INTELLIGENT DESIGN IN WIRELESS SYSTEM

A Dissertation

by

YANYAN ZHANG

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Shuguang Cui
Committee Members,	Tie Liu
	Xiaoning Qian
	Huiyan Sang
Head of Department,	Miroslav M. Begovic

December 2017

Major Subject: Electrical Engineering

ABSTRACT

We are living in an era full of data services, and the advancement in statistical learning encourages the development of intelligent system design algorithms based on practical data. In our work, we plan to study two potential applications with intelligent design in wireless systems based on statistical and machine learning techniques.

The first application we study is the spectrum sensing problem in energy harvesting based cognitive radio networks, which is a promising solution to address the shortage of both spectrum and energy. Since the spectrum access and power consumption pattern are interdependent, and the power value harvested from certain environmental sources are spatially correlated, the new power dimension could provide additional information to enhance the spectrum sensing accuracy. In our work, the Markovian behavior of the primary users is considered, based on which we adopt a hidden input Markov model to specify the primary vs. secondary dynamics in the system. Accordingly, we propose a 2-D spectrum vs. power (harvested) sensing scheme to improve the primary user detection performance, which is also capable of estimating the primary transmit power level. Theoretical and simulated results demonstrate the effectiveness of the proposed scheme, in terms of the performance gain achieved by considering the new power dimension. To the best of our knowledge, this is the first work to jointly consider the spectrum and power dimensions for the cognitive primary user detection problem.

The second work is about spatio-temporal base station traffic prediction with machine learning. Accurate prediction of user traffic in cellular networks is crucial to improve the system performance in terms of energy efficiency and resource utilization. However, existing work mainly considers the temporal traffic correlations within each cell while neglecting the spatial correlation across neighboring cells. In this work, machine learning

models that jointly explore the spatio-temporal correlations are proposed, where a multi-task learning approach is adopted to explore the commonalities and differences across cells in improving the prediction performance. Base on real data, we demonstrate the benefits of joint learning over spatial and temporal dimensions.

DEDICATION

This dissertation is dedicated to my mother Wei, my father Ping and my husband Renjie.

I could not have achieved this without their encouragement and support.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a dissertation committee consisting of Professor Shuguang Cui, Professor Tie Liu and Professor Xiaoning Qian of the Department of Electrical and Computer Engineering and Professor Huiyan Sang of the Department of Statistics.

The data analyzed for Chapter 3 was provided by Professor Lin Zhang in Beijing University of Posts and Telecommunications. The analyses depicted in Chapter 2 were conducted in contribution with Professor Weijia Han in Shaanxi Normal University and analyses depicted in Chapter 3 were with Mr. Chen Qiu from Beijing University of Posts and Telecommunications.

All other work conducted for the dissertation was completed by the student independently.

Funding Sources

Graduate study was supported by a research assistantship from Texas A&M University, with funding supported in part by DOD with grant HDTRA1-13-1-0029, by grant NSFC-61328102/61629101, and by NSF with grants DMS-1622433, AST-1547436, ECCS-1508051/1659025, and CNS-1343155.

NOMENCLATURE

HMM	Hidden Markov Model
HIMM	Hidden Input Markov Model
EM	Expectation-Maximization
RNN	Recurrent Neural Network
LSTM	Long-Short Term Memory
CR	Cognitive Radio
PU	Primary User
SU	Secondary User
EH	Energy Harvesting
QoS	Quality of Service
CRAN	Cloud Radio Access Network
BBU	Baseband Unit

TABLE OF CONTENTS

Page

Page

LIST OF FIGURES

FIGURE Page

1. INTRODUCTION

Due to the growing popularity of computation driven approaches, intelligent designs which are capable of perceiving environment then taking actions are on demand, since it helps increase the efficiency and maximize the chance of success. Intelligent designs fit the practical systems well with the flexibility to adapt in real time. Many statistical methods have been built and utilized with supervised, unsupervised and semi-supervised models, and the subsequent research attempts to use these methods to solve specific problems.

In our work, we plan to use several statistical techniques with two applications of intelligent design:

1. Spectrum sensing in energy harvesting based cognitive radio network. Since spectrum and power dimensions are correlated, we propose a Hidden Input Markov Model to explore the correlations, to support a joint sensing scheme. Such spectrum vs. power 2-D joint sensing scheme improves the primary user detection performance, which is also capable of estimating the primary transmit power level. Meanwhile, we also propose EM-based HIMM parameter estimation algorithms to learn the real parameters of the models.

2. Spatio-temporal base station traffic prediction. A lot of modern system designs require the prior knowledge of base station traffics; but the current base station traffic prediction techniques are limited to either model-based analysis or only taking the temporal relationship of data into consideration. Since in real wireless networks, the traffic flows of base stations are correlated over both time and space, we propose a spatio-temporal base station traffic prediction method with machine learning.

2. 2-D SPECTRUM SENSING IN ENERGY HARVESTING BASED COGNITIVE RADIO NETWORK

2.1 Introduction

With the rapid development of wireless services in the past few decades, spectrum resource, which is vital and hard-limited, faces a critical situation of being scarce. However, studies have revealed that we are wasting the spectrum as most allocated bands are being under-utilized. To address this problem, researchers have proposed the idea of dynamic spectrum access, which could help increase spectrum efficiency.

Cognitive Radio (CR) is the well-accepted technology to achieve dynamic spectrum access, with its core idea of allowing Secondary Users (SUs) to access spectrum when the licensed Primary Users (PUs) are idle. The goal for CRs is to maximize the overall spectrum efficiency while preventing harmful interference to PU transmissions. One crucial building block of CR is spectrum sensing, which determines whether certain spectrum is occupied by some active PUs.

2.1.1 Related Works and Motivations

Many statistical methods have been adopted for spectrum sensing in cognitive radio design. The authors in [1] apply the Neyman-Perason lemma to study both the local and cooperative PU detection schemes, in which the sufficient statistics is compared against a certain threshold to detect the channel status. When the PU transmission signaling is known at the SU side, cyclostationary features could be explored for PU detection [2]. For wideband cognitive radio systems, in addition to energy detection [3], compressive sensing

Part of this chapter is reprinted with permission from "Power Versus Spectrum 2-D Sensing in Energy Harvesting Cognitive Radio Networks," by Y. Zhang, W. Han, D. Li, P. Zhang and S. Cui, in IEEE Transactions on Signal Processing, vol. 63, no. 23, pp. 6200-6212, Dec.1, 2015 and "Two-dimensional sensing in energy harvesting cognitive radio networks," by Y. Zhang, W. Han, D. Li, P. Zhang and S. Cui, 2015 IEEE International Conference on Communication Workshop (ICCW), London, 2015, pp. 2029-2034.

[4] is also adopted to efficiently identify spectrum holes. However, most of methods above are sensitive to the shadowing/fading effects over the PU-to-SU link.

On the other hand, there are many related works that explore the PU Markovian behavior. The existence of PU Markov patterns is validated in [5] by real-time measurements in the paging band (928-948 MHz), with different effects of false alarm and miss detection are studied in [6], where a modified forward-backward detection algorithm is proposed to minimize the detection risks. In [7], a Hidden Bivariate Markov Model (HBMM) is used to quantify both the channel status and its dwelling time. For collaborative spectrum sensing, a parameter estimation algorithm with classification method is introduced in [8] to identify the malicious users based on a Hidden Markov Model (HMM).

Meanwhile, due to the growing demand of energy efficiency, the energy harvesting based cognitive radio network emerges to both improve channel utilization and meet the requirement of green communications. In [9], the average throughput of the energy harvesting secondary network is maximized, where the optimality is achieved by balancing the optimal sensing duration and the sensing threshold. In [10], the SU time slot is segmented into three non-overlapping fractions and the “harvesting-sensing-throughput" tradeoff is quantified with consideration of a generalized multi-slot spectrum sensing paradigm and two types of fusion rules. However, the above work does not consider PU energy harvesting. In the power domain, Markov chain models have been widely accepted [11] [12] to specify the energy arrival process, for which some other statistical models have also been applied. For example, a Poisson model with a known intensity λ_0 is adopted in [13] and a Gamma distribution model is adopted in [14]. Meanwhile, for wind energy harvesting systems, Weibull distribution [15] is widely adopted to forecast the wind speed and the corresponding harvested power level.

Unlike traditional communication systems, for the ones powered by the environment energy harvesters, power becomes a multiple access medium since the power usages across

different users (especially geographically neighboring ones) are correlated [16] [17] [18]. Accordingly, the spectrum access and power access are actually correlated events, and such correlations could be explored to help with improving the PU detection performance in CR, which used to be solely dependent on spectrum sensing. In this work, we consider the PU detection problem in energy harvesting based cognitive radio networks. Given the correlation between spectrum and power usages, and by considering the spatial correlation among energy harvesting users, we propose a 2-D sensing scheme that could infer the PU behaviors by jointly learning the spectrum and power access dynamics. This is promising to improve the PU detection performance since the traditional methods, which are solely based on spectrum sensing, have certain limitation: When the PU-to-SU transmission is under fading/shadowing, the detection performance degrades sharply as the channel observation is no longer reliable. Since the PU-to-SU channel quality does not affect the power-dimension inference, the proposed 2-D scheme could overcome the effect of channel fading/shadowing and provide a more reliable combined sensing result.

In addition, traditional spectrum sensing methods only focus on sensing the “on-off" activity of PUs. There is also a growing demand of knowing the transmission power levels of PUs. For example, in [19], a novel method that utilizes not only the temporal but also the spatial spectrum holes is proposed, which requires the knowledge of PU transmission power to estimate the PU coverage area. In [20], an optimal SU power allocation scheme is provided, which also depends on the knowledge of multiple PU transmission power levels. As a side product, the 2-D sensing scheme proposed in this paper could sense the spectrum and estimate the PU transmit power level simultaneously, which could provide more potentials to enhance the performance of the energy harvesting based CR networks.

2.1.2 Summary of Contributions

In this work, we discuss the 2-dimensional joint spectrum sensing scheme in energy harvesting based cognitive radio system, such that both spectrum and power information are utilized to enhance the detection performance.

The main contributions are summarized as follows:

1. First, we construct a so-called Hidden Input Markov Model structure to specify the correlation between PU and SU activities;

2. We propose algorithms to estimate the HIMM parameters, by which the exact HIMM is specified; we also propose a PU activities detection algorithm, which could jointly sense the existence of PU and estimate the PU energy level in each time slot;

3. Finally, we demonstrate the effectiveness of the proposed methods with analytical and simulation results.

2.2 System Model

We consider a simple cognitive system with one PU and one SU, which are both powered by harvested energy from the same renewable source. If the PU does not transmit at certain time slots, the harvested energy is discarded since no battery is assumed [21]. In our setup, it is possible that even when the PU has data, the energy level may not meet the reliable transmission minimum requirement, which implies that the PU does not occupy the channel. This is the extra PU idle case compared with the traditional PU system where PU is idle only if it has no data to transmit. Specifically, let H_0 denote the case when the PU does not transmit and H_1 denote the case when the PU occupies the channel; then a formal definition for H_0 and H_1 in an energy harvesting based cognitive radio network is:

$$H_0 : No\ Enough\ Energy \text{ or } No\ Data\ Available\ ;$$

H_1 : $Enough\ Energy$ and $Data\ Available$.

Let $\{E_t, t = 1, 2, 3, \cdots\}$ denote the energy arrival process at the PU on a time-slotted basis, where each E_t could take value from a set with a finite cardinality $\mathscr{L} = \{i, i+1, i+2, \cdots, i+L-1\}$, $i \geq 0$, i.e., the harvested energy value is quantized into L levels. Here L is set as a relatively large number, such that the effect of discretizing the PU energy level is negligible. The channel occupancy state $\{C_t, t = 1, 2, 3, \cdots\}$ is a discrete-time process such that each state takes value in $\mathscr{C} = \{0, 1, 2, \cdots, M-1\}$, with $M = 2$ in this paper, which implies: When $C_t = 0$ the channel is idle, and when $C_t = 1$ the channel is occupied by the PU.

Let E_{h} be the minimum energy level required for reliable primary transmission and P_0 be the probability of no data available. Then the relationship between the channel state $\{C_t\}$ and the PU energy level $\{E_t\}$ could be expressed as

$$\begin{aligned} P(C_t = 0) &= P(E_t < E_{\text{h}}) + P(E_t \geq E_{\text{h}}) \cdot P_0, \\ P(C_t = 1) &= P(E_t \geq E_{\text{h}}) \cdot (1 - P_0). \end{aligned} \tag{2.1}$$

Note that we assume no battery installed such that when a PU transmits, it uses up all the harvested energy available at that time slot; when a PU does not transmit, it discards the harvested energy in that time slot. On the SU side, since the channel state and PU energy level are not directly observable, SU could only estimate the hidden states with its available observations. First, as both PU and SU are powered by harvested energy from the same renewable source, the harvested SU energy could be treated as an observation for the PU energy level, due to the spatial correlation of the energy harvesting processes. The relationship between the latent PU harvested energy E_t and the SU harvested energy U_t will be further discussed in Section III (See Fig. 2.1).

Assuming that the PU and SU operate on a synchronous time-slotted fashion, the sampled received signal from PU to SU at time slot t is given as (under real-valued signaling assumption)

$$\begin{cases} \mathrm{H}_0: & x_t(n) = u_t(n), \quad n = 1, 2, \cdots, N, \\ \mathrm{H}_1: & x_t(n) = h_t \cdot s_t(n) + u_t(n), \quad n = 1, 2, \cdots, N, \end{cases} \tag{2.2}$$

where n is the sampling index with a total of N samples per slot, $s_t(n)$ is the signal transmitted by the PU, h_t is the channel gain constant over each slot, and $u_t(n)$ denotes the i.i.d. Gaussian noise. The signal sample $s_t(n)$ is assumed i.i.d and independent from the noise. Denote Y_t as the summation of N received signal energy samples at the SU, i.e.,

$$Y_t = \sum_{i=0}^{N-1} x_t^2(i). \tag{2.3}$$

2.3 Hidden Input Markov Model

Markov chain has been widely adopted [11] [12] to specify the environmental energy harvesting process, and to model the PU data arrival process in traditional cognitive radio networks [5] [7]. In our work here, we first adopt two discrete-time Markov chains to represent the PU data arrival and energy harvesting processes respectively, then quantify the 2-D signaling structure with a HIMM as shown in Fig. 2.1.

