \boldsymbol{\theta}_{\ell,k}) f(\mathbf{x}_{\ell,k}|\boldsymbol{\theta}_{\ell,k}) & \text{If case 2 happens} \\ \int_{\boldsymbol{\theta}} f(\mathbf{x}_{\ell,k}|\boldsymbol{\theta}) dH(\boldsymbol{\theta}) & \text{If case 3 happens} \end{cases} \tag{6.15}$$

for some density $f$ and a Dirichlet process with concentration parameter $\alpha$, and base distribution $H$, $\mathrm{DP}(\alpha, H)$. Note that $\Theta^*_{k-1} = \{\boldsymbol{\theta}^*_{\ell,k-1}\}_\ell^{D^*_{k|k-1}} \subset \Theta_{k-1}$ and $\mathbb{X}_k^{\ell-1}$ represent the set of unique parameters at time $k$ and the configuration at time $k$ up to the $\ell$th object.

### *6.2.3 Inference Model*

Upon receiving the measurements from the sensors, it is crucial to capture the dependency among the unordered sensor measurements. We need to partition the received measurements to use the dependency among them to robustly track the objects. We employ a HDP prior on the measurements parameters. The HDP mixture model allows us to model the measurements corresponding to various objects and also the dependency among the multiple sensor measurements. Each group of measurements in such a formulation corresponds to a sensor and multiple clusters within each group is associated with multiple objects. We utilize the proposed HDP as prior on the measurements to track the objects. What follows briefly describes this procedure. Assume the base measurement $G_0$ is a discrete random measure that is drawn from $\text{DP}(\eta, H_0)$, then

$$\begin{aligned}
&G_0 \sim \text{DP}(\eta, H),\\
&G_m | G_0 \sim \text{DP}(\gamma, G_0), \qquad m = 1, \cdots, L\\
&\boldsymbol{\phi}_{m,k}^{(j)} | G_m \sim G_m, \qquad j = 1, \ldots, L_{m,k}\\
&\mathbf{z}_{m,k}^{(j)} | \boldsymbol{\phi}_{m,k}^{(j)}, \mathbb{X}_k \sim R(\cdot | \boldsymbol{\phi}_{m,k}^{(j)}, \mathbf{x}_{\ell,k})
\end{aligned} \tag{6.16}$$

for some distribution $R$. Given Equation Equation (6.16) and the proposed DDP prior, one can compute the posterior distribution and thus track the multiple objects.

## 6.3 Simulations

### *6.3.1 HDP-DM Single Object Tracking: Synthetic Gaussian Data*

The performance of the proposed HDP-DM algorithm is demonstrated using simulations to track a single object given two dependent (and different) measurements. The object state vector is given by $\mathbf{x}_k = [x_k \; y_k \; \dot{x}_k \; \dot{y}_k]^T$, where $(x_k, y_k)$ and $(\dot{x}_k, \dot{y}_k)$ are the two-dimensional Cartesian coordinates of the position and velocity of the object, respectively, at time step $k$. For this example, the state transition in Equation (6.1) is given by the linear model

$$\mathbf{x}_k = F\mathbf{x}_{k-1} + \mathbf{u}_k \tag{6.17}$$

where

$$F = \begin{bmatrix} 1 & \Delta t & 0 & -2\Delta t \\ 0 & 1 & 0 & -(\Delta t)^3/3 \\ 0 & 2\Delta t & 1 & (\Delta t)^2/2 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{6.18}$$

where $\Delta t$ is the time between time steps, and $\mathbf{u}_k$ is a $(4\times1)$ zero-mean Gaussian random vector with covariance $C_\mathbf{u}$. The two measurement vectors $\mathbf{Z}_{m,k}$, $m=1,2$ are given by

$$\mathbf{Z}_{1,k} = h_1(\mathbf{x}_k) + \mathbf{w}_{1,k} \tag{6.19}$$

$$\mathbf{Z}_{2,k} = h_2(\mathbf{x}_k) + \mathbf{w}_{2,k} \tag{6.20}$$

where

$$h_1(\mathbf{x}_k) = \begin{bmatrix} \sqrt{(x_k^2 + y_k^2)} \\ x_k \\ 0 \end{bmatrix}, \quad h_2(\mathbf{x}_k) = \begin{bmatrix} \sqrt{(x_k^2 + y_k^2)} \\ 0 \\ y_k \end{bmatrix} \tag{6.21}$$

and $\mathbf{w}_k = [\mathbf{w}_{1,k}^T \ \mathbf{w}_{2,k}^T]^T$ is a $(6\times1)$ zero-mean Gaussian random vector with covariance matrix $C_\mathbf{w}$. Note that $\mathbf{u}_k$ and $\mathbf{w}_k$ are assumed to be mutually independent.

For this simulation, we set $\Delta t = 1$, $C_\mathbf{u} = 50\,\mathrm{I}_4$, and

$$C_\mathbf{w} = 10^5 \begin{bmatrix} 2\,\mathrm{I}_3 & 3\,\mathrm{I}_3 \\ 3\,\mathrm{I}_3 & 5\,\mathrm{I}_3 \end{bmatrix}, \tag{6.22}$$

where $\mathrm{I}_N$ is the $(N\times N)$ identity matrix.

For the HDP prior, the base distribution $H$ in Section 6.1.2 was selected to be a normal-inverse-Wishart distribution, $\mathcal{NIW}(\mu_0, \lambda, \nu, \mathbf{\Psi})$, with values $\mu_0 = 0$, $\lambda = 0.05$, $\nu = 100$, and $\mathbf{\Psi}$ equal to the identify matrix. The concentration parameters $\eta$ and

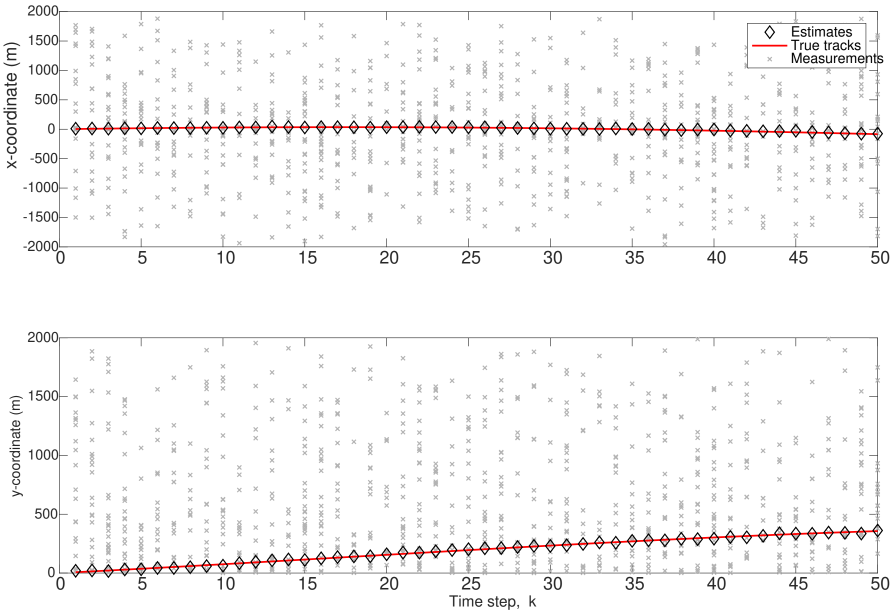


Figure 6.3: Actual and Estimated $x$-coordinate (Top) and $y$-coordinate (Bottom) of the Target Position Using Bayesian Tracking with HDP-DM.

$\gamma$ is drawn as independent and identically distributed from the Gamma distribution $\Gamma(\cdot\,; 1, 0.2)$.

