Chapter 2
Channel Usage Patterns and Their Impact on the Effectiveness of Machine Learning for Dynamic Channel Selection

Irene Macaluso, Hamed Ahmadi, Luiz A. DaSilva and Linda Doyle

Abstract The diverse behavior of different primary users (PU) in various spectrum bands impacts a cognitive radio's ability to exploit spectrum holes. This chapter summarizes the results of our previous studies on the impact of the complexity of primary users' behavior on the performance of learning algorithms applied to dynamic channel selection. In particular, we characterize the observable spectrum utilization with respect to the duty cycle of the channels and to the complexity of the primary user's activity. We use the term complexity to refer to the unpredictability associated with the primary user's wireless resource usage, which we quantitatively characterize using Lempel-Ziv complexity. We evaluate the effectiveness of two learning-based dynamic channel selection algorithms by testing them with real spectrum occupancy data collected in the GSM, ISM, and DECT bands. Our results show that learning performance is highly correlated with the level of PU activity, estimated by the duty cycle, and the amount of structure in the use of spectrum, estimated by the Lempel-Ziv complexity.

2.1 Introduction

Opportunistic spectrum access may rely on a combination of geolocation databases and spectrum sensing to detect spectrum holes. In their search for spectrum holes, secondary users (SUs) can use learning methods to predict the next channel state from

I. Macaluso (✉) · H. Ahmadi · L. A. DaSilva · L. Doyle
CTVR Telecommunications Research Center, Trinity College, Dublin, Ireland
e-mail: irene.macaluso@gmail.com

H. Ahmadi
e-mail: ahmadih@tcd.ie

L. A. DaSilva
Department of Electrical and Computer Engineering, Virginia Tech, Arlington, VA, USA
e-mail: dasilval@tcd.ie

L. Doyle
e-mail: linda.doyle@tcd.ie

M.-G. Di Benedetto and F. Bader (eds.), *Cognitive Communication and Cooperative HetNet Coexistence*, Signals and Communication Technology,
DOI: 10.1007/978-3-319-01402-9_2,

their past observations on each channel. However, because of the diverse behavior of primary users in different spectrum bands, spectrum holes exhibit different characteristics, which in turn affect the performance of a learning algorithm [1, 2].

This chapter summarizes the results of our previous studies on the impact of the complexity of PUs' behavior on the performance of learning algorithms applied to dynamic channel selection (DCS) [2, 3]. In particular, we consider two learning approaches. In [2], we presented a reinforcement learning-based method for spectrum opportunity prediction; in [3], we presented a Markov process-based learning. We analyzed the performance of the two approaches with respect to the duty cycle (DC) of the channels and to the complexity of the PU's activity, relying on actual spectrum measurement data.

In both cases we characterized the PU activity by using a measure of complexity proposed by Lempel and Ziv [4]. Our results showed that the amount of structure in the PU activity, estimated by the Lempel-Ziv (LZ) complexity, has a significant impact on the performance of the learning-based DCS approaches. In particular, our studies showed that the LZ complexity of the PU's behavior can account for up to a 30 and 20 % difference in the probability of success of reinforcement learning (RL) and Markov process-based learning respectively.

We begin with a short discussion on the literature of learning algorithms applied to DCS in Sect. 2.2. Section 2.3 details our use of Lempel-Ziv complexity to quantify the amount of structure in the usage of the bands by the PU. Section 2.4 describes the two learning-based dynamic channel selection approaches. Section 2.5 presents the relationship between the effectiveness of learning-based DCS and the amount of structure in the usage of the bands by the PU, relying on spectrum measurements conducted at RWTH Aachen and by us at Trinity College Dublin to determine, for any given time slot and set of channels, whether there is PU activity. We summarize our conclusions in Sect. 2.6.

2.2 Existing Works

In the literature, several learning algorithms have been proposed and applied to predict the channel state. All these approaches need to learn the channel occupancy model by observing the PU activity for a certain number of time slots. This phase is called training or learning period. The learning algorithms are able to make predictions on the channel state after they are trained. Learning algorithms with higher accuracy, lower complexity and shorter training time are preferred. In this section we briefly review some of the existing works.