From Fig. 2.1, we see that there are several differences between the traditional HMMs and the Markov structure abstracted in our problem. If the PU energy level is known, which implies that U_t reflects E_t perfectly, then the structure of our problem becomes similar to the Input-Output Hidden Markov Model (IOHMM) [22]. For IOHMM, the training process is a supervised learning problem, such that with the training input and output, the functional mapping between the input and the output could be inferred. Our

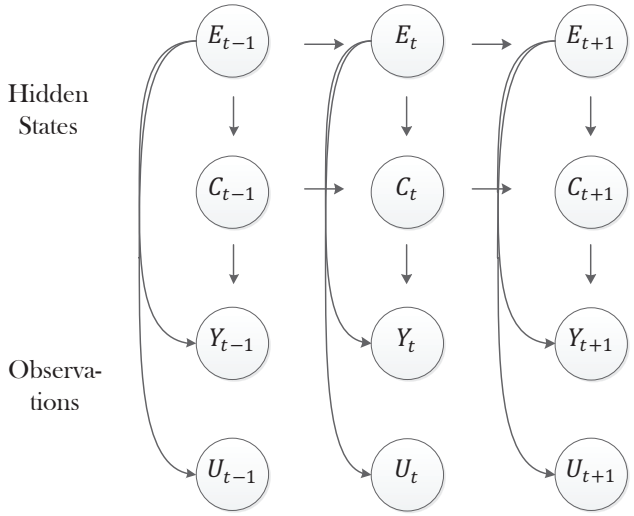


Figure 2.1: Hidden Input Markov Model Structure

structure is more complicated than IOHMM, since the unobservable Markov state C_t is a function of both the previous state C_{t-1} and the input E_t, while the input E_t is also a hidden Markov process from the SU's point of view. On the other hand, the hidden input could be observed not only from the SU energy level U_t, but also from the channel observation Y_t when the channel is active. We thus call this structure as HIMM, for which the existing HMM results could not be directly applied.

As both $\{E_t\}$ and $\{C_t\}$ are first-order Markov chains, according to (2.1) we have $C_t = f(C_{t-1}, E_t)$. Let C^t denote a collection of channel states from time 1 up to time t (with a similar definition for E^t). Throughout the paper, the subscript t means the state at a certain time slot, and the superscript t means a collection of all states happened until time t. Following the above notations, the first-order Markov property implies the following relationship

$$\begin{aligned} P(C_t|C^{t-1}, E_t) &= P(C_t|C_{t-1}, E_t), \\ P(E_t|E^{t-1}) &= P(E_t|E_{t-1}). \end{aligned} \tag{2.4}$$

Throughout the paper the notation $P(A)$ always refers to the probability of the random variable taking a particular value, i.e., $P(A) = P(A = a)$, unless specified otherwise.

We use $\mathbf{A}$ and $\mathbf{B}$ to denote the transition matrices for the processes of $\{E_t\}$ and $\{C_t\}$, respectively. Clearly, as $\{E_t\}$ is a traditional Markov chain and takes values from L states, $\mathbf{A} = \{a_{ij}\}$ is a $L \times L$ matrix, in which each component a_{ij} is given as

$$a_{ij} = P(E_t = j | E_{t-1} = i), \quad i, j \in \mathscr{L}. \tag{2.5}$$

The transition matrix of $\mathbf{B} = \{\mathbf{B}(q), q \in \mathscr{L}\}$ is actually a set of matrices. For each q, $\mathbf{B}(q)$ is a $M \times M$ matrix. Inside $\mathbf{B}(q)$, each component $b_{ij,q}$ indicates the following transition relationship of channel states:

$$b_{ij,q} = P(C_t = j | C_{t-1} = i, E_t = q), \quad i, j \in \mathscr{C}, q \in \mathscr{L}. \tag{2.6}$$

Besides the transition probability, the initial probability distribution is also important in describing a Markov chain. Here we use vector π^E and matrix $\boldsymbol{\pi}^C$ to specify the initial distributions, in which each element stands for

$$\begin{aligned} \pi_i^E &= P(E_1 = i), \quad i \in \mathscr{L} \\ \pi_{ij}^C &= P(C_1 = j | E_1 = i), \quad i \in \mathscr{L}, j \in \mathscr{C}. \end{aligned} \tag{2.7}$$

For the HIMM shown in Fig. 2.1, the SU observations include the SU energy level U_t and the SU received signal energy Y_t. Mathematically, due to spatial correlation, we could model U_t as a function of E_t. Without loss of generality, we assume $U_t \in \mathscr{L}$. Let the

$L \times L$ matrix $\mathbf{D} = \{d_{ij}\}$ be the emission matrix with each d_{ij} defined as

$$d_{ij} = P(U_t = j | E_t = i), \quad i, j \in \mathscr{L}. \tag{2.8}$$

Note that conditioned on E_t, the only randomness left at U_t is the measurement noise, which we could assume independent over time. Therefore, we could have the following decomposition:

$$P(U^t | E^t) = \prod_{i=1}^{t} P(U_i | E_i). \tag{2.9}$$

On the other hand, based on the Central Limit Theorem (CLT), when the number of received signal samples N is relatively large, the sum in (2.3) follows a Gaussian distribution. Therefore, conditioned on the hidden states, we assume that the observation $\{Y_t\}$ follows a Gaussian distribution, having its relationship with E_t and C_t shown in Fig. 2.1. According to the SU received signal in (2.2), when the channel state is idle, the output Y_t only reflects the energy of channel noise, which contains no information about the PU energy level; when the channel is busy, the channel output includes the PU signal, such that the distribution of Y_t is affected by both the hidden states E_t and C_t. For each t, Y_t is distributed as

$$P(Y_t | C_t = i, E_t = j) \sim N(\mu_{ij}, \sigma_{ij}^2), \quad i \in \mathscr{C}, j \in \mathscr{L}. \tag{2.10}$$

Based on our previous discussion, when $i = 0$, μ_{ij} and σ_{ij}^2 are respectively identical for all j's. Let matrix $\boldsymbol{\mu} = \{\mu_{ij}\}$ and matrix $\boldsymbol{\sigma}^2 = \{\sigma_{ij}^2\}$, the emission probability of Y_t is then specified by $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}^2$. Also, given the hidden states, the channel observation Y_t is conditionally independent over time.

The initial probability, transition probability, and emission probability are three im-

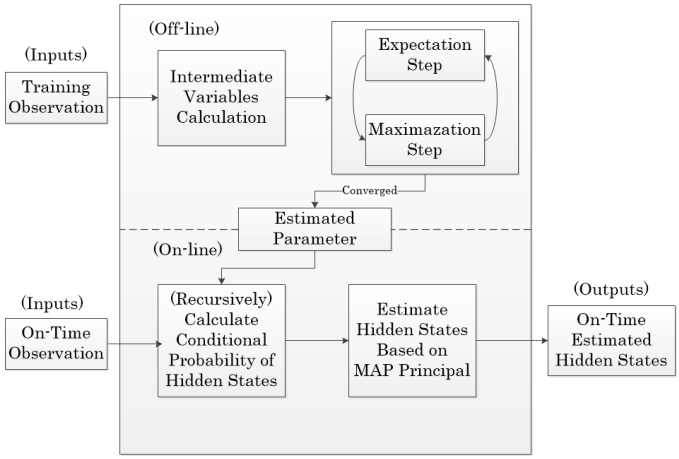


Figure 2.2: Structure of the Proposed Algorithm

portant parts that constitute the parameters of this unobservable Markov chain. Let $\eta = \{\pi^E, \boldsymbol{\pi}^C, \mathbf{A}, \mathbf{B}, \mathbf{D}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2\}$ be the collection of these parameters. When η is decided, the entire structure of this Markov model is specified. Therefore, learning these parameters is crucial in order for us to explore the application of this unobservable Markov chain, which is used in our PU detection scheme.

2.4 2-D Sensing

In this section, we analyze the PU detection scheme with 2-D sensing over the HIMM model. Let t denote the current time slot; since HIMM takes the past observations into consideration, at time t the past and current observations $\{U^t, Y^t\}$ are all available to contribute to the detection process. With the estimated parameter vector η', the probability measure of the current PU hidden states could be obtained by recursively marginalizing all the past states, while the observations in each time slot work as correctors to modify the previously predicted result.

2.4.1 Sensing Algorithm

First, we state the 2-D sensing scheme as follows. The detection for the hidden PU channel state $\hat{C}_t$ and the estimation for the PU energy level $\hat{E}_t$ could be jointly achieved by maximizing the conditional probability of the current hidden states, i.e., $(\hat{C}_t, \hat{E}_t)$ is decided as

$$\hat{C}_t, \hat{E}_t = \arg\max_{c_t, e_t} \quad P(C_t = c_t, E_t = e_t | U^t, Y^t). \tag{2.11}$$

In CR, such an estimation result further decides the activity of SU: When $\hat{C}_t = 0$, SU considers the channel as inactive and utilizes it; when $\hat{C}_t = 1$, SU considers the channel as occupied and stays silent or transmits carefully based on the knowledge of $\hat{E}_t$.

In fact, the joint probability $P(U^t, Y^t)$ is the proportional constant between $P(C_t, E_t | U^t, Y^t)$ and $P(C_t, E_t, U^t, Y^t)$, i.e., by Bayes rules there is a relationship

$$\begin{aligned} P(C_t, E_t | U^t, Y^t) &= P(C_t, E_t, U^t, Y^t) / P(U^t, Y^t) \\ &\propto P(C_t, E_t, U^t, Y^t). \end{aligned} \tag{2.12}$$

This result implies that the estimated hidden states $\hat{C}_t$, $\hat{E}_t$ also maximize the joint probability $P(C_t, E_t, U^t, Y^t)$. In [23], it is demonstrated that such estimated results are based on the Maximum A Posteriori (MAP) principle, which is considered as an optimal estimate to minimize the Bayes risk for a “hit-or-miss" cost function. Next we discuss how to evaluate and solve (2.11).

We first give the procedure to recursively calculate $P(C_t, E_t | U^t, Y^t)$ in (2.11), which is divided into the following three steps. Since the Markovian property could not be easily applied to decompose the conditional probability of the hidden states for recursive calculations, in the first step, the probability of the current observation given the past

$P(U_t, Y_t|U^{t-1}, Y^{t-1})$ is used to multiply the target conditional probability $P(C_t, E_t|U^t, Y^t)$, such that the production after multiplication could be directly decomposed and computed as

$$
\begin{aligned}
& P(U_t, Y_t|U^{t-1}, Y^{t-1})P(C_t, E_t|U^t, Y^t) \\
&= \sum_{C_{t-1}} \sum_{E_{t-1}} P(C_t, C_{t-1}, E_t, E_{t-1}, U_t, Y_t|U^{t-1}, Y^{t-1}) \\
&\overset{(a)}{=} \sum_{C_{t-1}} \sum_{E_{t-1}} P(Y_t|C_t, E_t)P(U_t|E_t) \\
&\qquad\qquad \cdot P(C_t, C_{t-1}, E_t, E_{t-1}|U^{t-1}, Y^{t-1}) \\
&= \underbrace{P(Y_t|C_t, E_t)P(U_t|E_t)}_{\text{corrector}} \cdot \\
&\underbrace{\sum_{C_{t-1}} \sum_{E_{t-1}} P(C_t|C_{t-1}, E_t)P(E_t|E_{t-1})P(C_{t-1}, E_{t-1}|U^{t-1}, Y^{t-1}),}_{\text{predictor}} \qquad (2.13)
\end{aligned}
$$

where (a) is obtained with the Markovian property and Bayes rules, and the current observations Y_t and U_t work as correctors to improve the prediction of the current states based on the past. Inside (2.13), the probabilities of the corrector part could be computed by the emission matrices from the parameter vector η', the first two probabilities of the predictor could be computed by the transition matrices in η', and the last probability is the previous conditional probability of the hidden states, which is obtained from the previous step in our recursive algorithm. Note that the recursion procedure is exactly connected by the last probability term inside (2.13), as it is the only term dependent on the previous information.

After we compute (2.13), to obtain the objective in (2.11), the value of $P(U_t, Y_t|U^{t-1}, Y^{t-1})$ has to be calculated in order to divide it out from (2.13). Thus the second step calculates $P(U_t, Y_t|U^{t-1}, Y^{t-1})$: As the summation of $P(C_t, E_t|U^t, Y^t)$ over all possible hidden

states equals one, the value of $P(U_t, Y_t|U^{t-1}, Y^{t-1})$ could be calculated as

$$\begin{aligned} & P(U_t, Y_t|U^{t-1}, Y^{t-1}) \\ &= P(U_t, Y_t|U^{t-1}, Y^{t-1}) \sum_{C_t} \sum_{E_t} P(C_t, E_t|U^t, Y^t) \\ &= \sum_{C_t} \sum_{E_t} P(U_t, Y_t|U^{t-1}, Y^{t-1}) P(C_t, E_t|U^t, Y^t). \end{aligned} \tag{2.14}$$

Although the individual probability terms inside (2.14) could not be directly computed, the multiplication of the two probabilities actually is already the result we obtained in (2.13) in our first step. Then marginalizing over all possible (C_t, E_t) pairs leads to the conditional probability of current observations $P(U_t, Y_t|U^{t-1}, Y^{t-1})$, as we desire.