For comparison, we simulate the HDP-DM-based Bayesian tracker and a Bayesian tracker that assumes that the two measurements are independent (BT-IM). The simulation results are obtained using 10,000 Monte Carlo runs. Figure 6.3 shows the estimated $x$ and $y$ coordinates obtained using the HDP-DM. Figure 6.4 (top) and Figure 6.4 (bottom) show the estimated range of the object obtained using the HDP-DM and the BT-IM, respectively. The corresponding mean-squared error (MSE) for each of the two approaches, obtained by averaging 100 measurement realizations, is shown in Figure 6.5. As it can be observed, even for this simple example of only two measurements, the dependency among the measurements results in the performance improvement. Figure 6.5 depicts that the MSE is reduced when the dependency of

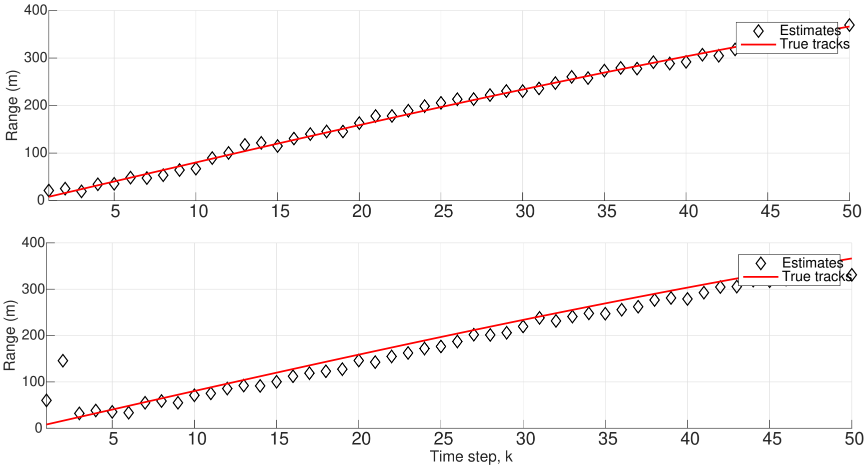


Figure 6.4: Range Estimation Using Bayesian Tracking with HDP-DM (Top) and Only Bayesian Tracking (Bottom).

the measurements is taken into consideration by the HDP-DM. The cardinality of the measurements is shown in Figure 6.6. For this example, there were two measurements at each time step.

### *6.3.2 HDP-DM Single Object Tracking: Waveform-Agile Multi Modal Data*

In this section, we apply the HDP-DM algorithm to waveform-agile multi modal data introduced in [122]. In this model, we apply the HDP algorithm on more realistic data models such as for radio frequency (RF) and electro optical (EO) sensors. We use the following nearly constant velocity motion model with state $\mathbf{x}_k = [x_k \; \dot{x}_k \; y_k \; \dot{y}_k]^T$ at time $k$ where $(x, y)$ and $(\dot{x}, \dot{y})$ are the location and the velocity, respectively. The state space representation $\mathbf{x}_k$ is given by

$$\mathbf{x}_k = F\mathbf{x}_{k-1} + \mathbf{u}_k \tag{6.23}$$

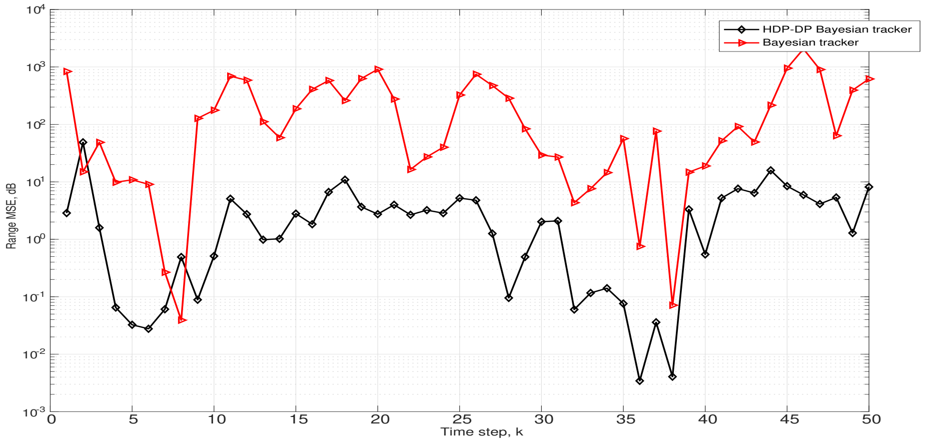


Figure 6.5: MSE for the Estimated Range of the Object for the Bayesian Tracking HDP-DM Approach and the Classical Bayesian Tracking Approach.

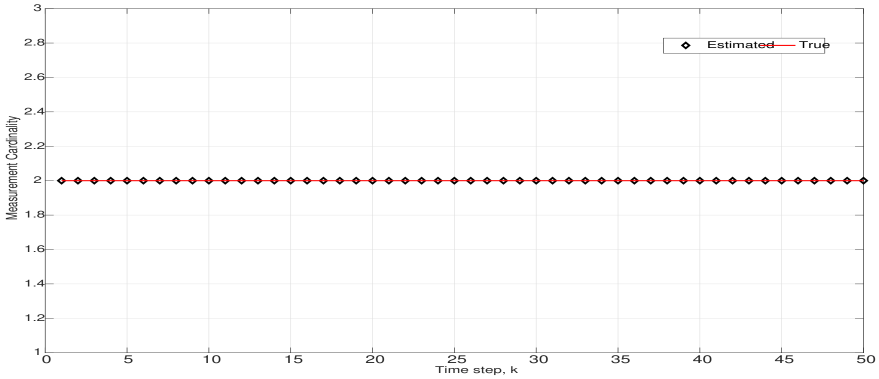


Figure 6.6: Time-varying Cardinality of the Measurements at Each Time Step.

where $\mathbf{u}_k$ is the transition error (noise) and $F$ is given by

$$F = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{6.24}$$

The noise vector has a Gaussian distribution $\mathcal{N}(0, Q_k)$ where the covariance matrix $Q_k$ is

$$Q_k = \begin{bmatrix} \Delta t^3/3 & 0 & \Delta t^2/2 & 0 \\ \Delta t^2/2 & 0 & \Delta t & 0 \\ 0 & \Delta t^3/3 & 0 & \Delta t^2/2 \\ 0 & \Delta t^2/2 & 0 & \Delta t \end{bmatrix} \tag{6.25}$$

where $\Delta t = 1$. For an RF-EO sensor measurements, the received RF and EO sensor signals are preprocessed to determine the presence of the target. The resulting range and range-rate estimates are used as a single measurement for tracking. Under low SNR environments, however, the probability of detection is low and a single such measurement cannot be accurately obtained. The sensor measurements model is based on the model in [122] involve two types of measurements:

**(i) Radio Frequency Sensor Measurement Model For the Signal**

For the radar measurement, divide a range-Doppler plane into $A \times B$ resolution cells and assume that each cell provides a matched filter output amplitude. The target contribution to the intensity is the ambiguity function, $AF_s(\tau, \nu)$, of the transmitted signal $s(t)$ as a function of the range, $r$, and range-rate, $\dot{r}$, of the target. Assuming $s(t) = (\frac{1}{\pi \Delta t^2})^{1/4} e^{-\frac{t^2}{2\Delta t^2}} e^{ibt^2}$, the ambiguity function is

$$AF_s(\tau, \nu) = \exp\left(\frac{-\tau^2}{4\Delta t^2} + \pi \Delta t^2 (\nu + \frac{b\tau}{\pi})^2\right) \exp(i\pi\tau\nu). \tag{6.26}$$