In [5], Clancy claims that a hidden Markov model (HMM) can be a suitable method to model the channel occupancy as a function of time. In [6], the authors model the channel state occupancy of a PU on each channel as a Poisson distribution and use an HMM to predict the availability of a channel. The HMM is trained with the Baum-Welsh algorithm (BWA) [7], predicting the presence of PUs to avoid transmission collisions. An SU will occupy an idle channel until a PU becomes active in that channel, then it will switch to another predicted idle channel. Simulation results show that the probability of collision is reduced compared to a random selection of channels to be sensed by the SU. An artificial neural network is proposed in [8] to predict the channel state for the next time slot. In [9], the authors evaluate the performance of their proposed neural network when the statistics of the channel are changing. The authors of [10] formulate this problem of efficiently using the spectrum as a Markov decision process and propose a solution strategy based on reinforcement learning techniques. In [11], we propose a modified, less complex HMM. We compare the prediction accuracy of our proposed method with that of conventional HMM and show that it achieves the same prediction accuracy with much less computational complexity.

The aforementioned works test their proposed algorithms on data which are generated based on some assumptions on the probability distribution of the PU activity (synthetic data), and none of them applies their algorithm on data that are collected from actual sensing (real data).

There are some recent works which use real data in their studies. In [12], authors conduct spectrum measurements in Guangzhou city, and then approximate the prediction error with the beta distribution. Kone et al. propose frequency bundling in [13], where secondary devices build reliable channels by combining multiple unreliable frequencies into virtual frequency bundles. Their experiments on real data show that bundling random channels together can provide sustained periods of reliable transmission.

The works above only study the performance of their proposed algorithms and compare them with other learning algorithms. On the other hand, few works [2, 3] investigated the predictability of the behavior of the PUs and its impact on the learning performance, which is the main focus of the rest of this chapter.

2.3 Complexity of Primary User Activity

The term complexity is used with different meanings in the literature [14]. In some domains, e.g. dynamic systems and statistical mechanics, both completely ordered or disordered sequences are associated with low complexity [15]. In other words, complexity does not increase monotonically with disorder. In this chapter our goal is to investigate the effectiveness of learning algorithms in exploiting the regularities of channel utilization; hence we use the term complexity to refer to the unpredictability, or uncertainty, associated with the PU's wireless resource usage. Accordingly, we adopt a measure of complexity that associates high values of complexity to completely disordered, i.e. unpredictable, spectrum occupancy sequences.

We quantitatively characterize the structure of a spectrum occupancy sequence by making use of a measure of complexity proposed by Lempel and Ziv [4]. In particular, we adopt the normalized Lempel-Ziv complexity, which measures the rate of production of new patterns in a sequence. The complexity coefficient c is computed by scanning the sequence and incrementing c every time a new substring of consecutive symbols is found. Then c is normalized via the asymptotic limit $n/\log_2(n)$, where n is the length of the sequence [16]. Lempel-Ziv complexity is a property of individual sequences and it can be computed without making any assumptions about the underlying process that generated the data. This feature is of the utmost importance when one is dealing with real data (in our case, sensed channel status in a variety of frequency bands). Furthermore, LZ complexity is strongly related to the source entropy. In fact, if the source is ergodic, the normalized LZ complexity has been proven [17] to be equal to the source entropy almost surely.

It is interesting to note that the LZ complexity is closely related to the concept of Kolmogorov complexity [18], which measures the complexity of a binary string s as the bit length of the shortest program that produces s and halts afterwards. Whereas Kolmogorov complexity refers to the shortest program among all the possible classes of programs, LZ complexity makes use of one class of programs that can only perform copy and paste operations [16]. Although Kolmogorov complexity is known to be algorithmically uncomputable, it should be noted that in the case of ergodic sources Brudno's theorem states that the entropy rate of the source is equal to the Kolmogorov complexity per symbol of almost all emitted strings ([19], as discussed in [20]). Lempel-Ziv complexity and Kolmogorov complexity are deterministic complexity measures: by looking for the shortest description that allows to exactly reproduce the data, they inevitably include the noise in such description.