At the last step, the conditional probability of hidden states $P(C_t, E_t|U^t, Y^t)$ is calculated by dividing (2.13) with (2.14), which actually could be treated as a normalization procedure. It is worth noting that when $t = 1$, which is the starting point of our recursion algorithm, the multiplication corresponding to (2.13) is expressed as

$$\begin{aligned} & P(U_1, Y_1) P(E_1, C_1|U_1, Y_1) \\ &= P(E_1, C_1, U_1, Y_1) \\ &= P(U_1, Y_1|E_1, C_1) P(E_1, C_1) \\ &= P(U_1|E_1) P(Y_1|E_1, C_1) P(C_1|E_1) P(E_1), \end{aligned} \tag{2.15}$$

where the first two probabilities are from the emission matrices in the parameter vector η' and the last two probabilities are from the initial probabilities $\boldsymbol{\pi}^C$ and π^E inside η'. Using the same method as in (2.14) to obtain the value of $P(U_1, Y_1)$ and dividing the multiplication result in (2.15) by $P(U_1, Y_1)$, the conditional probability of the hidden states at $t = 1$ could be computed.

After obtaining the conditional probability of the hidden states $P(C_t, E_t|U^t, Y^t)$, the maximization procedure in (2.11) could be solved by a simple 2-D exhaustive search, since the cardinality of the possible hidden states are within a finite range.

In summary, the algorithm to compute the conditional probability $P(C_t, E_t|U^t, Y^t)$ is summarized below.

ALGORITHM TO COMPUTE $P(C_t, E_t|U^t, Y^t)$

Input: η, U^t, Y^t

for $i \in \mathscr{L}$

  for $j \in \mathscr{C}$

    $P(E_1 = i, C_1 = j|U_1, Y_1) = \frac{Q(1)\pi_{ij}^c \pi_i^E}{\sum_{i\in\mathscr{L}}\sum_{j\in\mathscr{C}} Q(1)\pi_{ij}^c \pi_i^E}$

  end

end

for $k = 2, \cdots, t$

  for $i \in \mathscr{L}$

    for $j \in \mathscr{C}$

      $P(E_k = i, C_k = j|U^k, Y^k) =$

$$\frac{Q(k)\sum_{l\in\mathscr{L}}\sum_{m\in\mathscr{C}} b_{mj,i}a_{li}P(C_{k-1}=m,E_{k-1}=l|U^{k-1},Y^{k-1})}{\sum_{i\in\mathscr{L}}\sum_{j\in\mathscr{C}} Q(k)\sum_{l\in\mathscr{L}}\sum_{m\in\mathscr{C}} b_{mj,i}a_{li}P(C_{k-1}=m,E_{k-1}=l|U^{k-1},Y^{k-1})}$$

    end

  end

end

where $Q(k)$ stands for $d_{iU_k}\frac{\exp(-\frac{(Y_k-\mu_{ji})^2}{2\sigma_{ji}^2})}{\sqrt{2\pi\sigma_{ji}^2}}$.

Although the pseudo code to compute $P(C_t, E_t|U^t, Y^t)$ given above indicates the computation of $P(C_t, E_t|U^t, Y^t)$ from time 1 up to time t, as this algorithm runs online, in each time slot the SU only has to compute the current joint probability based on the probability

of the last time slot, i.e., $P(C_{t-1}, E_{t-1}|U^{t-1}, Y^{t-1})$ is already known in time slot t. Therefore, the computational complexity could be controlled within an acceptable range at each time slot. Specifically, the algorithm to compute the joint probability in each time slot leads to complexity $O(N^2M^2)$, and the exhaustive search for the estimated states leads to complexity $O(NM)$.

Remark: For an energy harvesting based PU, there is always a collection of energy states that are not enough to support reliable PU transmissions. Let $\mathscr{L}_0 \subset \mathscr{L}$ denote the subset that contains all the energy states from $\mathscr{L}$ that are insufficient for reliable PU transmissions and $\mathscr{L}_1$ be the complement of $\mathscr{L}_0$. From the relationship given in (2.1), when $\hat{E}_t \in \mathscr{L}_1$, the estimated channel state $\hat{C}_t$ could be either 1 or 0. However, when $\hat{E}_t \in \mathscr{L}_0$, $\hat{C}_t$ could only be zero. This is due to the fact that the joint probability of $C_t = 1$ and $E_t \in \mathscr{L}_0$ is always equal to zero. As such, the optimal solution for (2.11) could not be $(\hat{C}_t = 1, \hat{E}_t \in \mathscr{L}_0)$, which could be eliminated from the searching space in solving (2.11) to reduce the computational complexity.

2.4.2 Performance Comparison via Mutual Information

Compared with our proposed 2-D sensing scheme, the traditional PU detection methods do not take the spatial correlation of energy harvesting processes into consideration, i.e., for traditional methods the only available observation is Y^t. It implies that under the same MAP principle, the traditional sensing scheme is basically solving

$$\hat{C}_t, \hat{E}_t = \arg\max_{c_t, e_t} P(C_t = c_t, E_t = e_t, Y^t). \tag{2.16}$$

The mutual information between the hidden states and observations could be used to

quantify the estimation performance. In our setup, it can be evaluated as

$$I(C_t, E_t; Y^t, U^t) = I(C_t, E_t; Y^t) + I(C_t, E_t; U^t|Y^t), \tag{2.17}$$

where $I(C_t, E_t; Y^t)$ is the mutual information quantitying the performance of the traditional detection methods, and $I(C_t, E_t; U^t|Y^t)$ is the information gained by additionally considering the power domain information. The mutual information gain is zero if and only if (C_t, E_t) are independent of U^t conditioned on Y^t, which is not the case in our system, such that the mutual information gain is always greater than zero in our method.

2.5 Parameter Estimation

In this section, we discuss the proposed algorithm for parameter estimation. We need to first define several intermediate variables, which are used for algorithm implementation, then derive the algorithm based on the traditional EM algorithm. Note that the proposed algorithm is off-line, with observations U^T, Y^T for training.

2.5.1 Intermediate Variables

For U^t and E_t, define

$$\alpha_i^U(t) \triangleq P(U^t, E_t = i|\eta), \quad t = 1, 2, 3, \cdots, T, \tag{2.18}$$

which is the joint probability of getting the observation U^t and having the PU energy state at time t (E_t) equal i, conditioned on η, with η defined in Section III. Similarly, with the communication channel output observation, define

$$\alpha_{ij}^Y(t) \triangleq P(Y^t, C_t = i, E_t = j|\eta), \quad t = 1, 2, 3, \cdots, T, \tag{2.19}$$

which is the joint probability of getting the observations Y^t and having the channel state C_t as i and the energy level E_t as j, conditioned on η as well.

Since we estimate two hidden states with different observations, the information carried by the joint observations should also be considered. Therefore, define

$$\alpha_{ij}^{U,Y}(t) \triangleq P(U^t, Y^t, C_t = i, E_t = j|\eta), \quad t = 1, 2, 3, \cdots, T, \tag{2.20}$$

as the intermediate variables for joint observations, which indicates the probability of getting the observations U^t, Y^t, with $C_t = i$ and $E_t = j$.

As both $\{E_t\}$ and $\{C_t\}$ are Markov processes, with Markov property, the defined intermediate variable collections α^U, α^Y and $\alpha^{U,Y}$ can be quantified recursively. For α^U, the probability at time $t + 1$ is given as

$$\alpha_j^U(t+1) = f(U_{t+1}|j) \sum_{i \in \mathscr{L}} \alpha_i^U(t) a_{ij} \quad t = 2, \cdots, T, \tag{2.21}$$

which is traced to $\alpha_i^U(1) = \pi_{Ei} d_{iU_1}$ back at the initial state. This process is usually called a forward recursion, in which the next-time joint probability equals to the current joint probability weighted by the transition probability, followed by marginalization and multiplication with the emission probability. It can be interpreted that the forward recursion at each time equals to a predictor multiplied by a corrector. Similarly, the forward procedures for α^Y and $\alpha^{U,Y}$ are respectively given as

$$\begin{aligned} \alpha_{ij}^Y(1) &= \pi_{Cji} \pi_{Ej} f(Y_1|i, j), \\ \alpha_{ij}^Y(t+1) &= f(Y_{t+1}|i, j) \sum_{l \in \mathscr{L}} \sum_{m \in \mathscr{Y}} \alpha_{ml}^Y(t) a_{lj} b_{mi,j}, \\ & \qquad t = 2, \cdots, T. \end{aligned} \tag{2.22}$$

and

$$\alpha_{ij}^{U,Y}(1) = \pi_{Cji}\pi_{Ej}d_{jU_1}f(Y_1|i,j),$$
$$\alpha_{ij}^{U,Y}(t+1) = d_{jU_1}f(Y_{t+1}|i,j)\sum_{l\in\mathscr{L}}\sum_{m\in\mathscr{C}}\alpha_{ml}^{U,Y}(t)a_{lj}b_{mi,j},$$
$$t = 2,\cdots,T. \tag{2.23}$$

In addition to the forward recursion, backward recursions are then used to measure the probability of getting future output up to time T given the state at time t. Define

$$\beta_i^U(t) \triangleq P(U_{t+1}, U_{t+2},\cdots,U_T|E_t = i,\eta),$$
$$\beta_{ij}^Y(t) \triangleq P(Y_{t+1}, Y_{t+2},\cdots,Y_T|C_t = i, E_t = j,\eta),$$
$$\beta_{ij}^{U,Y}(t) \triangleq$$
$$P(U_{t+1}, Y_{t+1}, U_{t+2}, Y_{t+2},\cdots,U_T, Y_T|C_t = i, E_t = j,\eta),$$
$$t = 1,2,3,\cdots,T, \tag{2.24}$$

as the probabilities of getting the observation sequence from $t+1$ to T, conditioned on the hidden states at time t. For the SU energy state, $\beta_i^U(t)$ is calculated as

$$\beta_i^U(t) = \sum_{j\in\mathscr{L}}\beta_j^U(t+1)a_{ij}d_{jU_{t+1}}, \quad t = 1,\cdots,T-1, \tag{2.25}$$

with $\beta_i^U(T) = 1,\ \forall i$. This indicates that the probability of getting the future observations up to time T conditioned on the current hidden states equals to such a conditional probability at the next time step multiplying the transition and emission probabilities of all possible current latent states. As it can be seen, the backward recursion uses the future observations to smooth the current inference. Meanwhile, the calculations for $\beta_{ij}^Y(t)$ and

$\beta_{ij}^{U,Y}(t)$ could be expressed as

$$\begin{aligned}\beta_{ij}^{Y}(t) &= \sum_{m\in\mathscr{C}}\sum_{l\in\mathscr{L}} \beta_{ml}^{Y}(t+1) f(Y_{t+1}|m,l) a_{jl} b_{im,l},\\ \beta_{ij}^{U,Y}(t) &= \sum_{m\in\mathscr{C}}\sum_{l\in\mathscr{L}} \beta_{ml}^{U,Y}(t+1) f(Y_{t+1}|m,l) d_{lU_{t+1}} a_{jl} b_{im,l},\\ &\qquad\qquad t = 1,\cdots,T-1, \end{aligned} \tag{2.26}$$

with $\beta_{ij}^{Y}(T) = 1, \beta_{ij}^{U,Y}(T) = 1,\ \forall i,j$.

After obtaining the expressions for forward and backward recursions, we need to define two update variables for each observation process to make the implementation of the proposed parameter learning algorithm more straightforward. One update variable is defined to consider the joint probability of getting the entire observations from time 1 to time T and the hidden state at time t; the other one considers the joint probability of getting the entire observations and the hidden states at time t and $t+1$. Specfically, for the SU energy observation process, define

$$\begin{aligned}\gamma_i^U(t) &\triangleq P(U^T, E_t = i|\eta) = \alpha_i^U(t)\beta_i^U(t), \quad t = 1,\cdots,T\\ \varepsilon_{ij}^U(t) &\triangleq P(U^T, E_t = i, E_{t+1} = j|\eta)\\ &= \alpha_i^U(t) a_{ij} \beta_j^U(t+1) d_{jU_{t+1}}, \quad t = 1,\cdots,T-1. \end{aligned} \tag{2.27}$$

Clearly, the update variables above combine the forward and backward variables with transition and emission probabilities. For the channel observation and the joint observation

processs, variables are defined as

$$\begin{aligned}
\gamma_{ij}^{Y}(t) &\triangleq P(Y^T, C_t = i, E_t = j|\eta) = \alpha_{ij}^{Y}(t)\beta_{ij}^{Y}(t), \\
\varepsilon_{ij,ml}^{Y}(t) &\triangleq P(Y^T, C_t = i, E_t = j, C_{t+1} = m, E_{t+1} = l|\eta) \\
&= \alpha_{ij}^{Y}(t)a_{jl}b_{im,l}\beta_{ml}^{Y}(t+1)f(Y_{t+1}|m,l).
\end{aligned} \tag{2.28}$$

and

$$\begin{aligned}
\gamma_{ij}^{U,Y}(t) &\triangleq P(U^T, Y^T, C_t = i, E_t = j|\eta) = \alpha_{ij}^{U,Y}(t)\beta_{ij}^{U,Y}(t), \\
\varepsilon_{ij,ml}^{U,Y}(t) &\triangleq P(U^T, Y^T, C_t = i, E_t = j, C_{t+1} = m, E_{t+1} = l|\eta) \\
&= \alpha_{ij}^{U,Y}(t)a_{jl}b_{im,l}\beta_{ml}^{U,Y}(t+1)f(Y_{t+1}|m,l)d_{lU_{t+1}}.
\end{aligned} \tag{2.29}$$

Again, the reason to define these intermediate variables is that they will be used to implement the proposed parameter estimation algorithm for HIMM, which is introduced in the following subsection.

2.5.2 Parameter Estimation Algorithm

The EM algorithm [24] is adopted here used to iteratively find the unknown parameters that maximize the likelihood in a statistical model, in which the latent variables exist. Here, E stands for expectation and M stands for maximization, and the algorithm alternates between the E and M procedures. The current expectation step creates an expected log-likelihood function, which is averaged over the latent variables given the estimated parameters in the previous step; then the maximization step is designed to find a new parameter that maximizes the function of log-likelihood formulated in the current expectation step. The algorithm continues until the results converge (when the difference between the results of two consecutive iterations is within an acceptable region). As proved before,

the expected log-likelihood is non-decreasing during the iterations and the algorithm converges; however, it is not guaranteed to converge to a global optimum. This is due to the fact that although the maximization step is seeking the global optimum for the expected joint probability of hidden states and observations, the results may not be the global optimum for the likelihood function, which is from the conditional probability of observations. Detailed analysis about the EM algorithm convergence could be found in [25].