Assuming a Gaussian measurement noise with mean zero and variance $\sigma^2_{RF}$, the radar measurements for cell $(a, b)$ corresponding to a rectangle centered at $(r_a, \dot{r}_a)$ for $a = 1, 2, \ldots, A$ and $b = 1, 2, \ldots, B$ is

$$z^1_{k,(a,b)} = g_{k,(a,b)}(\mathbf{x}_k) + \mathbf{w}_{1,k,(a,b)} \tag{6.27}$$

where $g_{k,(a,b)}$ follows

$$g_{k,(a,b)}(\mathbf{x}_k) = I_{RF} AF_s(\frac{r_a - r_k}{2c}, \frac{2f_c \dot{r}_b - \dot{r}_k}{c}) \tag{6.28}$$

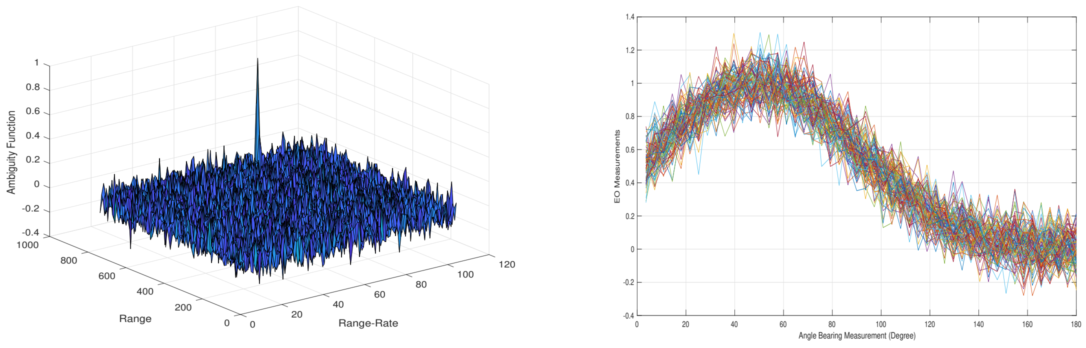


Figure 6.7: (a) RF Measurements with Gaussian Noise. (b) EO Measurements with Gaussian Noise.

for $r_k = \sqrt{x_k^2 + y_k^2}$, $\dot{r}_k = \frac{x_k \dot{x}_k + y_k \dot{y}_k}{r_k}$ defining the range and range-rate at time $k$, respectively. A realization of RF measurement is depicted in Figure 6.7a.

**(ii) EO Sensor Measurement Model**

The EO sensor $1 - D$ angle bearing measurement plane is divided into $C$ cells with center $\phi_c$ for $c = 1, \ldots, C$ at time $k$. The measurement obtained at the center of cell $c$ is given by

$$z_{k,c}^2 = h_{k,c}(\mathbf{x}_k) + \mathbf{w}_{2,k,c} \tag{6.29}$$

where the measurement noise is a Gaussian with mean zero and variance $\sigma_{EO}^2$. The target contribution to the intensity level at cell $c$ equals to

$$h_{k,c}(\mathbf{x}_k) = I_{EO} \frac{2}{\sqrt{2\pi\sigma_{EO}^2}} \exp - \frac{(\phi_c - \phi_k)^2}{2\sigma_{EO}^2} \tag{6.30}$$

with $\phi_k = \tan^{-1}(\frac{y_k}{x_k})$. A realization of the collected EO sensor measurement is depicted in Figure 6.7b. For this model, we simulated the HDP-DM-based Bayesian tracker and a Bayesian tracker (The Bayesian tracker assumes that the two measurements are independent.). Using the aforementioned model as Section 6.3.1, we estimated the target trajectory using both the HDP-DM and the BT-IM methods. Figure 6.8 shows the estimated $x$ and $y$ coordinates obtained using the HDP-DM. The corresponding location estimate is depicted in Figure 6.9.

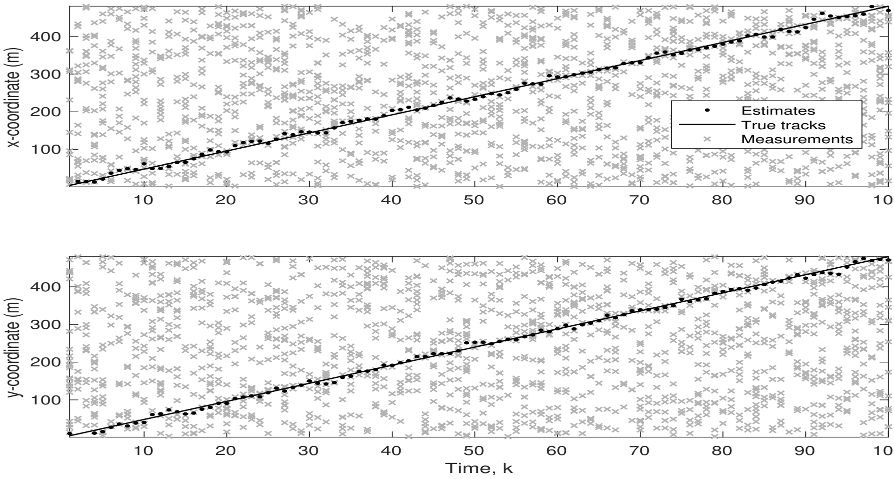


Figure 6.8: $x$ and $y$ Location True and Estimated Using HDP-DM Algorithm.

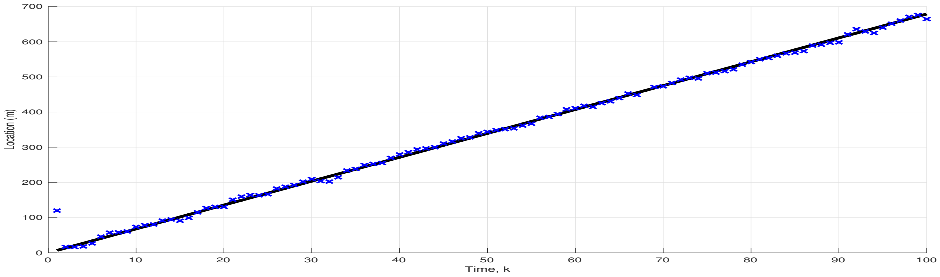


Figure 6.9: Location Estimated by HDP-DM Algorithm.

The mean-squared error (MSE) for each of the estimated target location obtained for both the HDP-DM and BT-IM methods, acquired by averaging over 1000 measurement realizations, is shown in Figure 6.10. It is observed that the MSE is reduced when the dependency of the measurements is taken into consideration by the HDP-DM.

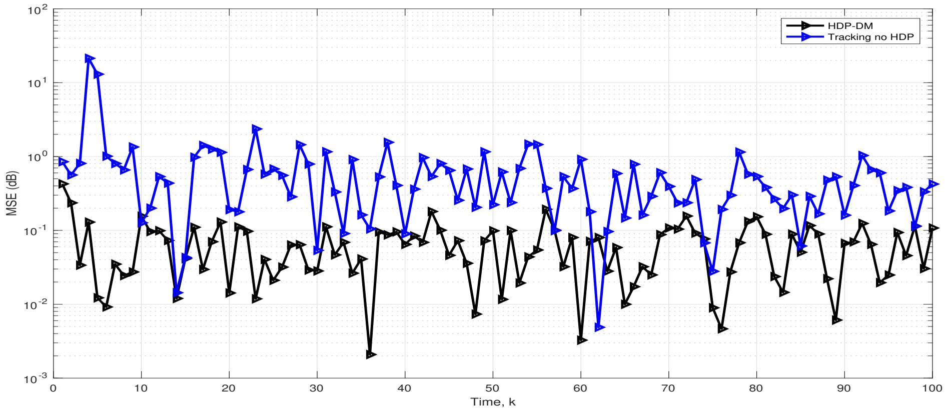


Figure 6.10: MSE Comparison between HDP-DM and A Bayesian Tracker.