2.4 Learning and DCS

We consider the problem of an SU searching for a channel to occupy opportunistically while the PU is inactive in that channel. We do not assume the SU to have any a priori knowledge of the pattern of activity of the PU or the long-term probability that each channel is occupied. We explore two alternative learning strategies to decide which channel to sense prior to each transmission slot: (i) the SU applies reinforcement learning; (ii) the SU applies Markov-based learning. Our goal is to study the effect of both the levels of PU activity and the complexity of the PU behavior as defined in Sect. 2.3 on the effectiveness of learning.

Our model considers a single SU that can use one of N equal-bandwidth frequency channels opportunistically. Time is slotted and alternates between a sensing phase and a transmission phase. The SU is allowed to transmit in the time slot if the selected channel in the sensing phase is still free. The SU's choice of which channel to attempt transmission in will therefore affect its performance: the more successful the SU is in predicting which channel is the least likely to contain PU activity in the next time slot, the greater its likelihood to opportunistically utillize the channels.

In our study, we rely on spectrum measurements conducted at RWTH Aachen and by us at Trinity College Dublin to determine, for any given time slot and set of channels, whether there is PU activity. For each band that we investigate, we evaluate the adaptation techniques described in Sects. 2.4.1 and 2.4.2 for different combinations of N channels to be explored by the SU.

2.4.1 Reinforcement Learning

A secondary user selects one among N channels according to the policy determined by an RL algorithm. Under the assumption of full observability of the state of the channels, Q-learning [21] is the most natural candidate. Moreover, as Q-learning does not require a model of the agent's environment, it is suitable to deal with real spectrum occupancy data.

The goal of Q-learning is to find an optimal policy, i.e. the sequence of actions that maximizes the expected sum of discounted rewards. The idea is to reward an SU if it selects a free channel, while also including a cost of switching channels to discourage too frequent channel changes. The SU state at time t is given by $\mathbf{s_t} = [X_{1,t}, \ldots, X_{N,t}, c_t]$, where $X_{i,t} \in \{0, 1\}$ indicates whether the ith channel is free (0) or occupied (1) and $c_t \in \{1, \ldots, N\}$ is the index of the channel the SU is accessing at time t. At time t the SU performs an action $a_t \in \{1, \ldots, N\}$, i.e. it selects a channel c_{t+1}. At time $t + 1$ it receives a reward $r_{(t+1)}(\mathbf{s_t}, a_t)$:

$$r_{(t+1)}(\mathbf{s_t}, a_t) = (1 - X_{a_t,t+1}) - e(\mathbb{1}_{a_t,c_t}) \tag{2.1}$$

where $\mathbb{1}_{a_t,c_t}$ is 0 if $a_t = c_t$ and 1 otherwise, and $e \in [0, 0.5]$.

Based on the received reward, the SU updates the Q-values according to [21]:

$$Q(\mathbf{s_t}, a_t) := Q(\mathbf{s_t}, a_t) + \alpha r_{(t+1)} + \alpha(\gamma \max_{a_{t+1}} Q(\mathbf{s_{t+1}}, a_{t+1}) - Q(\mathbf{s_t}, a_t)) \tag{2.2}$$

where $0 \leq \gamma < 1$ is the discount factor and α is the learning rate. If $\gamma > 0$, the agent takes into account not only the immediate reward but also the delayed reward when it chooses which action to take.

In a stationary environment Q-learning is proven to converge to the optimal policy if $\alpha \to 0$ and all the state-action pairs are visited an infinite number of times. During the learning stage, an exploration strategy is required to allow the agent to visit all the state-action pairs. A randomized strategy is commonly adopted: the agent selects a random action with a probability ε and the best estimated action with probability $1 - \varepsilon$. At the beginning the algorithm starts with a large value of ε, which decreases as the Q-learning converges.