2.5.2.1 Expectation Step

In the expectation step, we need to consider the average of the log-likelihood function over the latent Markov states. If, in an extreme case, the Markov states are known, then the E step could be decomposed into T number of subproblems since the temporal dependence of the Markov chain contains no additional information for the expectation of the log-likelihood function. Let $\eta^{(k-1)}$ denote the parameter estimated at the previous iteration; and let $O = \{U^T, Y^T\}$ stand for the collection of observations. The expectation of log-likelihood can then be expressed as

$$\begin{aligned} L(\eta; \eta^{(k-1)}) &= \mathbb{E}_{C^T, E^T}\{\log P(E^T, C^T, O|\eta)\} \\ &= \sum_{E^T \in \mathscr{L}^T} \sum_{C^T \in \mathscr{C}^T} P(E^T, C^T|O, \eta^{(k-1)}) \log P(E^T, C^T, O|\eta) \\ &= \sum_{E^T \in \mathscr{L}^T} \sum_{C^T \in \mathscr{C}^T} \frac{P(E^T, C^T, O|\eta^{(k-1)})}{P(O|\eta^{(k-1)})} \log P(E^T, C^T, O|\eta). \end{aligned} \tag{2.30}$$

Inside the equation, $\sum_{E^T \in \mathscr{L}^T}$ and $\sum_{C^T \in \mathscr{C}^T}$ are abbreviations for $\sum_{E_1 \in \mathscr{L}} \sum_{E_2 \in \mathscr{L}} \cdots \sum_{E_T \in \mathscr{L}}$ and $\sum_{C_1 \in \mathscr{C}} \sum_{C_2 \in \mathscr{C}} \cdots \sum_{C_T \in \mathscr{C}}$.

The achieved expected log-likelihood is utilized in the Maximization step to find a better parameter estimate, which should not decrease the averaged log-likelihood.

2.5.2.2 *Maximization Step*

The objective in the M step is to achieve a better parameter estimate $\eta^{(k)}$, which is consisted of $\{\pi^{E(k)}, \boldsymbol{\pi}^{C(k)}, \mathbf{A}^{(k)}, \mathbf{B}^{(k)}, \mathbf{D}^{(k)}, \boldsymbol{\mu}^{(k)}, \boldsymbol{\sigma}^{2(k)}\}$. Inside $L(\eta; \eta^{(k-1)})$, since the denominator $P(O|\eta^{(k-1)})$ is a constant with respect to $\eta^{(k)}$, the M step only needs to maximize

$$L(\eta; \eta^{(k-1)})' = \sum_{E^T \in \mathscr{L}^T} \sum_{C^T \in \mathscr{C}^T} P(E^T, C^T, O|\eta^{(k-1)}) \log P(E^T, C^T, O|\eta) \tag{2.31}$$

over the parameter. However, based on the structure of Fig. 2.1, the log-likelihood can be decomposed as

$$\begin{aligned} &\log P(E^T, C^T, U^T, Y^T|\eta) \\ &= \log\{P(U^T, Y^T|E^T, C^T, \eta)P(E^T, C^T|\eta)\} \\ &= \log\{P(U^T|E^T, \eta)P(Y^T|E^T, C^T, \eta)P(C^T|E^T, \eta)P(E^T|\eta)\} \\ &= \log P(U^T|E^T, \eta) + \log P(Y^T|E^T, C^T, \eta) \\ &\qquad + \log P(C^T|E^T, \eta) + \log P(E^T|\eta), \end{aligned} \tag{2.32}$$

where each element depends on different optimization decision variables and there are no decision variables that have been shared by any two elements above. This implies that the maximization over the entire log-likelihood could be factorized into several independent subproblems, while the summation over all the optimal subproblem results leads to the optimal result of the entire problem. Mathematically, it indicates

$$
\begin{aligned}
&\max_{\eta} L(\eta;\eta^{(k-1)})' \\
&= \max_{\mathbf{D}} \sum_{E^T\in\mathscr{L}^T}\sum_{C^T\in\mathscr{C}^T} P(E^T,C^T,O|\eta^{(k-1)})\log P(U^T|E^T,\eta) \\
&+ \max_{\pi^E,\mathbf{A}} \sum_{E^T\in\mathscr{L}^T}\sum_{C^T\in\mathscr{C}^T} P(E^T,C^T,O|\eta^{(k-1)})\log P(E^T|\eta) \\
&+ \max_{\boldsymbol{\pi}^C,\mathbf{B}} \sum_{E^T\in\mathscr{L}^T}\sum_{C^T\in\mathscr{C}^T} P(E^T,C^T,O|\eta^{(k-1)})\log P(C^T|E^T,\eta) \\
&+ \max_{\boldsymbol{\mu},\boldsymbol{\sigma}^2} \sum_{E^T\in\mathscr{L}^T}\sum_{C^T\in\mathscr{C}^T} P(E^T,C^T,O|\eta^{(k-1)})\log P(Y^T|E^T,C^T,\eta). \qquad (2.33)
\end{aligned}
$$

By solving the four subproblems above, the optimal solution for $\eta^{(k)}$ could be achieved, which is then used for the next iteration if the convergence requirement is not yet satisfied.

2.5.2.2.1 Optimal $\mathbf{D}^{(k)}$ First, let us focus on learning the parameter $\mathbf{D}$, which is the emission probability of the PU energy arrival process. Since $\mathbf{D} = \{d_{ij}\}$, learning d_{ij} for all $i,j \in \mathscr{L}$ is sufficient to have the estimated value of $\mathbf{D}$. Before moving on to the maximization procedure, the objective function could be further simplified as

$$
\max_{\mathbf{D}} \sum_{E^T\in\mathscr{L}^T} P(E^T,U^T|\eta^{(k-1)})\log P(U^T|E^T,\eta). \qquad (2.34)
$$

This is due to the fact that the hidden channel states C^T could be marginalized by summing over all the possible outcome, and also

$$
\begin{aligned}
&P(E^T,U^T,Y^T|\eta^{(k-1)}) \\
&= P(U^T,E^T|Y^T,\eta^{(k-1)})P(Y^T|\eta^{(k-1)}) \\
&= P(U^T|E^T,\eta^{(k-1)})P(E^T|Y^T,\eta^{(k-1)})P(Y^T|\eta^{(k-1)}),
\end{aligned}
$$

where $P(Y^T|\eta^{(k-1)})$, $P(E^T|Y^T,\eta^{(k-1)})$ are constant values with respect to $\mathbf{D}$. Besides, $P(E^T|\eta^{(k-1)})$ is positive and independent of the decision variable $\mathbf{D}$, such that its multiplication with the objective function will not affect the estimation of $\mathbf{D}$. As such we consider joint probability $P(E^T,U^T|\eta^{(k-1)})$ instead of the conditional probability $P(U^T|E^T,\eta^{(k-1)})$ without influencing the maximization step.

On the other hand, since d_{ij} is the emission probability, the objective function needs to be decomposed from a joint probability into T independent emission probabilities, in order to estimate the value of d_{ij}. The decomposition procedure is

$$\begin{aligned}
&\sum_{E^T\in\mathscr{L}^T} P(E^T,U^T|\eta^{(k-1)})\log P(U^T|E^T,\eta)\\
&=\sum_{E^T\in\mathscr{L}^T} P(E^T,U^T|\eta^{(k-1)})\log\prod_{t=1}^{T}P(U_t|E_t,\eta)\\
&=\sum_{t=1}^{T}\sum_{i=0}^{L-1}\sum_{E^T\in\mathscr{L}^T} P(U^T,E^T|\eta^{(k-1)})\delta(E_t-i)\log P(U_t|E_t=i,\eta)\\
&=\sum_{t=1}^{T}\sum_{i=0}^{L-1} P(U^T,E_t=i|\eta^{(k-1)})\log P(U_t|E_t=i,\eta)\\
&=\sum_{t=1}^{T}\sum_{i=0}^{L-1} P(U^T,E_t=i|\eta^{(k-1)})\log\sum_{j=0}^{L-1}\delta(U_t-j)d_{ij}. \qquad (2.35)
\end{aligned}$$

Inside the equation, $\delta(x)$ is the indicator function such that $\delta(x)=1$ when $x=0$ and $\delta(x)=0$ for any other values of x. Since d_{ij} denotes the emission probability of the PU energy level, for any i the summation of d_{ij} over all the possible outputs equals to one,

i.e., $\sum_{j=0}^{L-1} d_{ij} = 1, \forall i$. Therefore, the maximization problem is finalized as

$$\begin{aligned} \max_{\mathbf{D}} & \sum_{t=1}^{T} \sum_{i=0}^{L-1} P(U^T, E^t = i|\eta^{(k-1)}) \log \sum_{j=0}^{L-1} \delta(U_t - j) d_{ij} \\ \text{s.t.} & \sum_{j=0}^{L-1} d_{ij} = 1 \end{aligned} \tag{2.36}$$

This is a typical optimization problem to find a maximum of the objective function subject to an equality constraint, which could be solved by the Lagrange multiplier method [26]. After calculations, the optimal result for $d_{ij}^{(k)}$ is given as

$$\begin{aligned} d_{ij}^{(k)} &= \frac{\sum_{t=1}^{T} \delta(U_t - j) P(U^T, E_t = i|\eta^{(k-1)})}{\sum_{t=1}^{T} P(U^T, E_t = i|\eta^{(k-1)})} \\ &= \frac{\sum_{t=1}^{T} \delta(U_t - j) \gamma_i^U(t)}{\sum_{t=1}^{T} \gamma_i^U(t)}, \quad i, j \in \mathscr{L}. \end{aligned} \tag{2.37}$$

which implies that the estimated probability of emitting j at state i for the PU energy level equals the number of times that the output from the latent state i is j divided by the total number of times that the latent state is i.

2.5.2.2.2 Optimal $\pi^{E(k)}$, $\mathbf{A}^{(k)}$ The objective function for this subproblem could be further decomposed into two parts, in which the decision variables are π^E and $\mathbf{A} = \{a_{i,j}\}$,

respectively. The decomposed objective function is

$$\begin{aligned}
&\sum_{E^T \in \mathscr{L}^T} \sum_{C^T \in \mathscr{C}^T} P(E^T, C^T, U^T, Y^T | \eta^{(k-1)}) \log P(E^T | \eta) \\
&= \sum_{E^T \in \mathscr{L}^T} P(E^T, U^T, Y^T | \eta^{(k-1)}) \log P(E^T | \eta) \\
&= \sum_{E^T \in \mathscr{L}^T} P(E^T, U^T, Y^T | \eta^{(k-1)}) \\
&\qquad \times \left\{ \log \pi^E + \sum_{t=2}^{T} \log a_{E_{t-1} E_t} \right\}. \qquad (2.38)
\end{aligned}$$

Clearly, the optimization procedures to maximize the objective function over π^E and $\mathbf{A}$ could be treated as two independent processes. Therefore, after factorizing the joint probablities into T independent probabilities based on the Markovian property, the problems become

$$\begin{aligned}
&\max_{\pi_i^E} \sum_{i \in \mathscr{L}} P(U^T, Y^T, E_1 = i | \eta^{(k-1)}) \log \pi_i^E \\
&\quad \text{s.t.} \sum_{i=0}^{L-1} \pi_i^E = 1, \quad i \in \mathscr{L}, \qquad (2.39)
\end{aligned}$$

and

$$\begin{aligned}
&\max_{a_{ij}} \sum_{t=2}^{T} \sum_{i \in \mathscr{L}} \sum_{j \in \mathscr{L}} P(U^T, Y^T, E_{t-1} = i, E_t = j | \eta^{(k-1)}) \log a_{ij} \\
&\quad \text{s.t.} \sum_{j=0}^{L-1} a_{ij} = 1, \quad i \in \mathscr{L}. \qquad (2.40)
\end{aligned}$$

By solving the subproblems with the Lagrange multipler method, the estimated values of π^E and $\mathbf{A}$ for step k are

$$\begin{aligned}\pi_i^{E(k)} &= \frac{P(U^T, Y^T, E_1 = i|\eta^{(k-1)})}{P(U^T, Y^T|\eta^{(k-1)})} \\ &= \frac{\sum\limits_{j\in\mathscr{C}} \gamma_{ji}^{U,Y}(1)}{\sum\limits_{j\in\mathscr{C}}\sum\limits_{i\in\mathscr{L}} \gamma_{ij}^{U,Y}(1)},\end{aligned} \tag{2.41}$$

and

$$\begin{aligned}a_{ij}^{(k)} &= \frac{\sum\limits_{t=2}^{T} P(U^T, Y^T, E_{t-1} = i, E_t = j|\eta^{(k-1)})}{\sum\limits_{t=2}^{T} P(U^T, Y^T, E_{t-1} = i|\eta^{(k-1)})} \\ &= \frac{\sum\limits_{t=2}^{T}\sum\limits_{m,l\in\mathscr{C}} \varepsilon_{mi,lj}^{U,Y}(t-1)}{\sum\limits_{t=2}^{T}\sum\limits_{m\in\mathscr{C}} \gamma_{mi}^{U,Y}(t-1)},\end{aligned} \tag{2.42}$$

for any $i \in \mathscr{L}$. It can be seen that as channel and energy observations contain the information of the PU energy level, they both contribute to the estimation of the PU energy transition matrix.