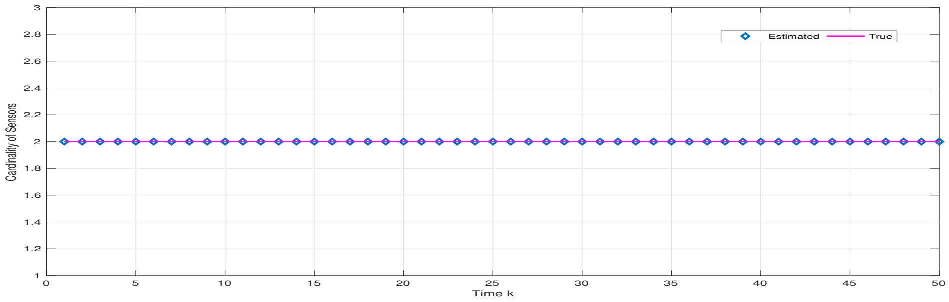


Figure 6.11: True and Estimated Sensor Cardinality.

The cardinality of the measurements is shown in Figure 6.11. For this example, there are $RF$ and $EO$ sensor measurements at each time step. To illustrate the performance of this nonparametric approach, we also compare this method for various SNRs and display it in Figure 6.12. Note that in this example we assume the correlation coefficient between the $RF$ and $EO$ to be $\rho = 0.5$.

### *6.3.3 Multi Object Tracking: Multi Sensor Dependent Measurements*

In this section we use the same state model as Section 3.5.1 for five objects. In addition to the generated measurements in Section 3.5.1, we also generate $\mathbf{z}^2_{l,k} =$

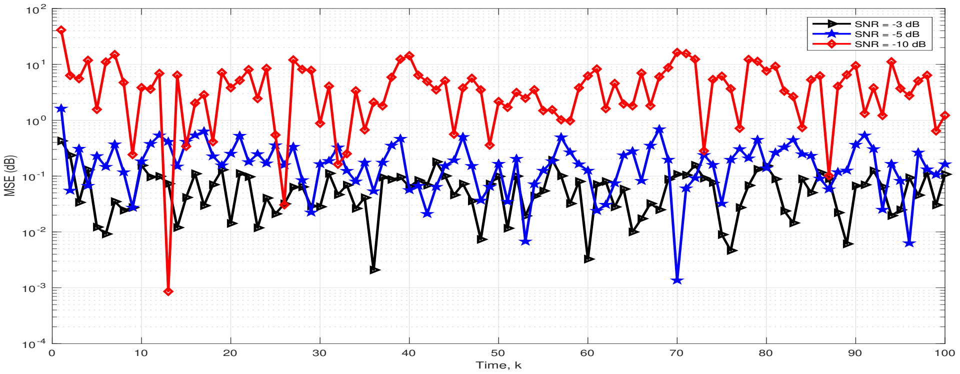


Figure 6.12: MSE Error under Different SNRs When Using HDP-DM Algorithm for *RF* and *EO* Measurements.

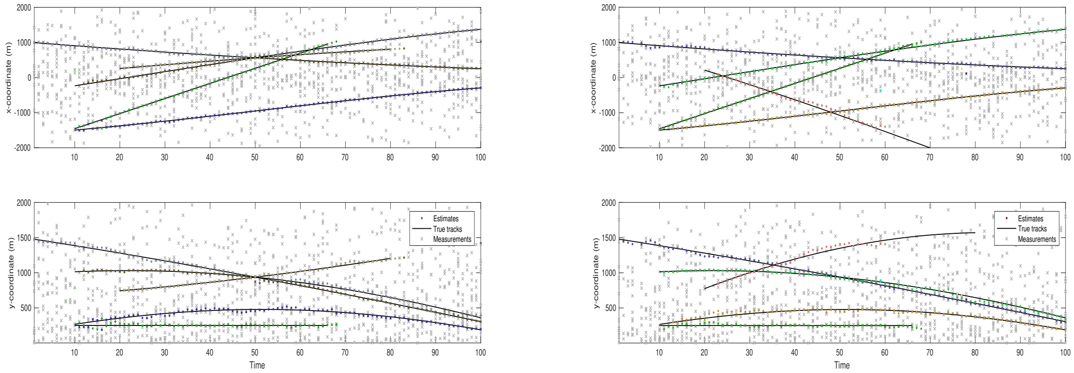


Figure 6.13: (a) Actual and Estimated $x$ and $y$ Positions Incorporating Dependent Measurements Using HDP with DDP-EMM(b) Actual and Estimated $x$ and $y$ Positions through DDP-EMM.

$\frac{1}{\sqrt{2\pi\sigma_l^2}} \exp \frac{\phi_{l,k}^2}{2\sigma_l^2}$, where $\phi_{l,k} = tan^{-1}(y_{l,k}/x_{l,k})$. This model is compared to that of assuming independent measurements through DDP-EMM. The compassion results indicate, under the same circumstances, that dependent measurement assumption improves the performance of the DDP-EMM tracker. Our results are obtained using 10,000 Monte Carlo runs, SNR = -5 dB, and OSPA measure parameters of order

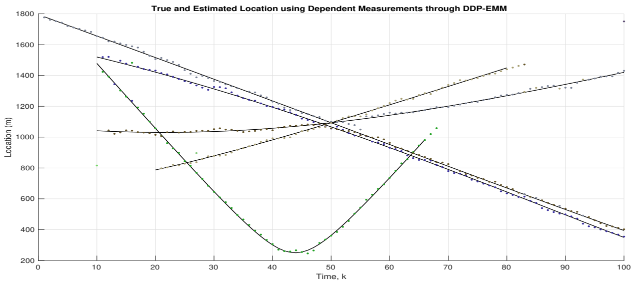


Figure 6.14: Location Estimation in the Presence of Multiple Dependent Measurements.

$p = 1$ and cut-off $c = 100$.

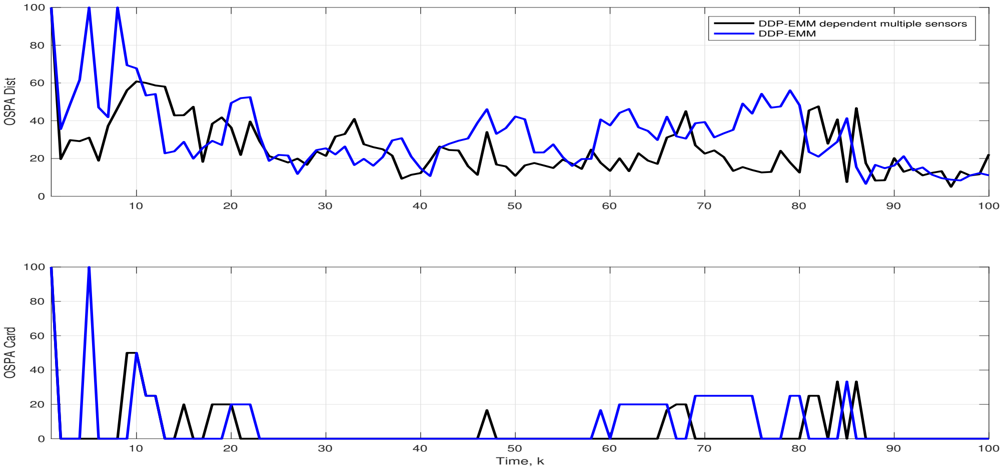


Figure 6.15: OSPA Comparison for Multi-Target Tracking with and without Using the Dependent Measurements.

Figure 6.13 shows the estimated $x$ and $y$ coordinates obtained using the dependent measurements to track multiple objects and its comparison to DDP-EMM. The corresponding location estimate is depicted in Figure 6.14. The OSPA comparison depicted in Figure 6.15 also manifests the advantage of incorporating multiple measurement with DDP-EMM tracker. The provided cardinality graphs in Figure 6.16 is a proof that dependent measurements not only improve the tracker performance but also provide a more robust and accurate object cardinality estimation.