As the stationary condition is generally not satisfied for real spectrum occupancy data, we fix the learning factor to 0.1 to allow the agent to adapt to the changes in the environment. Moreover, we set the ε-value to 0.01 to allow the agent to perform exploratory actions from time to time in order to discover changes in the environment.

2.4.2 Markov-Based learning

As mentioned earlier, the SU needs to select a channel (from the set of channels that it can access) for transmission which is the least likely to be occupied by the

PU. Other than reinforcement learning, we can use Markov process-based learning algorithms to learn the channels' occupancy model and predict the availability of the channels for the next time slot. Here, we study a low complexity Markov process-based learning algorithm which can accurately predict the channel status of the next time slot.

Our Markov process-based learning algorithm has K states and M possible observations, where $\mathscr{S}$ and $\mathscr{O}$ represent the sets of possible states and observations, respectively.

We denote sequences of states by $\mathbf{x}$, and we use $\mathbf{y}$ to indicate the sequence of observations. Each element of $\mathbf{x}$, denoted by $x(t) \in \mathscr{S}$, $\forall t$, is the state at time t; each element of $\mathbf{y}$, denoted by $y(t) \in \mathscr{O}$, $\forall t$, is the observation at time t.

The transition probabilities between the states are stored in a $K \times K$ matrix ($\mathbf{A}$). The distribution of the observation outcomes at each state is described by the respective column vector of the $K \times M$ emission matrix $\mathbf{B}$. We represent this Markov process by $\lambda = \{\mathbf{A}, \mathbf{B}, \pi\}$, where π is the initial state distribution.

The algorithm is trained off-line over a training sequence. In this Markov process-based learning model, the number of states (i.e. K) grows dynamically as learning proceeds. Here, we have two possible observations ($M = 2$). We observe a zero when we sense a free channel, and we observe a one by sensing a busy channel.

The transitions between states depend on the length of the string of consecutive zeros or ones observed. This means that in our system each state represents a number of observed consecutive zeros or ones. Positive states represent the number of observed consecutive ones, and non-positive (negative and zero) states represent the number of observed consecutive zeros.

During the training phase, we create the Markov chain using the training data set and based on the number of consecutive zeros and ones. Then, it is possible to count the number of times each particular transition or output observation is applied in a set of training data. As proven in [7], counting functions for the output observations provide maximum likelihood estimates for the desired model parameters. Suppose that the maximum number of negative and positive states in the Markov chain, after the training, are q and p, respectively. The set of states is $S = \{s_{q-}, \ldots, s_0, \ldots, s_p\}$ which has the cardinality of $N = q + p + 1$. The elements of transition and emission matrices will be computed by:

$$a_{i,j} = \frac{f_{i,j}(\mathbf{x})}{\sum_{k=-q}^{p} f_{i,k}(\mathbf{x})} \; \forall i, j \in \{-q, \ldots, p\}, \tag{2.3}$$

$$b_{i,j} = \frac{g_{i,j}(\mathbf{x}, \mathbf{y})}{\sum_{m=1}^{M} g_{i,m}(\mathbf{x}, \mathbf{y})} \; \forall i \in \{-q, \ldots, p\}, j \in \{1, \ldots, M\}, \tag{2.4}$$

where $a_{i,j}$ is the transition probability from s_i to s_j, and $b_{i,j}$ is the probability of o_j at s_i. The counting functions $f_{i,j}$ and $g_{i,j}$ simply count the number of transitions from state s_i to state s_j and the number of observations o_j at state s_i, respectively.

During the training process, the transition and emission probabilities over the observations can be easily calculated by (2.3) and (2.4). As a result, we can predict the observation by:

$$\check{y}(t) = \begin{cases} 0 & p(x(t), 0|\lambda) \geq p(x(t), 1|\lambda) \\ 1 & otherwise \end{cases}, \tag{2.5}$$

where $\check{y}(t)$ indicates the predicted observation for time t.