2.5.2.2.3 Optimal $\boldsymbol{\pi}^{C(k)}, \mathbf{B}^{(k)}$ Following the previous arguments, the objective function here can also be decomposed as

$$\begin{aligned}&\sum_{E^T\in\mathscr{L}^T}\sum_{C^T\in\mathscr{C}^T} P(U^T, Y^T, C^T, E^T|\eta^{(k-1)}) \log P(C^T|E^T, \eta) \\ =& \sum_{E^T\in\mathscr{L}^T}\sum_{C^T\in\mathscr{C}^T} P(U^T|E^T, \eta^{(k-1)}) P(E^T, C^T, Y^T|\eta^{(k-1)}) \\ &\qquad \times \left\{\log \pi^C + \sum_{t=1}^{T} \log b_{C_{t-1}E_t,C_t}\right\}.\end{aligned} \tag{2.43}$$

Since $P(U^T|E^T, \eta^{(k-1)})$ is not changing with respect to $\boldsymbol{\pi}^C$ and $\mathbf{B}$, it can be neglected for estimating these parameters. After factorizing the objective function into T independent elements, the optimal parameters are obtained as

$$\begin{aligned}\pi_{ij}^{C(k)} &= \frac{P(Y^T, E_1 = i, C_1 = j|\eta^{(k-1)})}{P(Y^T, E_1 = i|\eta^{(k-1)})} \\ &= \frac{\gamma_{ji}^Y(1)}{\sum\limits_{j\in\mathscr{C}} \gamma_{ji}^Y(1)}, \quad i \in \mathscr{L}, j \in \mathscr{C},\end{aligned} \tag{2.44}$$

and

$$b_{ij,k}^{(k)} = \frac{\sum\limits_{t=2}^{T} P(Y^T, C_{t-1} = i, E_t = k, C_t = j|\eta^{(k-1)})}{\sum\limits_{t=2}^{T} P(Y^T, C_{t-1} = i, E_t = k|\eta^{(k-1)})}$$

$$= \frac{\sum\limits_{t=2}^{T}\sum\limits_{m=0}^{L-1} \varepsilon_{im,jk}^Y(t-1)}{\sum\limits_{t=2}^{T}\sum\limits_{m=0}^{L-1}\sum\limits_{j\in\mathscr{C}} \varepsilon_{im,jk}^Y(t-1)}, \quad i, j \in \mathscr{C}, k \in \mathscr{L}. \tag{2.45}$$

2.5.2.2.4 Optimal $\boldsymbol{\mu}$, $\boldsymbol{\sigma}^2$ The channel output $\{Y_t\}$ is a continous states process, such that given the latent states of channel and PU energy, $\{Y_t\}$ follows a Gaussian distribution for any t. Learning $\boldsymbol{\mu} = \{\mu_{ij}\}$ and variance $\boldsymbol{\sigma}^2 = \{\sigma_{ij}^2\}$ for $i \in \mathscr{C}$ and $j \in \mathscr{L}$ of the conditional Gaussian distribution could determine the emission probabilities of channel and PU energy states.

Conditioned on the hidden PU energy process, the observation of the SU energy level is independent of the hidden channel state and the channel output; therefore, U^T does not contribute to the estimation of the channel output parameters. Since given the latent states,

the channel output $\{Y_t\}$ is independent over time, we have

$$\begin{aligned}
&\sum_{E^T}\sum_{C^T} P(Y^T, C^T, E^T|\eta^{(k-1)}) \log P(Y^T|E^T, C^T, \eta) \\
&= \sum_{E^T}\sum_{C^T} P(Y^T, C^T, E^T|\eta^{(k-1)}) \log \prod_{t=1}^{T} P(Y_t|E_t, C_t, \eta) \\
&= \sum_{t=1}^{T}\sum_{i\in\mathscr{C}}\sum_{j\in\mathscr{L}} P(Y^T, C_t = i, E_T = j|\eta^{(k-1)}) \\
&\qquad\qquad \times \log P(Y_t|C_t = i, E_t = j, \eta), \qquad (2.46)
\end{aligned}$$

where $P(Y_t|C_t = i, E_t = j, \eta) \sim N(Y_t; \mu_{ij}, \sigma_{ij}^2)$ and $\sum_{E^T}, \sum_{C^T}$ stand for $\sum_{E^T\in\mathscr{L}^T}, \sum_{C^T\in\mathscr{C}^T}$ respectively. Since this is an unconstrained convex optimization problem with a differentiable objective function, letting the first derivative of the objective function equals to zero and solving the rest will provide the optimal point for μ_{ij} and σ_{ij}^2. Then the estimated values of these parameters are

$$\mu_{ij}^{(k)} = \frac{\sum_{t=1}^{T} P(Y^T, C_t = i, E_t = j|\eta^{(k-1)}) Y_t}{\sum_{t=1}^{T} P(Y^T, C_t = i, E_t = j|\eta^{(k-1)})} = \frac{\sum_{t=1}^{T} \gamma_{ij}^Y(t) Y_t}{\sum_{t=1}^{T} \gamma_{ij}^Y(t)} \qquad (2.47)$$

and

$$\begin{aligned}
\sigma_{ij}^{2(k)} &= \frac{\sum_{t=1}^{T} P(Y^T, C_t = i, E_t = j|\eta^{(k-1)})(Y_t - \mu_{ij}^{(k)})^2}{\sum_{t=1}^{T} P(Y^T, C_t = i, E_t = j|\eta^{(k-1)})} \\
&= \frac{\sum_{t=1}^{T} \gamma_{ij}^Y(t)(Y_t - \mu_{ij}^{(k)})^2}{\sum_{t=1}^{T} \gamma_{ij}^Y(t)}. \qquad (2.48)
\end{aligned}$$

However, the expressions above only work for $i = 1$, $\forall j \in \mathscr{L}$. Since when $i = 0$, PU

does not transmit signal and the channel output does not contain any information about the PU energy level. As we mentioned, in this case μ_{ij} and variance σ_{ij}^2 are identical for any $j \in \mathscr{L}$. Then the estimated values of these parameters are

$$\begin{aligned}\mu_{ij}^{(k)} &= \frac{\sum_{t=1}^{T} P(Y^T, C_t = i|\eta^{(k-1)})Y_t}{\sum_{t=1}^{T} P(Y^T, C_t = i|\eta^{(k-1)})} \\ &= \frac{\sum_{t=1}^{T}\left\{\sum_{j=0}^{L-1}\gamma_{ij}^{Y}(t)\right\}Y_t}{\sum_{t=1}^{T}\sum_{j=0}^{L-1}\gamma_{ij}^{Y}(t)}\end{aligned} \tag{2.49}$$

and

$$\begin{aligned}\sigma_{ij}^{2(k)} &= \frac{\sum_{t=1}^{T} P(Y^T, C_t = i|\eta^{(k-1)})(Y_t - \mu_{ij}^{(k)})^2}{\sum_{t=1}^{T} P(Y^T, C_t = i|\eta^{(k-1)})} \\ &= \frac{\sum_{t=1}^{T}\left\{\sum_{j=0}^{L-1}\gamma_{ij}^{Y}(t)\right\}(Y_t - \mu_{ij}^{(k)})^2}{\sum_{t=1}^{T}\sum_{j=0}^{L-1}\gamma_{ij}^{Y}(t)},\end{aligned} \tag{2.50}$$

when $i = 0$.

Based on the calculations above, the estimated parameter could be updated to $\eta^{(k)}$. As the algorithm is derived from EM algorithm, the expected log-likelihood is non-decreasing over iterations. Let ϵ be the acceptable difference for convergence, if $L(\eta; \eta^{(k)}) - L(\eta; \eta^{(k-1)}) \leq \epsilon$, we claim $\eta^{(k)}$ as the final estimated value for parameter η and use it for further applications; otherwise, we use $\eta^{(k)}$ to calculate the expectation of the log-likelihood and continue EM iteration to get $\eta^{(k+1)}$.

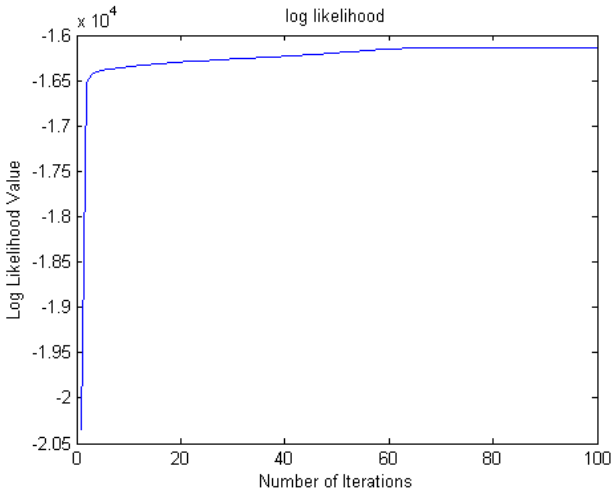


Figure 2.3: Increasing Log-Likelihood with Parameter Learning

2.5.3 Initialization

As can be seen from [27], since at every iteration the estimated parameter increases the expected log-likelihood function value, the proposed algorithm based on the EM algorithm converges to a stationary point, which can be a local optimum or a saddle point. In other words, the global maximum is not guaranteed such that convergence to which local optimum depends on the initialization of the algorithm.

For simplicity, a multi-try random initialization routine is adopted in our algorithm. Since at each time the algorithm with a random initialization converges to a local optimum, it is logical to run the algorithm multiple times with different initializations and choose the outcome that has the maximum likelihood value. Thorough discussions of optimal initialization strategies is out of the scope of this paper.

In summary, the proposed algorithm for the HIMM parameter estimation is illustrated below.

PARAMETER ESTIMATION ALGORITHM

1 Define iteration variable $k = 1$. Take the initialization value $\eta^{(0)} = \{\pi^{E(0)}, \boldsymbol{\pi}^{C(0)}, \mathbf{A}^{(0)}, \mathbf{B}^{(0)}, \mathbf{D}^{(0)}, \boldsymbol{\mu}^{(0)}, \boldsymbol{\sigma}^{2(0)}\}$ from certain distributions.

2 Use $\eta^{(k-1)}$ to calculate intermediate variables $\alpha^U, \beta^U, \gamma^U, \varepsilon^U, \alpha^Y, \beta^Y, \gamma^Y, \varepsilon^Y, \alpha^{U,Y}, \beta^{U,Y}, \gamma^{U,Y}, \varepsilon^{U,Y}$.

3 Use the intermediate variables from step 2 to calculate the current estimated parameter $\eta^{(k)}$.

4 Based on $\eta^{(k)}$, calculate the expected log-likelihood $L(\eta; \eta^{(k)})$.

5 Repeat steps 2 to 4 until $L(\eta; \eta^{(k)}) - L(\eta; \eta^{(k-1)}) \leq \epsilon$.

6 Repeat steps 1 to 5 with different initializations to find $\eta^{(k)}$ with the largest likelihood value.

7 The estimated parameter is set $\eta' = \eta^{(k)}$.

2.6 Simulation Results

In this section, numerical results are provided to validate the effectiveness of the proposed parameter learning and 2-D sensing algorithms. Our scheme considers both channel and energy observations while the reference energy detector based spectrum sensor, which is one of the well-accepted traditional PU detection methods, only uses channel observations.

For the parameter learning, we consider a training model with 5000 channel outputs and SU energy observations. The trend of the increasing log-likelihood with the proposed parameter learning algorithm is showed in Fig. 2.3. It can be seen that there is a sharp log-likelihood increase in the first five iterations, and after 60 steps the algorithm converges. As one property of the EM algorithm, the log-likelihood converges monotonically and

every random initial value guarantees the convergence. In this simulation, we tried 15 different initial values, and the estimated parameters are obtained by choosing the one with the maximum converged log-likelihood value. The initial values are randomly chosen according to the uniform distribution between 0 and 1 with proper normalizations.

In Fig. 2.4, we show the differences of probability of detection versus SNR at the SU between the traditional sensing and the proposed 2-D sensing methods, with HIMM parameters learned by the proposed learning algorithm. The SNR at the SU is defined as the SU received signal power (from PU) divided by the channel noise power. In this figure, the channel noise level is set as $\sigma_n^2 = 3$ dBw, and the channel path loss is -4 dB. The energy state takes values from $\{1, 2, 3, 4\}$, in which the insufficient energy state subset is $\mathscr{L}_0 = \{1\}$. Note that with each SNR value, we set the two schemes in comparison to have the same false alarm performance.

From Fig. 2.4, we see that the 2-D sensing method outperforms the traditional energy detector based scheme over all SNR values, since the 2-D method considers observations of both the channel state and the PU energy state, and it also utilizes the hidden Markov signaling structure between the PU and the SU. As the estimator used to estimate the hidden channel and energy states is a MAP estimator, it actually provides the optimal result that jointly estimates the hidden states of the Markov model. When the SNR is low, the observation over the PU-to-SU channel is not reliable for PU detection, which largely affects the performance of the traditional method. Since 2-D sensing also takes the energy observation into account, which is not affected by the PU-to-SU channel, the performance of 2-D sensing is much better. As SNR goes up, the channel observations become more reliable and the benefit of using the additional energy information is less, which implies that the relative gain of using 2-D sensing will decrease. However, as the 2-D sensing method does not sacrifice any information during the estimation, although the gain may become less, 2-D sensing still outperforms the traditional method in the high SNR region.

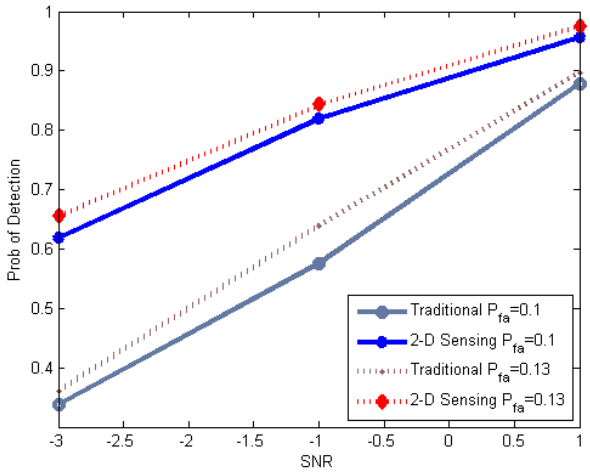


Figure 2.4: Comparison of Detection Performance

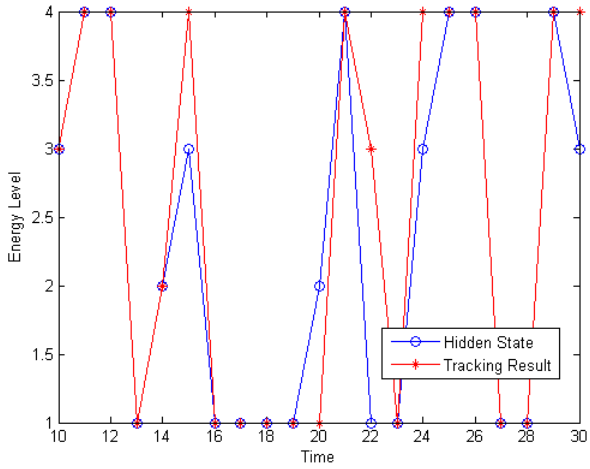


Figure 2.5: Tracking Performance of Energy States

Another advantage of using the proposed 2-D sensing method is that SU could estimate the transmit power of PU, which is proportional to E_t for broader CR applications [19] [20]. In Fig. 2.5, we show the tracking performance of the hidden PU energy state E_t, where it can be seen that the proposed method could estimate the hidden energy state well.