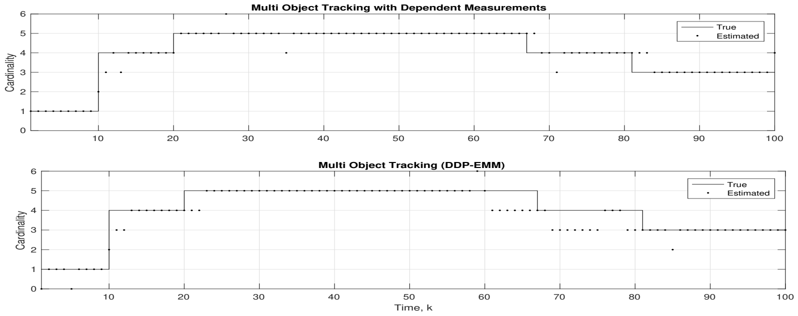


Figure 6.16: Cardinality Estimation in the Presence of Multiple Dependent Measurements.

## 6.4 Discussion

Chapter 6 tackled a more challenging problem of tracking in multimodal dependent measurements. In this chapter, we developed a class of nonparametric models to estimate the dependent measurement density. We used a Bayesian hierarchical model to incorporate the dependency among the measurements. Our hierarchical model took advantage of additional information provided through the dependency of measurements to improve the tracking performance. We extended this model to track multiple objects using measurements received from multiple dependent sensors. A dependent Dirichlet process as a prior on the time-varying object state distributions was integrated with and a hierarchical Dirichlet process mixture to model the evolving objects as well as the measurement dependency. We demonstrated through simulations that considering dependency among the collected measurements by multiple sensors may improve the tracker.

# Chapter 7

# CONTRIBUTIONS AND RECOMMENDATIONS

Proceeding chapters detail statistical models for multi-object and multi-sensor tracking. We outline the primary contributions of this thesis and propose several research problems that can further this thesis.

## 7.1 Summary of Methods and Contributions

Tracking a time-varying number of moving objects using measurements received from multiple dependent sources under adverse operational and environmental conditions has become a principal and highly-involved problem. This problem is prominent in diverse applications, including defense, medical, and surveillance. For instance, this problem arises in tracking multiple moving targets using different types of radar on a multimodal system under high clutter and high noise conditions; or in locating specific cognitive and behavioral information in different regions in the brain by tracking multiple neural dipole sources using patient-dependent eletroencephalography (EEG) recordings which include interference from physiologic and extraphysiologic artifacts.

This thesis primarily integrates the nonparametric Bayesian statistical models as priors with multiple object tracking to perform learning tasks and adapt to poor environmental conditions. These statistical methods are required to robustly and accurately track the trajectory of time-varying objects. We examine the general theme in the context of object tracking and develop nonparametric models to follow this theme. We have developed a class of nonparametric processes to model object evolution and robustly and accurately determine the object identity. These methods robustly estimate the trajectory of each object as well as object cardinality at each time step. We also exploit a hierarchical nonparametric model to make use of the extra information provided by the multiple sensors to robustly track time-dependent objects. Nevertheless, we demonstrate that nonparametric methods can flexibly characterize problems arise in multi-object tracking. We also show these statistical models are both weakly and strongly consistent and the contraction rate matches the minimax rate.

Tracking time-varying object cardinality and identity is a crutial task. Chapter 3 leverages a dependent process based on the Dirichlet process in which the complete dependency among the objects are considered. This model is shown to be an optimal model and can robustly estimate the trajectory of objects as well as the cardinality of the objects in the scene. MCMC methods are also provided to do inference. We show that the introduced MCMC method converges to the true posterior distribution. In addition, the consistency of this process is examined. We show that the proposed process is weakly and strongly consistent. The contraction of posterior distribution coincides with the optimal frequentist rate.

A more flexible model to obtain a robust tracking model is offered by the Pitman-Yor process. The Pitman-Yor process provides a model in which the expected number of clusters follows the power law property. the Pitman-Yor tends to generate more clusters with smaller size; therefore, it can better capture the dependency among objects. Chapter 4 develops a dependent model where the marginal distribution follows a Pitman-Yor process. In addition, by integrating an infinite mixture model, we develop a learning model to infer the object identity and cardinality at each time step. We introduce an efficient MCMC method to do sample from the posterior distribution. Conditions under which this process is consistent is also studied. Simulation results show that these nonparametric models increase the performance of the tracking model compared to existing models such as the DDP-EMM and the labeled multi-Bernoulli trackers.

Generalizing these object tracking models, Chapter 5 proposes a novel class of distributions over infinite trees. We first construct a dependent prior over infinite trees by employing diffusion processes. By utilizing the prior distribution, a learning model to track multiple objects is then introduced. This model efficiently estimates the object identity and cardinality at each time step. The trajectory of each object can be obtained only by searching paths on the infinite random tree which makes this model computationally inexpensive. Empirical results demonstrate the advantages of this nonparametric tracking method over other tracking methods.

Multiple dependent measurements with unknown origin, high noise, time-varying object cardinality, and identity, unknown stochastic state transition models, etc., make object tracking a challenging task. However, it is shown that the information provided due to the dependency of measurements can be utilized towards a more accurate tracking procedure. The challenge is not only to extract the most information but to find a solution for the problem of association. A hierarchical Dirichlet process delivers a promising framework such that the received data can be grouped. Chapter 6 studies this model in detail and provides methods to robustly and accurately track in both single and multiple objects fashions. In particular, we define hierarchical models which describe several dependent and related measurements received by multiple sensors through a common set of shared parts.

## 7.2 Suggestions for Future Research

Approaches discussed in this thesis can potentially be expanded to other fields of study. We conclude this chapter by providing a variety of research directions that can benefit from Bayesian nonparametric modeling, specifically our statistical models. In addition, we briefly discuss the implication of our statistical and computational approach for other tracking problems.

### *7.2.1 Nonparametric Models for Clutter and Spawning*

Bayesian nonparametric approaches can address several tracking problems. Consider the problem of object tracking in the presence of clutter. Tracking in the presence of clutter is a challenging task, wherein the identification of true measurements from a large number of noisy measurements becomes crucial for optimal tracking results. Also, high noise conditions in addition to clutter makes tracking even more difficult. Nevertheless, a generative Bayesian nonparametric approach can be employed to model the measurements. In particular, a joint Dirichlet distribution prior over true measurements and the clutter can be employed to address this issue. This generative model enables us to distinguish the measurements originated from the objects from the clutter or noise. The estimated object measurement distribution may be used in a classical Bayesian single object tracking or an advanced Bayesian multi-object tracking setup. Furthermore, dependent Bayesian nonparametric models can offer a solution to the problem of spawning. Spawning occurs when each measurement is originated from more than one object, and hence the classical clustering cannot provide a solution to the measurement association. However, the Beta-Bernoulli process can provide a Bayesian nonparametric solution for the spawning. This model entails a collection of binary-valued features which can provide information on whether a measurement is originated from a specific target.

### *7.2.2 Nonparametric Models and Causation*

Causality is a relationship between cause (source) and effect (consequence). In object tracking, causal relationships can often be found between the motions of the sensors and that of the tracked object. For example, in a visual object tracking, an abrupt movement of the camera can cause the tracker to fail, even in simple tracking scenarios. Hence, causal relationships are employed to ensure robust prediction/estimation, for example, of the object location, which is crucial to any tracking algorithm. These causal relationships are amalgamated with Bayesian nonparametric models to estimate the object locations accurately. For this reason, the joint distribution of the observed data (outcome, treatment, and confounders) may be modeled through a general Bayesian nonparametric model, such as a Dirichlet process. The combination of the observed data model and causal assumptions allows us to identify any type of causal effect-differences, ratios, or quantile effects, either marginally or for subpopulations of interest. The Bayesian nonparametric model is well-suited for the multi-object tracking and causal inference problems, as it can estimate the location of each object and does not require parametric assumptions about the distribution of confounders and naturally leads to computationally efficient MCMC methods.