In case of an inaccurate prediction, the system will notice the prediction error after observing $y(t)$. Since $x(t+1)$ only depends on the observation outcome rather than on the predicted result, the system will move to the correct state and errors will not propagate. Retraining the system is only needed when the statistics of the behavior of PUs on the channel are changing. To account for this, the system can be retrained after a certain number of time slots or whenever the prediction accuracy drops below a certain threshold.

2.5 Results

In this section we analyze the performance of the two learning solutions presented in Sect. 2.4. The results presented in this section refer to sequences of spectrum occupancy over 12 h (from 11:00 to 23:00) and duty cycle $DC \in [0.3, 0.8]$. We examined a number of frequency bands: the 2.4 GHz ISM band, the DECT band, and the GSM900 and GSM1800 bands. For each band, we considered all the possible combinations of $N = 3$ and $N = 4$ channels to evaluate the RL approach and the Markov-based learning approach respectively. Each combination corresponds to one instance of the DCS problem. One of the fundamental issues common to all RL approaches is the convergence time when the dimension of the state-action space is large. For this reason, we consider combinations of $N = 3$ channels in the case of RL. The convergence time of the Markov process-based learning algorithm increases linearly with N, thus allowing us to consider combinations of 4 channels. However, the number of channels that an SU can observe (N) strongly depends on the hardware and sensing capabilities of the SU.

Our empirical findings show that the performance benefits of the two learning algorithms are highly correlated with the level of PU activity observed and the amount of structure in these observations, estimated by the LZ complexity.

2.5.1 Reinforcement Learning

Each combination of channels corresponds to one instance of the RL problem. For each instance, we run 10^3 independent simulations. For each simulation, first we compute the optimal policy using the Q-learning algorithm and considering only

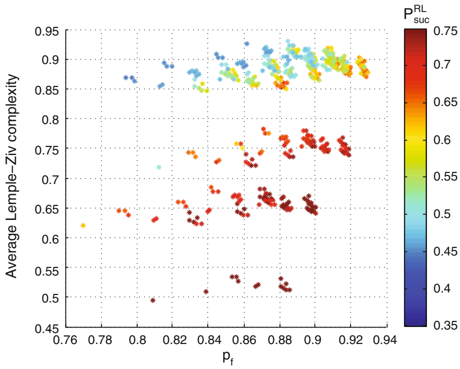

Fig. 2.1 ISM band (outdoor), RWTH Aachen dataset. Probability of success of Q-learning as a function of the average LZ complexity and the probability of at least one free channel existing. Each point represents a particular instance of Q-learning applied to $N = 3$ channels. The total number of possible combinations which we analyzed is $\binom{15}{3} = 455$, where 15 is the number of channels with $DC \in [0.3, 0.8]$

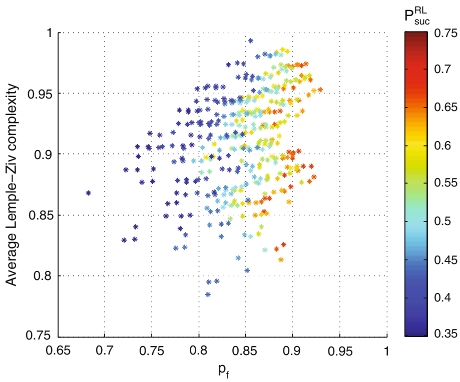

Fig. 2.2 GSM1800 band (outdoor), RWTH Aachen dataset. Probability of success of Q-learning as a function of the average LZ complexity and the probability of at least one free channel existing. Each point represents a particular instance of Q-learning applied to $N = 3$ channels. The total number of possible combinations which we analyzed is $\binom{15}{3} = 455$, where 15 is the number of channels with $DC \in [0.3, 0.8]$

the first hour of the sequences of spectrum occupancy. Then, the resulting policy is evaluated over the remaining 11 h. For each simulation the probability of success is computed according to the number of times that a free channel was selected over the length of the spectrum occupancy sequences. The probability of success of each RL instance is the average over the 10^3 simulations.