3. SPATIO-TEMPORAL BASED BASE STATION TRAFFIC PREDICTION

3.1 Introduction and Motivation

With the rapid development of wireless communication networks, there is an increasing demand of accurate cellular traffic prediction to improve the network performance. In fact, a lot of modern wireless system designs require the prior knowledge about base station traffic, such that with the accurate traffic information the network could enhance its performance by resource allocation schemes. For example, to reduce the energy consumption of cellular networks, the functional base station sleeping mechanism could be adopted based on the knowledge of future traffic [28].

An accurate traffic prediction model should have the ability to capture the traffic characteristics while guaranteeing the Quality of Service (QoS) for real-time applications. One potential application that requires the traffic prediction is the design of dynamic base station sleeping mechanism, which has the ability to improve the network energy efficiency by inactivating the underutilized base stations. With the prediction of base station traffic, the network can effectively shut down the underutilized base stations while maintain the service of users. Another similar application is about the resource sharing in Cloud Radio Access Network(CRAN), such that the underutilized baseband units (BBUs) could be utilized to process the information of heavy traffic base stations and this technology could improve the overall network efficiency.

However, most existing prediction methods only consider the temporal traffic correlation within each cell to learn its pattern [29, 30], neglecting the potential benefits of jointly considering spatial correlations across the entire network. Some efforts have already been made to model the spatio-temproal characteristics of wireless traffic [31, 32]. Since users continuously move within a given cellular network, the traffic flows across neighboring

base stations are correlated, such that learning over both the spatial and temporal dimensions would improve the traffic prediction performance. In fact, in real wireless network the traffic of base stations are correlated over both time and spatial, which encourage a spatio-temporal data driven model for base station traffic predictions.

Artificial nerual network could be easily adapted to learn and predict the base station traffic over the temporal dimension. However, a traditional regular neural network is hard to be generalized into the joint spatio-temporal setup. On the other hand, Recurrent Neural Network (RNN) is a special class of artificial neural network such that it is capable of keeping the internal memory while processing the sequential inputs [33]. In this work, we adopt RNN to exploit the spatial and temporal correlations among neighboring base stations.

In addition, multi-task learning is a promising way to improve the learning and predicting performance by jointly considering multiple inputs, while the different features between tasks could be utilized effectively [34]. In applying multi-task learning under our problem setup, we develop a the multi-task learning approach and analyze the corresponding experimental results.

3.2 System Model

3.2.1 Problem Formulation

Traditional traffic prediction only considers the temporal traffic correlation within each base station to estimate the possible traffic volumes in the future. However, since geometrically distributed traffics are actually correlated, using both spatial and temporal correlations across the cellular network could help improve the accuracy by feeding more information into the learning machine. We first start with a general formulation of the problem. Consider a system with N base stations and let the observation be over the past K time slots. In time t, let $\boldsymbol{x}_t = \left[x_t^1, x_t^2, \cdots, x_t^N\right]$ be the input vector with length N, which de-

notes the traffic volumes of all the base stations at time t. We know that if we merely consider the temporal correlation within each base station, the input sequence $\{\boldsymbol{x}_t\}$ would be degraded to a scalar sequence containing the current local traffic volumes. Here our objective is to find a prediction function $\hat{\boldsymbol{x}}_{t+1} = f(\boldsymbol{x}_t, \boldsymbol{x}_{t-1}, \cdots, \boldsymbol{x}_{t-K+1})$ that achieves:

$$\min_{f} \lim_{T\to\infty} \frac{1}{T} \sum_{t}^{T} L(\hat{\boldsymbol{x}}_{t+1}, \boldsymbol{x}_{t+1})$$

where the loss function $L(\cdot)$ measures the difference between the predicted and real traffic values.

Nevertheless, a general solution for minimizing the above objective function would be intractable, which encourages us to approximate the optimal predicting function with a pre-defined structure. The authors in [28, 35] adopted the artificial neural network as a framework to predict the base station traffic under different wireless network setups. However, a regular neural network could not distinguish the correlation structures across inputs such that it would not be able to differentiate the temporal and spatial correlations, when we process and predict the traffic volumes from multiple cells.

On the other hand, RNN is an extended form from the regular neural network. Instead of computing over all the inputs at the same time, the internal states of a RNN are calculated step-by-step, such that each neuron is serving as an internal memory that summarizes the past inputs. The capability of storing memory in RNN provides an efficient way to jointly explore the spatio-temporal relationships. Therefore, we adopt RNN as the basic framework to learn from the correlations over both space and time in a multicell network to accurately predict the future traffic.

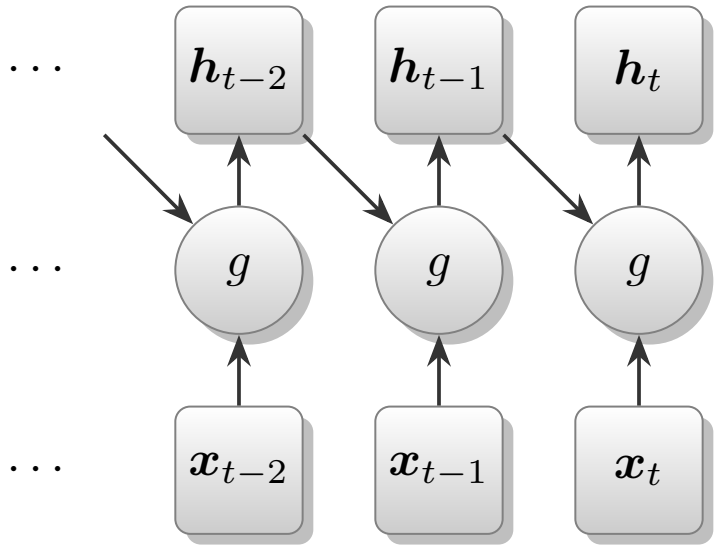


Figure 3.1: Recurrent Neural Network Structure

3.2.2 Recurrent Neural Network

An unfolded version of RNN is shown in Figure 3.1. In this model, the hidden state $\boldsymbol{h}_t$ is a function of both the memory of the previous neural network state and the current input vector $\boldsymbol{x}_t$, i.e., $\boldsymbol{h}_t = g(\boldsymbol{x}_t, \boldsymbol{h}_{t-1})$. As the previous state is taken as input, it carries the memory for learning from the internal correlations over time. The output $\boldsymbol{h}_t$ is considered to be a summary over the input sequence, which can be used to produce the predicted results of cellular traffic volumes for the next time slot.

From the Figure 3.1, we see that the function g transfers the input value into the hidden state, which in addition takes the previous hidden state into account. As we are doing the same task over different steps, the same g function is used over the entire procedures. Furthermore, we adopte the Long Short Term Memory (LSTM) structure [36] as a special case of RNN, which allows the network to also capture the long term memory while the general form of RNN given in Figure 3.1 is not capable of doing so. Specifically, LSTM has a pre-defined structure [36] for the function g that employs an additional vector c_t to

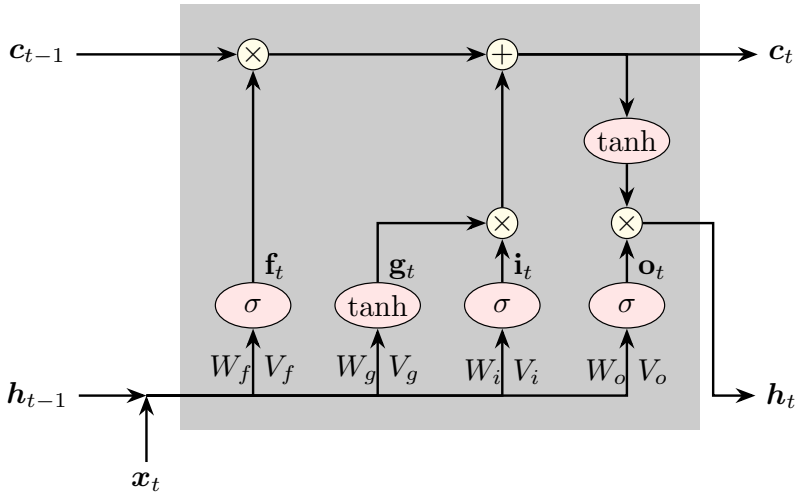


Figure 3.2: Long Short-Term Memory (LSTM)

carry the long-term memory, as shown in Figure 3.2, which is described as:

$$\mathbf{g}_t = \tanh(W_g \boldsymbol{x}_t + V_g \boldsymbol{h}_{t-1}),$$

$$\mathbf{f}_t = \sigma(W_f \boldsymbol{x}_t + V_f \boldsymbol{h}_{t-1}),$$

$$\mathbf{i}_t = \sigma(W_i \boldsymbol{x}_t + V_i \boldsymbol{h}_{t-1}),$$

$$\mathbf{o}_t = \sigma(W_o \boldsymbol{x}_t + V_o \boldsymbol{h}_{t-1}),$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t,$$

$$\boldsymbol{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t),$$

where $\odot$ denotes the element-wise product, σ denotes the sigmoid function, $\mathbf{g}_t$ is the new information generated from the current input, $\mathbf{f}_t$ is the forget gate that controls the amount of previous long-term memory $\mathbf{c}_{t-1}$ to remember, $\mathbf{i}_t$ is the input gate that controls the amount of new information acquired into the current long-term memory $\mathbf{c}_t$, $\mathbf{o}_t$ is the output gate that controls the output from $\mathbf{c}_t$, and W_* together with V_* are the input and recurrent weights for each gate and new information, respectively, with appropriate subscripts.

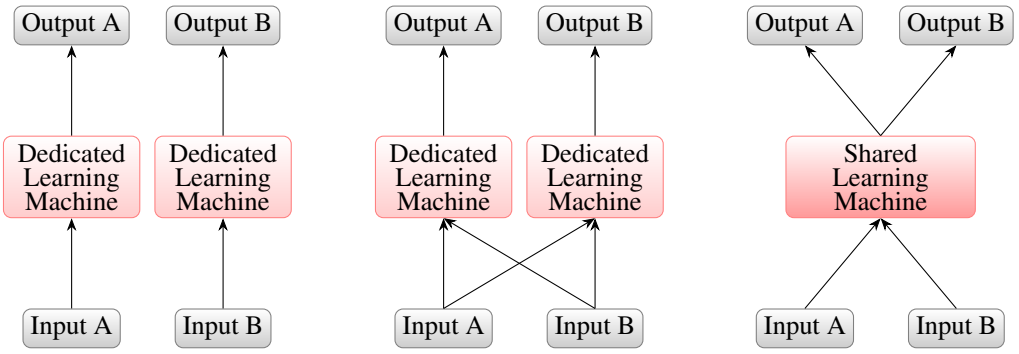
Moreover, we develop a multi-task learning approach that could deal with several tasks at the same time to leverage the mutual benefits. By considering the traffic history over multiple base stations as samples drawn from different but related distributions, the joint spatio-temporal prediction is cast as simultaneous learning over several correlated tasks. As the wireless traffic volumes are generated in different but neighboring areas, the resemblance and dissimilarity across the multiple tasks are both important components to explore. Therefore, employing such a multi-task learning framework should lead to performance gains.

3.3 Learning Architecture

In this section, we first propose several spatio-temporal learning architectures for traffic prediction. Then we describe how to integrate those spatio-temporal learning architectures into a unified multi-task learning framework. Before we get into further details, let us consider a decomposition of our predictor f as $f = \psi \circ \xi$, where the input data first go through the feature learning machine $\xi(\cdot)$, which is used to transform inputs into features. The second step involves the representation function $\psi(\cdot)$, which maps features into a prediction. In this work, we use RNN as the feature learning machine where we take the final hidden state $\boldsymbol{h}_t$ generated after processing the whole observation window as the output of $\xi(\cdot)$. The representation function $\psi(\cdot)$ is implemented as a fully connected feedforword neural network layer, which transform the final state into a prediction.

3.3.1 Basic Spatial-Temporal Learning Architectures

As RNN naturally captures the temporal information, here we mainly focus on how to explore the spatial correlation across base stations. As shown in Figure 3.3, three basic architectures with different spatial information exploration schemes are first proposed, which could be later generalized into the multi-task learning framework. For simplicity, only a two-cell scenario is presented.