### *7.2.3 Dependent Nonparametric Models in Pattern Recognition and its Application to DNA Structure*

Exploring the use of Bayesian nonparametrics to pattern recognition problems may provide interesting results in this field of study. In particular, the problem of tracking patterns in biosequences via Bayesian nonparametric modeling. Patterns in biosequences, such as sequences from peptides microarrays obtained from biological samples, can potentially provide, for example, presymptomatic diagnosis for infectious diseases. Current methods of pattern recognition in peptide sequences rely on long searches using the amino acid one-letter notation representation, as used in presenting alignments of homologous sequences. Using Bayesian nonparametric and advanced processing techniques to perform the search has the potential to improve the classification and identification of the patterns. The use of Bayesian nonparametric adaptive learning techniques allows for further clustering if additional data is received. Pattern discovery is interdisciplinary and can be used in multiple sequence alignments, protein structure, function prediction, characterization of protein families, signal detection, and other areas.

# BIBLIOGRAPHY

[1] A. Yilmaz, O. Javed, and M. Shah, “Object tracking: A survey,” *ACM Computing Surveys (CSUR)*, vol. 38, no. 4, p. 13, 2006.

[2] D. Comaniciu, V. Ramesh, and P. Meer, “Kernel-based object tracking,” *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 5, pp. 564–575, 2003.

[3] W. Koch, *Tracking and Sensor Data Fusion.* Springer, 2013.

[4] D. A. Forsyth and J. Ponce, *A modern approach*, 2nd ed. Pearson, 2003.

[5] S. Avidan, “Support vector tracking,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1. IEEE, 2001, pp. I–I.

[6] M. Nieto, O. Otaegui, G. Vélez, J. D. Ortega, and A. Cortés, “On creating vision-based advanced driver assistance systems,” *IET Intelligent Transport Systems*, vol. 9, no. 1, pp. 59–66, 2014.

[7] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, and S. Shafer, “Multi-camera multi-person tracking for easyliving,” in *IEEE International Workshop on Visual Surveillance*, 2000, pp. 3–10.

[8] C. Wang and M. S. Brandstein, “Multi-source face tracking with audio and visual data,” in *IEEE Third Workshop on Multimedia Signal Processing*, 1999, pp. 169–174.

[9] D. Sun, E. B. Sudderth, and M. J. Black, “Layered image motion with explicit occlusions, temporal consistency, and depth ordering,” in *Advances in Neural Information Processing Systems*, 2010, pp. 2226–2234.

[10] E. B. Sudderth and W. T. Freeman, “Signal and image processing with belief propagation,” *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 114–141, 2008.

[11] S. Churchill, C. Randell, D. Power, and E. Gill, “Data fusion: Remote sensing for target detection and tracking,” in *IEEE International Geoscience and Remote Sensing Symposium*, vol. 1. IEEE, 2004.

[12] H. G. Okuno, K. Nakadai, K. I. Hidai, H. Mizoguchi, and H. Kitano, “Human-robot interaction through real-time auditory and visual multiple-talker tracking,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium*, vol. 3, 2001, pp. 1402–1409.

[13] B. T. Vo, M. Mallick, Y. Bar-shalom, S. Coraluppi, R. Osborne III, R. Mahler, and B.-N. Vo, “Multitarget tracking,” *Wiley Encyclopedia of Electrical and Electronics Engineering*, pp. 1–15, 1999.

[14] B.-T.Vo, B.-N. Vo, and A. Cantoni, “The cardinality balanced multi-target multi-Bernoulli filter and its implementations,” *IEEE Transactions on Signal Processing*, vol. 57, pp. 409–423, 2009.

[15] X. Wang, T. Li, S. Sun, and J. M. Corchado, “A survey of recent advances in particle filters and remaining challenges for multitarget tracking,” *Sensors*, vol. 17, p. 12, 2017.

[16] G. Pulford, “Taxonomy of multiple target tracking methods,” *IEE Proceedings-Radar, Sonar and Navigation*, vol. 152, no. 5, pp. 291–304, 2005.

[17] S. S. Blackman, “Multiple-target tracking with radar applications,” *Dedham, MA, Artech House, Inc., 1986, 463 p.*, 1986.

[18] W. R. Blanding, P. K. Willett, and Y. Bar-Shalom, “Multiple target tracking using maximum likelihood probabilistic data association,” in *IEEE Aerospace Conference*, 2007, pp. 1–12.

[19] F. Y. Jakubiec and A. Ribeiro, “Distributed maximum a posteriori probability estimation for tracking of dynamic systems,” in *Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, 2012, pp. 1478–1482.

[20] E. H. Aoki, P. K. Mandal, L. Svensson, Y. Boers, and A. Bagchi, “Labeling uncertainty in multitarget tracking,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 3, pp. 1006–1020, 2016.

[21] T. Quach and M. Farooq, “Maximum likelihood track formation with the viterbi algorithm,” in *IEEE Conference on Decision and Control*, vol. 1, 1994, pp. 271–276 vol.1.

[22] G. W. Pulford and A. Logothetis, “An expectation-maximisation tracker for multiple observations of a single target in clutter,” in *IEEE Conference on Decision and Control*, vol. 5, 1997, pp. 4997–5003 vol.5.

[23] D. A. Castanon, “Efficient algorithms for finding the K best paths through a trellis,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 26, no. 2, pp. 405–410, 1990.

[24] Y. Bar-Shalom and X.-R. Li, *Multitarget-Multisensor Tracking: Principles and Techniques.* YBs Storrs, CT, 1995, vol. 19.

[25] T. Kirubarajan, Y. Bar-Shalom, and K. R. Pattipati, “Multiassignment for tracking a large number of overlapping objects [and application to fibroblast cells],” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 37, no. 1, pp. 2–21, 2001.

[26] 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.

[27] S. S. Blackman, “Multiple hypothesis tracking for multiple target tracking,” *IEEE Aerospace and Electronic Systems Magazine*, vol. 19, no. 1, pp. 5–18, 2004.

[28] R. Mahler, *Random Set Theory for Target Tracking and Identification.* CRC Press, 2001.

[29] B.-N. Vo and W.-K. Ma, “The Gaussian mixture probability hypothesis density filter,” *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4091–4104, 2006.

[30] S. Reuter, B.-T. Vo, B.-N. Vo, and K. Dietmayer, “The labeled multi-Bernoulli filter,” *IEEE Transactions on Signal Processing*, vol. 62, pp. 3246–3260, 2014.

[31] ——, “The labeled multi-Bernoulli filter,” *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3246–3260, 2014.

[32] B.-N. Vo, B.-T. Vo, and H. G. Hoang, “An efficient implementation of the generalized labeled multi-Bernoulli filter,” *IEEE Transactions on Signal Processing*, vol. 65, no. 8, pp. 1975–1987, 2017.

[33] F. Caron, M. Davy, A. Doucet, E. Duflos, and P. Vanheeghe, “Bayesian inference for linear dynamic models with Dirichlet process mixtures,” *IEEE Transactions on Signal Processing*, vol. 56, no. 1, pp. 71–84, 2008.

[34] E. B. Fox, E. B. Sudderth, and A. S. Willsky, “Hierarchical Dirichlet processes for tracking maneuvering targets,” in *International Conference on Information Fusion*, 2007, pp. 1–8.

[35] E. Fox, E. B. Sudderth, M. I. Jordan, and A. S. Willsky, “Bayesian nonparametric inference of switching dynamic linear models,” *IEEE Transactions on Signal Processing*, vol. 59, pp. 1569–1585, 2011.