Figures 2.1, 2.2 show the probability of success of Q-learning for the ISM band and the GSM1800 band respectively, as a function of the average LZ complexity and the probability of at least one free channel existing $p_f = 1 - \prod_{i=1}^{N} \delta_{1,i}$, where

$\delta_{1,i}$ is the duty cycle of the i-th channel. The LZ complexity of each channel is computed using the algorithm described in [16], while the average value is used as an approximation of the complexity of each combination of channels.

As expected, the probability of success increases with p_f. However, it can be observed that the performance of RL is also strongly dependent on the complexity of the PU behavior. For example, for the ISM band (see Fig. 2.1), the values for the average LZ complexity span a considerable range and we can easily observe that, when p_f remains constant, the RL performance decreases when LZ complexity increases.

Both the average LZ complexity and p_f do not cover the same range for each band. As it can be observed in Figs. 2.1 and 2.2 the DC values, and therefore the p_f values, span almost the same range of values for both the ISM band and the GSM1800 band. However, the GSM1800 channels exhibit a less structured, i.e. more complex, activity on average than the ISM channels. Accordingly, the performance in the ISM band is on average better.

The data collected by us at TCD exhibits a significantly lower LZ complexity than the measurements taken in Aachen (see Fig. 2.3). Accordingly, the performance of RL in the ISM band in TCD is significantly better than the performance in the ISM band for the Aachen dataset.

In any case, all the frequency bands examined by us exhibit the same kind of relationship between average LZ complexity, p_f and the performance of RL, confirming that the LZ complexity is a valid metric for the analysis of learning performance.

2.5.2 *Markov-Based Learning*

We test the Markov process-based learning algorithm on exactly the same sets of channels that we have used for the RL algorithm. Unlike the RL algorithm, to measure the prediction accuracy of the Markov process-based learning algorithm for each combination of channels we need to run the algorithm only once. In other words, averaging the prediction accuracy results over a number of independent simulations is not required because the prediction accuracy of the Markov process-based learning algorithm over the same test set having the same training set will always be the same.

The data set is recorded over 12 h and it consists of 25000 samples. We train the Markov process-based algorithm over the first 1000 data samples (∼first 30 min) and test it over the remaining 24000 samples. At each time slot the algorithm either selects the channel with the highest predicted probability of being free, or predicts that all the channels are busy.

Figures 2.4, 2.5 show the probability of success as a function of average LZ complexity and the probability of at least one free channel existing. We use the same data sets that we have used for measuring the performance of the Q-learning algorithm, and we compute the probability of success over all possible combinations of existing channels in the dataset for $N = 4$.

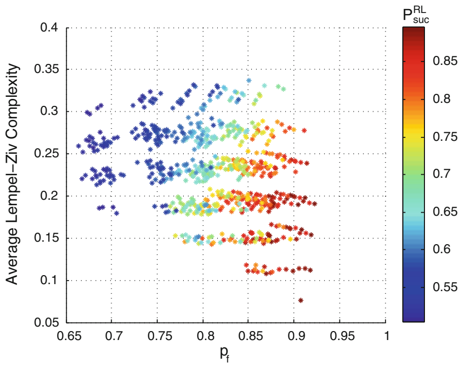

Fig. 2.3 ISM band (indoor), TCD dataset. Probability of success of Q-learning as a function of the average LZ complexity and the probability of at least one free channel existing. Each point represents a particular instance of Q-learning applied to $N = 3$ channels. The total number of possible combinations which we analyzed is $\binom{17}{3} = 680$, where 17 is the number of channels with $DC \in [0.3, 0.8]$

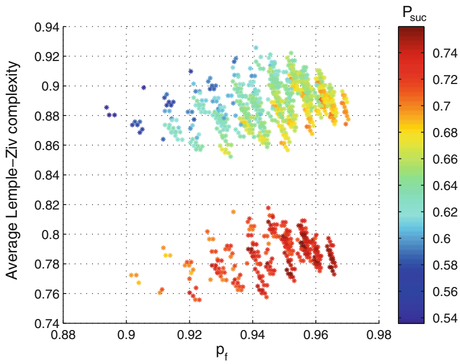

Fig. 2.4 ISM band (outdoor), RWTH Aachen dataset. Probability of success of Markov process-based learning algorithm as a function of the average LZ complexity and the probability of at least one free channel existing. Each point represents a particular instance of the algorithm applied to $N = 4$ channels. The total number of possible combinations which we analyzed is $\binom{15}{4} = 1365$, where 15 is the number of channels with $DC \in [0.3, 0.8]$