(a) 1-to-1 (b) n-to-1 (c) n-to-n

Figure 3.3: Spatio-Temporal Learning Architectures

3.3.1.1 1-to-1

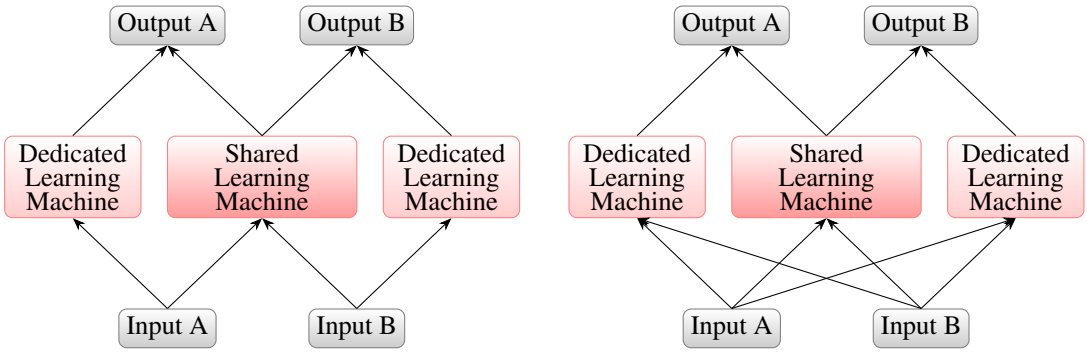
The throughput history $X_t^i = \left[x_t^i, x_{t-1}^i, x_{t-2}^i, \cdots, x_{t-K+1}^i\right]^T$ of a particular base station i is used to predict its own traffic with a local learning machine ξ^i, which is actually a pure temporal model and mainly used as a benchmark. The prediction process for each base station is given by

$$\hat{x}_{t+1}^i = \psi^i \circ \xi^i(X_t^i). \tag{3.1}$$

3.3.1.2 n-to-1

In this architecture, the prediction for each base station would still be served by its own dedicated learning machine. However, the full set of traffic volumes $\boldsymbol{X}_t = \left[X_t^1, X_t^2, \cdots, X_t^N\right]$ from all base stations is provided to each learning machine for the joint exploration of the spatio-temporal information. The prediction process for each base station can be formulated as

$$\hat{x}_{t+1}^i = \psi^i \circ \xi^i(\boldsymbol{X}_t). \tag{3.2}$$



(a) n-to-n with 1-to-1 (b) n-to-n with n-to-1

Figure 3.4: Multi-task Learning Architectures

3.3.1.3 n-to-n

Different from the previous setup, no dedicated RNN blocks are used. Instead, a shared RNN block is adopted. All the traffic volumes are provided to this shared block ξ^S to produce the shared features for the prediction of all the base station traffics at the same time. Then we have the n-to-n prediction process for each base station as

$$\hat{x}^i_{t+1} = \psi^i \circ \xi^S(\boldsymbol{X}_t). \tag{3.3}$$

3.3.2 Multi-task Learning Architecture

In the sense of simultaneous learning, the n-to-n architecture in Figure 3.3(c) could be seen as one special case of multi-task learning. However, such a n-to-n model is still a simple sequential layout of neural networks. As can be seen in (3.3), the predictions for different base stations are based on the same set of features, which implies that the differences between tasks could not be expressed effectively. To further clarify this, let us assume without the loss of generality that the loss function takes the following form $L(\hat{\boldsymbol{x}}_{t+1}, \boldsymbol{x}_{t+1}) = \frac{1}{N}\sum_{i=1}^{N} ||\hat{x}^i_{t+1} - x^i_{t+1}||$ and take the derivative of the loss function with

respect to features as:

$$\frac{\partial L}{\partial \xi_j^S} = \sum_{i=1}^{N} \frac{\partial L}{\partial \psi^i} \frac{\partial \psi^i}{\partial \xi_j^S}. \tag{3.4}$$

When using the gradient descent to minimize the loss, each feature represented in the shared part is always influenced by the other tasks. Thus the ability to represent the difference between base stations is limited under such a fully shared architecture.

To overcome this problem, we propose the multi-task learning architecture, which combines the shared and dedicated learning machines. Hence, the task-specific features could be generated and exploited to improve the performance. More specifically, the n-to-n architecture in Figure 3.3 is combined with either the 1-to-1 or the n-to-1 architecture to form the multi-task learning architectures, as illustrated in Figure 3.4. The predicting functions for the n-to-n with 1-to-1 and the n-to-n with n-to-1 models can be respectively cast as

$$\hat{x}_{t+1}^i = \psi^i \circ \left\{ \xi^S(\boldsymbol{X}_t), \xi^i(X_t^i) \right\}, \tag{3.5}$$

$$\hat{x}_{t+1}^i = \psi^i \circ \left\{ \xi^S(\boldsymbol{X}_t), \xi^i(\boldsymbol{X}_t) \right\}. \tag{3.6}$$

Under such a formulation there is one special set of features ξ^i generated for each base station i, which only serves a particular task, whose derivative in the loss function L is

$$\frac{\partial L}{\partial \xi_j^i} = \frac{\partial L}{\partial \psi^i} \frac{\partial \psi^i}{\partial \xi_j^i}. \tag{3.7}$$

These special feature sets are handled by the individual learning machines as shown in Figure 3.4. The remaining n-to-n feature set ξ^S collects the common features shared among all the base stations, which is handled by a shared learning machine.

3.4 Experiment Results

In this section, numerical experiments are conducted to demonstrate the effectiveness of the proposed spatio-temporal wireless traffic prediction framework. We first discuss the dataset and evaluation metrics. Then the results from different learning architectures are compared and analyzed.

3.4.1 Experiment Setup

Our methods are evaluated over a real cellular traffic data set collected from a big city in Asia. The data used in this work covers the hourly throughputs of 16 different base stations within a 15-day period in year 2013. The geographic locations of base stations are also provided. Such a group of 16 base stations is located in the same geographic neighborhood; thus a high level of spatial correlation is presented.

In the experiments, we use the first 70% samples to train the learning model, and the remaining 30% to validate the results. The Mean Squared Error ($\frac{1}{N}\frac{1}{T}\sum_{i=1}^{N}\sum_{t=1}^{T}(\hat{x}_{t+1}^{i} - x_{t+1}^{i})^2$) is employed to measure the accuracy of traffic prediction for N base stations over T time steps. To make the result more comparable, MSE is measured on normalized data with standard deviation equal to 1 for each base station.

3.4.2 Result of Spatio-Temporal Learning

In this section, the capabilities of our spatio-temporal models are investigated by comparing with other existing methods. The reference approaches selected include the Online Support Vector Regression (OSVR) [37], the Nonparametric Regression (NR) [38], and the Adaptive Kalman (AK) filter [39].

The performance comparisons among different models are illustrated in Figure 3.5. Since the RNN performance is influenced by random initialization, the results of our spatio-temporal models are evaluated by averaging over 100 different runs. The best re-

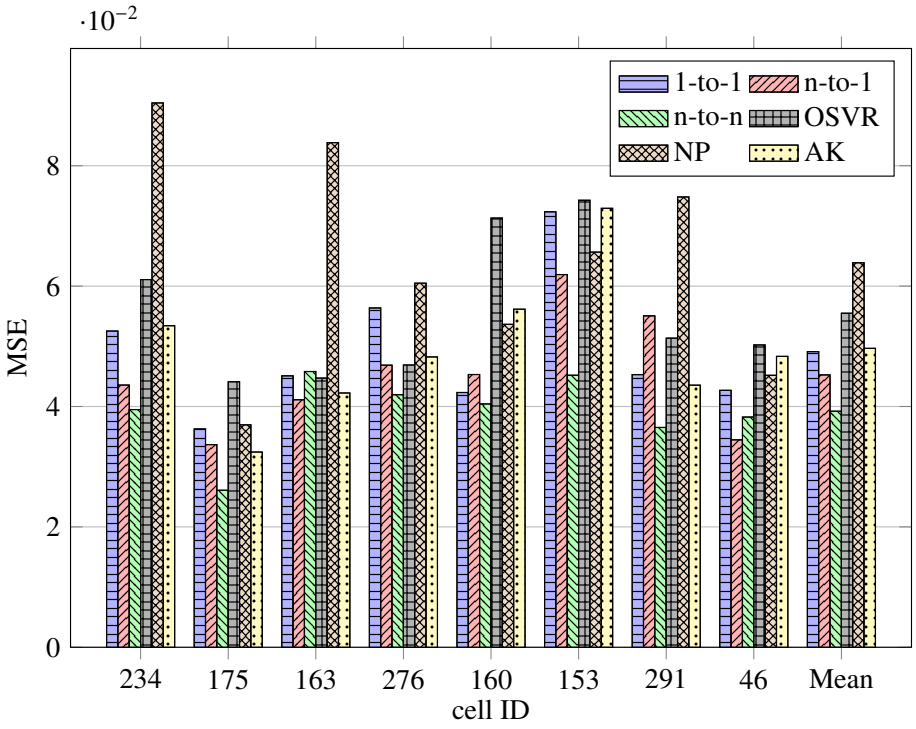


Figure 3.5: Performance for Different Cells

sults of the comparing methods are used, which are obtained by tuning hyper-parameters with the Bayesian optimization algorithm [40]. We see that the RNN based models outperform all comparing approaches in most cases. The AK model is a linear model that is not able to explore the non-linear correlations. Although the kernel trick used in OSVR is very powerful, its representation power is still limited. Meanwhile, the NP model is trying to mimic the historic data and failed to actually capture the characteristics.

Among those proposed RNN based models in Figure 3.3, the pure temporal model (the first one in Figure 3.3) is very often the worst. Although at some base stations it outperforms the n-to-1 model, the n-to-1 model is still the better one in most cases.

Another important observation is that the n-to-n model outperforms the n-to-1 model, i.e., instead of training a model for each base station, predicting those base stations all together could provide us even better results. This observation may be somehow counter-

intuitive that a multi-objective optimization solution can outperform the dedicated solutions. However, we should think over this multi-objective solution from a different angle. By predicting multiple base station traffics at the same time, the mapping between multiple inputs and outputs can provide extra information to the model. In particular, by introducing additional optimization objectives we enforces the network to extract more general features from the training data and prevent overfitting issues.

Some experiments are also designed to illustrate the impact of several experiment parameters. The numeric results under different settings can help us better understand the spatio-temporal information embedded in our data set.

3.4.2.1 Size of Recurrent Network

Figure 3.6(a) shows the result under different numbers of RNN neurons. The best performance of OSVR and NP models is also drawn as a reference. When RNN does not have enough neurons, the information representation ability is limited, especially for the n-to-n case, where overwhelmed information causes under-fitting. By increasing the number of neurons, more features are extracted. However, the improvement stops after the learning machine size of 150 is reached. This experiment further shows that the n-to-n model explores extra information, which are extracted by the increased number of neurons.

3.4.2.2 Size of Spatial Input

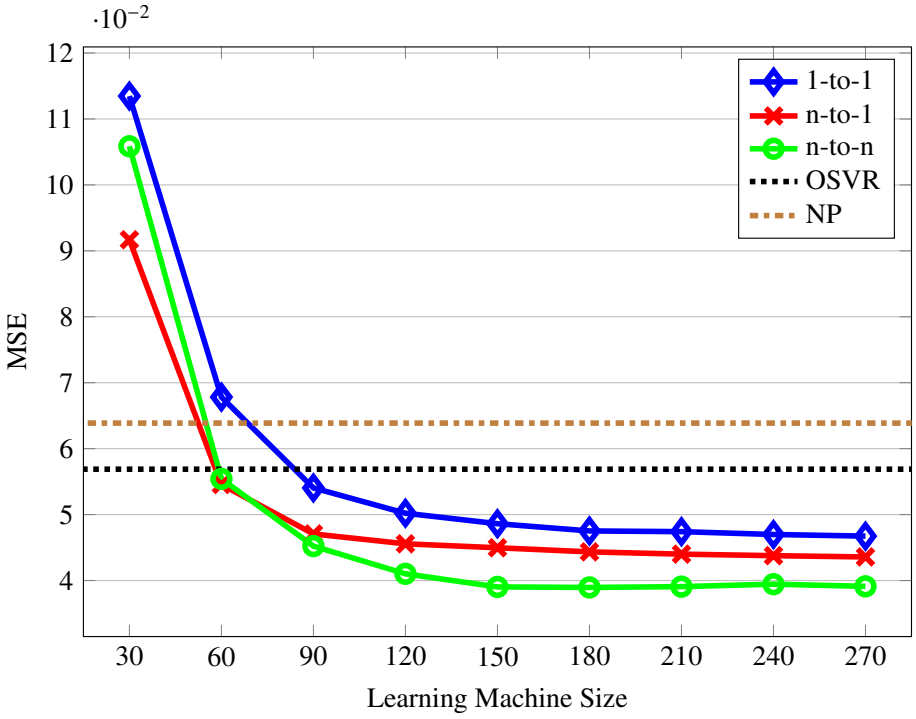
In this experiment, the most correlated n "neighbors" of the target base station are selected to provide the spatio-temporal information. Although the n-to-n model is used, only the predicted result of the target base station is evaluated. In this way, the benefit of spatial information to a particular base station is presented. As shown in Figure 3.6(b), the overall performance is improved by the increased size of spatial inputs, but the improvement almost stops when the input size is greater than 8. The result is intuitive since each

base station only has a few highly correlated “neighbors”, such that increasing the size of spatial inputs does not always help with the prediction performance for a particular base station.

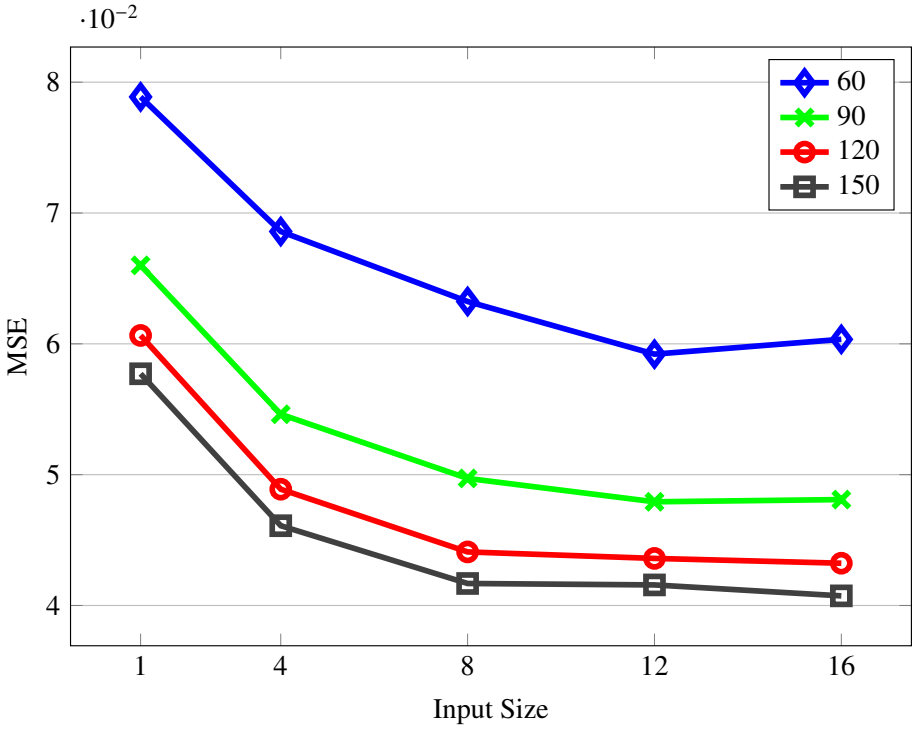
3.4.3 Result of Multi-task Learning

To validate the capability of multi-task learning, we chose the n-to-n model as the shared learning machine in the multi-task learning framework, where the size of the shared learning machines is set as 150. The best result achieved by the n-to-n model alone is also shown as a reference. As illustrated in Figure 3.7, improved performance is achieved by both of two multi-task learning frameworks.