[36] F. Caron, M. Davy, and A. Doucet, “Generalized Pólya urn for time-varying Dirichlet process mixtures,” in *Conference on Uncertainty in Artificial Intelligence*, 2007, pp. 33–40.

[37] F. Caron, W. Neiswanger, F. Wood, A. Doucet, and M. Davy, “Generalized Pólya urn for time-varying Pitman-Yor processes,” *Journal of Machine Learning Research*, vol. 18, no. 27, pp. 1–32, 2017.

[38] M. J. Wainwright, M. I. Jordan *et al.*, “Graphical models, exponential families, and variational inference,” *Foundations and Trends® in Machine Learning*, vol. 1, no. 1–2, pp. 1–305, 2008.

[39] L. D. Brown, *Fundamentals of statistical exponential families: With applications in statistical decision theory.* Institute of Mathematical Statistics, 1986.

[40] C. Robert, *The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation.* Springer, 2007.

[41] J. M. Bernardo and A. F. M. Smith, *Bayesian Theory.* Wiley Series in Probability and Statistics, 2007.

[42] E. B. Sudderth, "Graphical models for visual object recognition and tracking," Ph.D. dissertation, Massachusetts Institute of Technology, 2006.

[43] A. Gelman, H. S. Stern, J. B. Carlin, D. B. Dunson, A. Vehtari, and D. B. Rubin, *Bayesian Data Analysis.* Chapman and Hall/CRC, 2013.

[44] G. Casella and R. Berger, *Statistical Inference*, ser. Duxbury advanced series in statistics and decision sciences. Duxbury Press, 2002.

[45] P. D. Hoff, *A First Course in Bayesian Statistical Methods.* Springer, 2009, vol. 580.

[46] P. Orbanz, "Lecture notes on bayesian nonparametrics," *Department of Statistics, Columbia University, New York, NY*, Spring 2017.

[47] T. S. Ferguson, "A Bayesian analysis of some nonparametric problems," *The Annals of Statistics*, pp. 209–230, 1973.

[48] J. Sethuraman, "A constructive definition of Dirichlet priors," *Statistica sinica*, pp. 639–650, 1994.

[49] D. J. Aldous, "Exchangeability and related topics," *Lecture Notes in Mathematics*, vol. 1117, pp. 1–198, 1985.

[50] D. Blackwell, J. B. MacQueen *et al.*, "Ferguson distributions via Pólya urn schemes," *The Annals of Statistics*, vol. 1, no. 2, pp. 353–355, 1973.

[51] J. Kingman, "Completely random measures," *Pacific Journal of Mathematics*, vol. 21, no. 1, pp. 59–78, 1967.

[52] R. M. Neal, "Markov chain sampling methods for Dirichlet process mixture models," *Journal of Computational and Graphical Statistics*, vol. 9, pp. 249–265, 2000.

[53] M. D. Escobar and M. West, "Bayesian density estimation and inference using mixtures," *Journal of the American Statistical Association*, vol. 90, no. 430, pp. 577–588, 1995.

[54] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Hierarchical Dirichlet processes," *Journal of the American Statistical Association*, pp. 1566–1581, 2006.

[55] M. E. Newman, "Power laws, Pareto distributions and Zipf's law," *Contemporary Physics*, vol. 46, no. 5, pp. 323–351, 2005.

[56] J. Pitman and M. Yor, "The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator," *The Annals of Probability*, pp. 855–900, 1997.

[57] M. Perman, J. Pitman, and M. Yor, “Size-biased sampling of Poisson point processes and excursions,” *Probability Theory and Related Fields*, vol. 92, no. 1, pp. 21–39, 1992.

[58] S. Goldwater, T. L. Griffiths, and M. Johnson, “Producing power-law distributions and damping word frequencies with two-stage language models,” *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2335–2382, 2011.

[59] C. Robert and G. Casella, *Monte Carlo Statistical Methods*. Springer Science & Business Media, 2013.

[60] C. Andrieu, N. De Freitas, A. Doucet, and M. I. Jordan, “An introduction to MCMC for machine learning,” *Machine learning*, vol. 50, no. 1-2, pp. 5–43, 2003.

[61] D. J. MacKay and D. J. Mac Kay, *Information Theory, Inference and Learning Algorithms*. Cambridge university press, 2003.

[62] D. J. MacKay, “Introduction to monte carlo methods,” in *Learning in Graphical Models*. Springer, 1998, pp. 175–204.

[63] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe, “Variational inference: A review for statisticians,” *Journal of the American Statistical Association*, vol. 112, no. 518, pp. 859–877, 2017.

[64] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[65] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. MIT press, 2012.

[66] S. N. MacEachern, M. Clyde, and J. S. Liu, “Sequential importance sampling for nonparametric Bayes models: The next generation,” *Canadian Journal of Statistics*, vol. 27, no. 2, pp. 251–267, 1999.

[67] F. Liang, “Dynamically weighted importance sampling in Monte Carlo computation,” *Journal of the American Statistical Association*, vol. 97, no. 459, pp. 807–821, 2002.

[68] S. Chib and E. Greenberg, “Understanding the Metropolis-Hastings algorithm,” *The American Statistician*, vol. 49, no. 4, pp. 327–335, 1995.

[69] K. L. Mengersen, R. L. Tweedie *et al.*, “Rates of convergence of the Hastings and Metropolis algorithms,” *The Annals of Statistics*, vol. 24, no. 1, pp. 101–121, 1996.

[70] J. S. Liu, W. H. Wong, and A. Kong, “Covariance structure and convergence rate of the Gibbs sampler with various scans,” *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 57, no. 1, pp. 157–169, 1995.

[71] G. O. Roberts and S. K. Sahu, “Updating schemes, correlation structure, blocking and parameterization for the Gibbs sampler,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 59, no. 2, pp. 291–317, 1997.

[72] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the EM algorithm,” *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 39, no. 1, pp. 1–22, 1977.

[73] R. M. Neal and G. E. Hinton, “A view of the EM algorithm that justifies incremental, sparse, and other variants,” in *Learning in graphical models*. Springer, 1998, pp. 355–368.

[74] E. L. Lehmann and G. Casella, *Theory of Point Estimation*. Springer Science & Business Media, 2006.

[75] G. Casella and C. P. Robert, “Rao-Blackwellisation of sampling schemes,” *Biometrika*, vol. 83, no. 1, pp. 81–94, 1996.

[76] A. E. Gelfand and A. F. Smith, “Sampling-based approaches to calculating marginal densities,” *Journal of the American Statistical Association*, vol. 85, no. 410, pp. 398–409, 1990.

[77] S. N. MacEachern, “Estimating normal means with a conjugate style dirichlet process prior,” *Communications in Statistics-Simulation and Computation*, vol. 23, no. 3, pp. 727–741, 1994.

[78] S. N. MacEachern and P. Müller, “Estimating mixture of Dirichlet process models,” *Journal of Computational and Graphical Statistics*, vol. 7, no. 2, pp. 223–238, 1998.

[79] R. M. Neal, “Slice sampling,” *The Annals of Statistics*, vol. 31, no. 3, pp. 705–767, 2003.

[80] B. Moraffah and A. Papandreou-Suppappola, “Dependent Dirichlet process modeling and identity learning for multiple object tracking,” in *2018 52nd Asilomar Conference on Signals, Systems, and Computers*, 2018, pp. 1762–1766.

[81] ——, “Nonparametric Bayesian methods and the dependent Pitman-Yor process for modeling evolution in multiple object tracking,” in *22nd International Conference on Information Fusion*, 2019.

[82] ——, “Tracking multiple objects with dependent measurements using Bayesian nonparametric modeling,” in *Asilomar Conference on Signals, Systems, and Computers*, 2019.

[83] ——, “Inference for multi object tracking: A Bayesian nonparametric approach,” in *IEEE Transaction on Signal Processing*, 2019.