Similar to the Q-learning algorithm results, in Figs. 2.4, 2.5, we see that the probability of success using Markov process-based learning algorithm increases with p_f and it also strongly depends on the complexity of the PU behavior. In these two figures we again observe the same phenomenon. For any given p_f, the probability of success has an inverse relation with the average LZ complexity, i.e., when the PU behavior is more complex the probability of success reduces and vice versa.

In Fig. 2.6, we apply the Markov process-based learning algorithm to the data collected from DECT band. For the DECT band, the average LZ complexity is always greater than 0.93. Accordingly, the probability of success is only moderate.

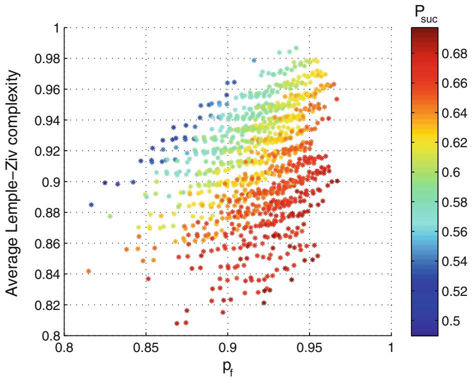

Fig. 2.5 GSM1800 band (outdoor), RWTH Aachen dataset. Probability of success of Markov process-based learning algorithm as a function of the average LZ complexity and the probability of at least one free channel existing. Each point represents a particular instance of Markov process-based learning algorithm applied to $N = 4$ channels. The total number of possible combinations which we analyzed is $\binom{15}{4} = 1365$, where 15 is the number of channels with $DC \in [0.3, 0.8]$

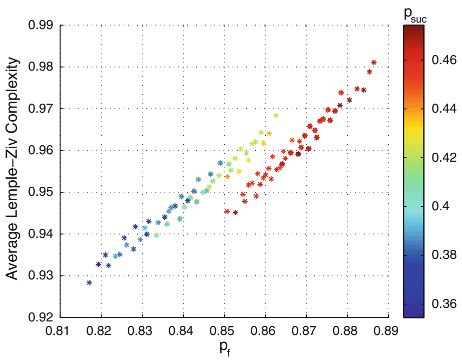

Fig. 2.6 DECT band (indoor), RWTH Aachen dataset. Probability of success of Markov process-based learning algorithm as a function of the average LZ complexity and the probability of at least one free channel existing. Each point represents a particular instance of Markov process-based learning algorithm applied to $N = 4$ channels. The total number of possible combinations which we analyzed is $\binom{10}{4} = 210$, where 10 is the number of channels with $DC \in [0.3, 0.8]$

From the above results, we can conclude that, if possible, it is more beneficial to observe channels (and frequency bands) with higher probability of being free and lower complexity for opportunistic access.

2.6 Conclusions

In this chapter, we presented two learning algorithms that are used to predict the availability of a channel. Moreover, we analyzed the probability of success in finding an unoccupied channel for these algorithms. Our findings show that the performance of both algorithms strongly depends on the behavior of PUs. This means that, although the duty cycle of a channel is the metric usually taken into the account in previous works on learning algorithms applied to DCS, the complexity of PU behavior plays an equally important role.

We also showed that both learning algorithms performed similarly. Both algorithms performed the same for the combinations of channels with high probability of success, while the Markov process-based learning algorithm performed better on the channel combinations with low probability of success. The additional observed channel on Markov process-based scenarios might be the reason of its better performance on low quality channel combinations.

Acknowledgments This material is based upon works supported by the Science Foundation Ireland under Grants No. 10/CE/I1853 and 10/IN.1/I3007.