Furthermore, the n-to-n with 1-to-1 framework performs slightly better than the n-to-n with n-to-1 one. Since the shared learning machine has explored the spatial correlation, providing spatial information to the dedicated learning machine would not further improve the performance; it may make the training more difficult to converge. In addition, we see that the performance gets worse with the size of the dedicated learning machine increase. This degenerated result is caused by the large dedicated learning machine size that dominates the behavior of the overall multi-task learning and leads to overfitting issues.



(a) Effect of different learning machine size



(b) Effect of different input size

Figure 3.6: Performance Under Different Experiment Parameters

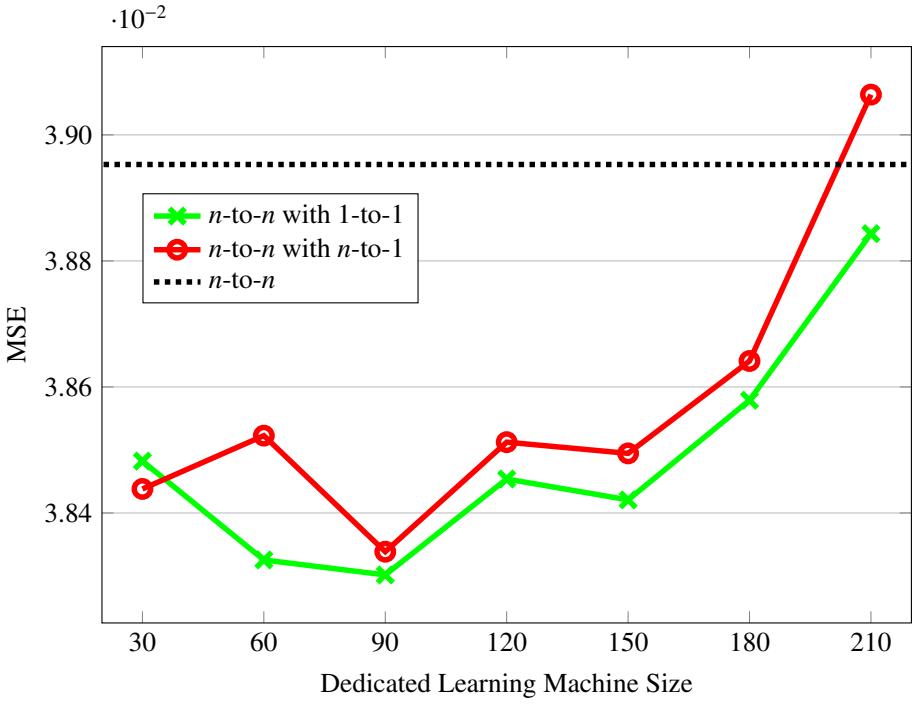


Figure 3.7: Performance of Multi-Task Learning

4. SUMMARY AND CONCLUSIONS

In this dissertation, we have applied several statistical learning and data mining techniques to conduct intelligent designs in wireless communication systems. Specifically, in the first problem, we applied a HIMM model to represent the interactions between energy harvesting based primary and cognitive radios, and provided an EM based algorithm to estimate the parameters in this HIMM structure. Then we proposed a novel 2-D sensing scheme, which jointly considers the observations from the spectrum and power dimensions. The proposed scheme could sense the spectrum and estimate the energy level for PU transmission simultaneously. We showed that the proposed 2-D sensing method outperforms the traditional spectrum sensing method, since it utilizes the facts that the PU power usage and channel usage are interdependent events, and the PU/SU energy harvesting processes are spatially correlated.

In the second work we presented multiple RNN based learning models along with an unified multi-task learning frameworks to explore spatio-temporal correlations among base stations, in the goal of improving the traffic prediction performance. Base on real data, we provided detailed evaluations on different learning models and demonstrated that the spatial correlation among base stations could provide valuable information to improve the prediction accuracy. In addition, we showed that the commonalities and differences across different base stations could be better exploited by the proposed multi-task learning frameworks.

REFERENCES

[1] Z. Quan, S. Cui, and A. H. Sayed, "Optimal linear cooperation for spectrum sensing in cognitive radio networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 2, pp. 28–40, Feb. 2008.

[2] Z. Ye, J. Grosspietsch, and G. Memik, "Spectrum sensing using cyclostationary spectrum density for cognitive radios," in *IEEE Workshop on Signal Processing Systems*, (Shanghai, China), pp. 1–6, Oct. 2007.

[3] Z. Quan, S. Cui, A. H. Sayed, and H. V. Poor, "Wideband spectrum sensing in cognitive radio networks," in *IEEE International Conference on Communications (ICC)*, (Beijing, China), pp. 901–906, May 2008.

[4] Z. Tian and G. B. Giannakis, "Compressed sensing for wideband cognitive radios," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 4, (Honolulu, USA), pp. IV–1357–IV–1360, Apr. 2007.

[5] C. Ghosh, C. Cordeiro, D. Agrawal, and M. Rao, "Markov chain existence and hidden markov models in spectrum sensing," in *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, (Galveston, USA), pp. 1–6, Mar. 2009.

[6] Z. Sun, G. J. Bradford, and J. N. Laneman, "Sequence detection algorithms for PHY-layer sensing in dynamic spectrum access networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, pp. 97–109, Feb. 2011.

[7] N. Thao, B. L. Mark, and Y. Ephraim, "Spectrum sensing using a hidden bivariate markov model," *IEEE Transactions on Wireless Communications*, vol. 12, pp. 4582–4591, Sep. 2013.

[8] X. He, H. Dai, and P. Ning, "HMM-based malicious user detection for robust collaborative spectrum sensing," *IEEE Journal on Selected Areas in Communications*, vol. 31, pp. 2196–2208, Nov. 2013.

[9] W. Chung, S. Park, S. Lim, and D. Hong, "Spectrum sensing optimization for energy-harvesting cognitive radio systems," *IEEE Transactions on Wireless Communications*, vol. 13, pp. 2601–2613, May 2014.

[10] S. Yin, Z. Qu, and S. Li, "Achievable throughput optimization in energy harvesting cognitive radio systems," *IEEE Journal on Selected Areas in Communications*, vol. 33, pp. 407–422, March 2015.

[11] S. Reddy and C. R. Murthy, "Dual-stage power management algorithms for energy harvesting sensors," *IEEE Transactions on Wireless Communications*, vol. 11, pp. 1434–1445, Apr. 2012.

[12] A. E. Susu, A. Acquaviva, D. Atienza, and G. De Micheli, "Stochastic modeling and analysis for environmentally powered wireless sensor nodes," in *International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks and Workshops (WiOPT)*, (Berlin Germany), pp. 125–134, Apr. 2008.

[13] Q. Bai, R. A. Amjad, and J. A. Nossek, "Average throughput maximization for energy harvesting transmitters with causal energy arrival information," in *IEEE Wireless Communications and Networking Conference (WCNC)*, (Shanghai, China), pp. 4232–4237, Apr. 2013.

[14] S. Luo, R. Zhang, and T. J. Lim, "Optimal save-then-transmit protocol for energy harvesting wireless transmitters," *IEEE Transactions on Wireless Communications*, vol. 12, pp. 1196–1207, Mar. 2013.

[15] M. He, L. Yang, J. Zhang, and V. Vittal, "Spatio-temporal analysis for smart grids with wind generation integration," in *International Conference on Computing, Networking and Communications (ICNC)*, (San Diego, USA), pp. 1107–1111, Jan. 2013.

[16] S. Murugesan, J. Zhang, and V. Vittal, "Finite state markov chain model for wind generation forecast: A data-driven spatiotemporal approach," in *IEEE PES Innovative Smart Grid Technologies (ISGT)*, (Washington, DC), pp. 1–8, Jan. 2012.

[17] R. G. Cid-Fuentes, A. Cabellos, and E. Alarcon, "Energy harvesting enabled wireless sensor networks: Energy model and battery dimensioning," in *Proceedings of the 7th International Conference on Body Area Networks*, (Oslo, Norway), pp. 131–134, Sep. 2012.

[18] M. He, L. Yang, J. Zhang, and V. Vittal, "A spatio-temporal analysis approach for short-term forecast of wind farm generation," *IEEE Transactions on Power Systems*, vol. 29, pp. 1611–1622, Jul. 2014.

[19] T. Do and B. Mark, "Joint spatial-temporal spectrum sensing for cognitive radio networks," in *Conference on Information Sciences and Systems (CISS)*, (Baltimore, USA), pp. 124–129, Mar. 2009.

[20] Z. Chen, F. Gao, X. Zhang, J. Li, and M. Lei, "Sensing and power allocation for cognitive radio with multiple primary transmit powers," *IEEE Wireless Communications Letters*, vol. 2, pp. 319–322, Jun. 2013.

[21] O. Ozel and S. Ulukus, "AWGN channel under time-varying amplitude constraints with causal information at the transmitter," in *Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, (Pacific Grove, USA), pp. 373–377, Nov. 2011.

[22] Y. Bengio and P. Frasconi, "Input-output HMMs for sequence processing," *IEEE Transactions on Neural Networks*, vol. 7, pp. 1231–1249, Sep. 1996.

[23] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1993.

[24] J. Bilmes, "A gentle tutorial of the EM algorithm and its application to parameter estimation for gaussian mixture and hidden markov models," *Technical Report 97-021*, Apr. 1998.

[25] C. F. J. Wu, "On the convergence properties of the EM algorithm," *The Annals of Statistics*, vol. 11, pp. 95–103, Mar. 1983.

[26] S. Boyd, *Convex optimization*. Cambridge, UK: Cambridge University Press, 2004.

[27] T. K. Moon, "The expectation-maximization algorithm," *IEEE Signal Processing Magazine*, vol. 13, pp. 47–60, Nov. 1996.

[28] J. Hu, W. Heng, G. Zhang, and C. Meng, "Base station sleeping mechanism based on traffic prediction in heterogeneous networks," in *Proceedings of the 2015 International Telecommunication Networks and Applications Conference*, (Sydney, Australia), pp. 83–87, Nov. 2015.

[29] T. Lagkas, *Wireless Network Traffic and Quality of Service Support: Trends and Standards*. Information Science Reference, 2010.

[30] I. Loumiotis, E. Adamopoulou, K. Demestichas, T. Stamatiadi, and M. Theologou, "On the predictability of next generation mobile network traffic using artificial neural networks," *International Journal of Communication Systems*, vol. 28, pp. 1484–1492, May 2015.

[31] X. Chen, Y. Jin, S. Qiang, W. Hu, and K. Jiang, "Analyzing and modeling spatio-temporal dependence of cellular traffic at city scale," in *Proceedings of the 2015 IEEE International Conference on Communications*, (London, United Kingdom.), pp. 3585–3591, June 2015.

[32] H. Wang, J. Ding, Y. Li, P. Hui, J. Yuan, and D. Jin, "Characterizing the spatio-temporal inhomogeneity of mobile traffic in large-scale cellular data networks," in *Proceedings of the 7th International Workshop on Hot Topics in Planet-scale mObile Computing and Online Social neTworking*, (Hangzhou, China), pp. 19–24, June 2015.

[33] R. Jozefowicz, W. Zaremba, and I. Sutskever, "An empirical exploration of recurrent network architectures," in *Proceedings of the 32nd International Conference on International Conference on Machine Learning*, (Lille, France), pp. 2342–2350, July 2015.

[34] R. Caruana, "Multitask Learning," *Machine Learning*, vol. 28, pp. 41–75, July 1997.

[35] I. Loumiotis, E. Adamopoulou, K. Demestichas, P. Kosmides, and M. Theologou, "Artificial neural networks for traffic prediction in 4G networks," in *Proceedings of the 8th International Wireless Internet Conference*, (Lisbon, Portugal), pp. 141–146, Nov. 2014.

[36] Y. Gal and Z. Ghahramani, "A theoretically grounded application of dropout in recurrent neural networks," in *Proceedings of the 29th Advances in Neural Information Processing Systems*, (Barcelona, Spain), pp. 1019–1027, Dec. 2016.

[37] M. Castro-Neto, Y.-S. Jeong, M.-K. Jeong, and L. D. Han, "Online-SVR for short-term traffic flow prediction under typical and atypical traffic conditions," *Expert Systems with Applications*, vol. 36, pp. 6164–6173, Apr. 2009.

[38] S. Clark, "Traffic prediction using multivariate nonparametric regression," *Journal of Transportation Engineering*, vol. 129, pp. 161–168, Feb. 2003.

[39] J. Guo, W. Huang, and B. M. Williams, "Adaptive kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification," *Transporta-*

tion Research Part C: Emerging Technologies, vol. 43, pp. 50–64, June 2014.

[40] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," in *Proceedings of the 25th Advances in Neural Information Processing Systems*, (Lake Tahoe, NV, USA), pp. 2951–2959, Dec. 2012.