[84] B.-N. Vo, M. Mallick, Y. Bar-Shalom, S. Coraluppi, R. O. III, R. Mahler, and B.-T. Vo, “Multitarget tracking,” *Wiely Encyclopedia of Electrical Engineering*, 2015.

[85] S. N. MacEachern, “Dependent Dirichlet processes,” Department of Statistics, Ohio State University, Tech. Rep., 2000.

[86] S. N. MacEarchern, “Computational methods for mixture of Dirichlet process models,” in *Practical Nonparametric and Semiparametric Bayesian Statistics*, D. Dey, P. Müller, and D. Sinha, Eds. Springer, 1998, vol. 133.

[87] S. N. MacEachern, “Dependent Dirichlet processes,” Unpublished manuscript, Department of Statistics, Ohio State University, pp. 1–40, 2000.

[88] C. E. Antoniak, “Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems,” *The Annals of Statistics*, pp. 1152–1174, 1974.

[89] L. Tierney, “Markov chains for exploring posterior distributions,” *The Annals of Statistics*, pp. 1701–1728, 1994.

[90] M. D. Escobar, “Estimating normal means with a Dirichlet process prior,” *Journal of the American Statistical Association*, vol. 89, no. 425, pp. 268–277, 1994.

[91] L. Schwartz, “On consistency of Bayes procedures,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 52, no. 1, p. 46, 1964.

[92] S. Ghosal, J. K. Ghosh, and Ramamoorthi, “Posterior consistency of Dirichlet mixtures in density estimation,” *Annual of Statistics*, vol. 27, no. 1, pp. 143–158, 1999.

[93] A. Barron, M. J. Schervish, and L. Wasserman, “The consistency of posterior distributions in nonparametric problems,” *The Annals of Statistics*, vol. 27, no. 2, pp. 536–561, 1999.

[94] S. Ghosal and V. D. Vaart, “Posterior convergence rates of Dirichlet mixtures at smooth densities,” *The Annals of Statistics*, vol. 35, no. 2, pp. 697–723, 2007.

[95] V. Tikhomirov, “$\varepsilon$-Entropy and $\varepsilon$-Capacity of Sets in Functional Spaces,” in *Selected works of AN Kolmogorov*. Springer, 1993, pp. 86–170.

[96] J. Wellner *et al.*, *Weak Convergence and Empirical Processes: With Applications to Statistics*. Springer Science & Business Media, 2013.

[97] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, “A consistent metric for performance evaluation of multi-object filters,” *IEEE transactions on signal processing*, vol. 56, no. 8, pp. 3447–3457, 2008.

[98] J. Pitman, “Poisson-Dirichlet and GEM invariant distributions for split-and-merge transformations of an interval partition,” *Combinatorics, Probability and Computing*, vol. 11, no. 5, pp. 501–514, 2002.

[99] F. Papi, B.-N. Vo, B.-T. Vo, C. Fantacci, and M. Beard, “Generalized labeled multi-Bernoulli approximation of multi-object densities,” *IEEE Transactions on Signal Processing*, vol. 63, no. 20, pp. 5487–5497, 2015.

[100] P. De Blasi, A. Lijoi, and I. Prünster, “On consistency of Gibbs-type priors,” in *World Statistics Congress of ISI*, 2011.

[101] A. Lijoi and I. Prünster, "Models beyond the Dirichlet process," *Bayesian Nonparametrics*, vol. 28, no. 80, p. 342, 2010.

[102] A. Lijoi, I. Prünster, and S. G. Walker, "Investigating nonparametric priors with Gibbs structure," *Statistica Sinica*, pp. 1653–1668, 2008.

[103] B. Moraffah and A. Papandreou-Suppappola, "Random infinite tree and dependent Poisson diffusion process for nonparametric Bayesian modeling in multiple object tracking," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 5217–5221.

[104] R. P. S. Mahler, *Statistical Multisource-Multitarget Information Fusion.* Artech House, Inc., 2007.

[105] W. Koch, *Tracking and Sensor Data Fusion.* Springer, 2016.

[106] B.-N. Vo, M. Mallick, Y. Bar-Shalom, S. Coraluppi, R. O. III, R. Mahler, and B.-T. Vo, "Multitarget tracking," *Wiely Encyclopedia of Electrical Engineering*, 2015.

[107] V. Kettnaker and R. Zabih, "Bayesian multi-camera surveillance," in *Conference on Computer Vision and Pattern Recognition*, 1999, pp. 253–259.

[108] L. Mihaylova, P. Brasnett, N. Canagarajah, and D. Bull, "Object tracking by particle filtering techniques in video sequences," in *Advances and Challenges in Multisensor Data and Information*, E. Lefebvre, Ed. IOS Press, 2007, vol. 8, pp. 260–268.

[109] R. Kümmerle, M. Ruhnke, B. Steder, C. Stachniss, and W. Burgard, "Autonomous robot navigation in highly populated pedestrian zones," *Journal of Field Robotics*, vol. 32, pp. 565–589, 2015.

[110] B. Moraffah and A. Papandreou-Suppopola, "Dependent Dirichlet process modeling and identity learning for multiple object tracking," in *Asilomar Conference on Signals, Systems, and Computers*, 2018, pp. 1762–1766.

[111] R. M. Neal, "Density modeling and clustering using Dirichlet diffusion trees," in *Bayesian Statistics 7*, J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid *et al.*, Eds. Oxford University Press, 2003, pp. 619–629.

[112] D. A. Knowles and Z. Ghahramani, "Pitman-Yor diffusion trees," in *Conference in Uncertainty in Artificial Intelligence*, 2011, pp. 410–418.

[113] Z. Ghahramani, "Bayesian non-parametrics and the probabilistic approach to modelling," *Philosophical Transactions of the Royal Society*, p. 20 pgs, 2013.

[114] E. W. Meeds, D. A. Ross, R. S. Zemel, and S. T. Roweis, "Learning stick-figure models using nonparametric Bayesian priors over trees," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[115] D. A. Knowles, J. V. Gael, and Z. Ghahramani, “Message passing algorithms for Dirichlet diffusion trees,” in *International Conference on Machine Learning*, 2011, pp. 721–728.

[116] K. Granström, M. Baum, and S. Reuter, “Extended object tracking: Introduction, overview and applications,” *Journal of Advances in Information Fusion*, vol. 12, pp. 139–174, 2017.

[117] J. J. Zhang, Q. Ding, S. Kay, A. Papandreou-Suppappola, and M. Rangaswamy, “Agile multi-modal tracking with dependent measurements,” in *2010 Conference Record of the Forty Fourth Asilomar Conference on Signals, Systems and Computers*. IEEE, 2010, pp. 1653–1657.

[118] S. Kay and Q. Ding, “Exponentially embedded families for multimodal sensor processing,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2010, pp. 3770–3773.

[119] B. Moraffah, C. Brito, B. Venkatesh, and A. Papandreou-Suppappola, “Use of hierarchical Dirichlet processes to integrate dependent observations from multiple disparate sensors for tracking,” in *International Conference on Information Fusion*, 2019.

[120] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, “Sharing clusters among related groups: Hierarchical Dirichlet processes,” in *Advances in Neural Information Processing Systems*, 2005, pp. 1385–1392.

[121] L. Ren, D. B. Dunson, and L. Carin, “The dynamic hierarchical Dirichlet process,” in *International Conference on Machine Learning*. ACM, 2008, pp. 824–831.

[122] S. Liu, S. Bhat, J. J. Zhang, Q. Ding, R. Narayanan, A. Papandreou-Suppappola, S. Kay, and M. Rangaswamy, “Design and performance of an integrated waveform-agile multi-modal track-before-detect sensing system,” *Asilomar Conference on Signals, Systems and Computers*, pp. 1530–1534, 2011.