References

1. Macaluso, I., Forde, T., DaSilva, L., Doyle, L.: Recognition and informed exploitation of grey spectrum opportunities. IEEE Veh. Tech. Mag. **7**(2), 85–90 (2012)
2. Macaluso, I., Finn, D., Ozgul, B., DaSilva, L.: Complexity of spectrum activity and benefits of reinforcement learning for dynamic channel selection. IEEE J. Sel. Areas Commun. (Cogn. Radio Ser.) **31**(11), 2237–2248 (2013)
3. Ahmadi, H., Macaluso, I., DaSilva, L.: The effect of the spectrum opportunities diversity on opportunistic access. In: IEEE ICC'13 (2013)
4. Lempel, A., Ziv, J.: On the complexity of finite sequences. IEEE Trans. Inform. Theory **22**(1), 75–81 (1976)
5. Clancy, T., Walker, B.: Predictive dynamic spectrum access. In: SDR Forum's Technical Conference (2006)
6. Akbar, I., Tranter, W.: Dynamic spectrum allocation in cognitive radio using Hidden Markov Models: Poisson distributed case. In: IEEE SoutheastCon'.07 (2007)
7. Candy, J.: Bayesian Signal Processing: Classical, Modern, and Particle Filtering Methods. Wiley-Interscience, New York (2009)
8. Tumuluru, V., Wang, P., Niyato, D.: A neural network based spectrum prediction scheme for cognitive radio. In: IEEE ICC'10 (2010)
9. Tumuluru, V., Wang, P., Niyato, D.: Channel status prediction for cognitive radio networks. Wireless Commun. Mobile Comput. (2010)
10. Berthold, U., Fu, F., Van der Schaar, M., Jondral, F.: Detection of spectral resources in cognitive radios using reinforcement learning. In: New Frontiers in Dynamic Spectrum Access Networks, 2008. DySPAN 2008. 3rd IEEE Symposium on (2008)
11. Ahmadi, H., Chew, Y.H., Tang, P.K., Nijsure, Y.A.: Predictive opportunistic spectrum access using learning based hidden markov models. In: IEEE PIMRC'11 (2011)
12. Song, C., Zhang, Q.: Intelligent dynamic spectrum access assisted by channel usage prediction. In: IEEE INFOCOM'10 (2010)

13. Kone, V., Yang, L., Yang, X., Zhao, B., Zheng, H.: On the feasibility of effective opportunistic spectrum access. In: Proceedings of the 10th Annual Conference on Internet Measurement. ACM (2010)
14. Lindgren, K., Nordahl, M.: Complexity measures and cellular automata. Complex Syst. **2**(4), 409–440 (1988)
15. Grassberger, P.: Toward a quantitative theory of self-generated complexity. Int. J. Theor. Phys. **25**(9), 907–938 (1986)
16. Kaspar, F., Schuster, H.: Easily calculable measure for the complexity of spatiotemporal patterns. Phys. Rev. A **36**(2), 842–848 (1987)
17. Ziv, J.: Coding theorems for individual sequences. IEEE Trans. Inform. Theory **24**(4), 405–412 (1978)
18. Kolmogorov, A.: Three approaches to the definition of the concept quantity of information. Problemy Peredachi Informatsii **1**(1), 3–11 (1965)
19. Brudno, A.: Entropy and the complexity of the trajectories of a dynamical system. Trans. Moscow Math. Soc. **44**(2), 127–151 (1983)
20. Boffetta, G., Cencini, M., Falcioni, M., Vulpiani, A.: Predictability: a way to characterize complexity. Phys. Rep. **356**(6), 367–474 (2002)
21. Watkins, C., Dayan, P.: Q-learning. Mach. Learn. **8**(3), 279–292 (1992)

Springer

http://www.springer.com/978-3-319-01401-2

Cognitive Communication and Cooperative HetNet Coexistence
Selected Advances on Spectrum Sensing, Learning, and Security Approaches
Di Benedetto, M.-G.; Bader, C.F. (Eds.)
2014, XVI, 365 p. 129 illus., 7 illus. in color., Hardcover
ISBN: 978-3-319-01401-2